Introduction au filtrage de Kalman et à la commande optimale

Olivier Herscovici-Schiller

Version du 28 août 2022

Table des matières

Avant-propos

Ce document a pour but d'accompagner le cours d'introduction au filtrage de Kalman et à la commande optimale pour les élèves-ingénieurs spcialisés en génie physique à l'école d'ingénieurs Denis Diderot. L'enseignement en classe mêle théorie, exercices et applications sous Matlab. De plus, le programme du cours est légèrement différent chaque année en fonction du rythme d'avancée et des intérêts des élèves. La lecture de ce document ne saurait donc dispenser les élèves-ingénieurs de suivre le cours ! La plupart des démonstrations sont données en classe, vous ne trouverez ici que l'idée de la preuve.

J'ai opté pour un formalisme principalement en temps discret, qui permet de maintenir au minimum les difficultés mathématiques, au prix parfois d'un alourdissement des calculs. Les approches retenues sont élémentaires ; l'étudiant·e intéressé·e pourra approfondir les notions présentées ici en s'inscrivant à un cours de commande optimale plus avancé, ou en étudiant les ouvrages présentés dans la bibliographie.

Si vous avez une question, vous pouvez me contacter à l'adresse olivier.herscovici@onera.fr.

Si vous remarquez une erreur, une coquille ou une omission, je vous serais reconnaissant de me la signaler à cette même adresse.

Je remercie les générations successives d'élèves de l'EIDD pour leur rôle dans la mise au point de ce cours, et je remercie Clément Gazzino de m'avoir signalé des coquilles.

1 Stabilité d'un système : motivation, rappels et stabilisation

1.1 Position du problème

L'ingénieur qui conçoit un système veut généralement que ce système se comporte d'une certaine manière. Par exemple, il est hors de question qu'un avion de transport civil se mette à piquer du nez. Pour que le système se comporte comme le veut l'ingénieur, deux questions essentielles se posent.

1. Dans quel état est le système ?
2. Comment amener le système de son état actuel à l'état désiré ?

La théorie de l'estimation et la théorie de la commande tentent de répondre à ces questions. Dans le cadre de ce cours, on se limitera aux approches simples que constitutent le filtrage de Kalman et la commande linéaire quadratique.

L'outil de base de l'ingénieur pour modéliser le comportement d'un système physique est les équations différentielles (ou les équations aux dérivées partielles). Nous allons donc commencer par quelques rappels sur la stabilité des équations différentielles.

1.2 Rappel : solution d'une équation différentielle linéaire scalaire à coefficient constant

Considérons l'équation différentielle linéaire à coefficient constant $a \in \mathbb{R}$ suivante :

$$\frac{\mathrm{d}x}{\mathrm{d}t} = a\,x\,. \tag{1}$$

La solution de cette équation différentielle sur $\mathbb{R}$ est

$$x(t) = x(0)\mathrm{e}^{at}\,. \tag{2}$$

$$\text{Si } a < 0,\ \lim_{t\to+\infty} x(t) = 0\,. \tag{3}$$
$$\text{Si } a > 0,\ \lim_{t\to+\infty} x(t) = \pm\infty\,. \tag{4}$$
$$\text{Si } a = 0,\ \lim_{t\to+\infty} x(t) = x(0)\,. \tag{5}$$

Dans le premier cas, on dira dans le cadre de ce cours que le système est stable. Dans le deuxième cas, on dira dans le cadre de ce cours que système est instable. Dans le troisième cas, qu'on étudiera de toute façon peu en pratique, on dira que le système est marginalement stable.

1.3 Discrétisation d'une équation différentielle et représentation d'état

En pratique, on utilise généralement un calculateur discret pour commander les système. On va donc s'intéresser à la manière d'approximer de manière discrète une évolution continue. On étudie de nouveau l'équation 1.

Discrétisons cette équation différentielle par le schéma d'Euler explicite de pas $\Delta t \in \mathbb{R}_*^+$: $x(t)$ devient x_n, et $\frac{\mathrm{d}x}{\mathrm{d}t}(t)$ devient $\frac{x_{n+1}-x_n}{\Delta t}$. Alors, l'équation 1 devient

$$\frac{x_{n+1}-x_n}{\Delta t} = a\,x_n\,. \tag{6}$$

Ce schéma numérique est dit cohérent (on trouve très souvent l'anglicisme « consistant ») : si on prend $\Delta t \to 0$ dans l'équation précédente, on retrouve l'équation différentielle 1.

Le modèle discret s'écrit donc finalement

$$x_{n+1} = (1 + a\Delta t)x_n\,. \tag{7}$$

1.4 Stabilité du système discrétisé et lien avec le système continu

La dynamique discrète est décrite par l'équation 7, qui est une suite géométrique de raison $1 + a\Delta t$. Le terme général de cette suite est donc

$$\forall n \in \mathbb{N},\ x_n = x_0 \times (1 + a\Delta t)^n\,. \tag{8}$$

Selon la valeur de la raison de la suite, le système peut se comporter de différentes façons.

$$\text{Si } |1 + a\Delta t| < 1,\ \lim_{n\to+\infty} x_n = 0\,. \tag{9}$$
$$\text{Si } |1 + a\Delta t| > 1,\ \lim_{n\to+\infty} |x_n| = \infty\,. \tag{10}$$
$$\text{Si } 1 + a\Delta t = 1,\ x_n = x_0\,. \tag{11}$$
$$\text{Si } 1 + a\Delta t = -1,\ x_n = (-1)^n \times x_0\,. \tag{12}$$

Dans le cadre de ce cours, on dira dans le premier cas que le système discret est stable, dans le deuxième cas que le système discret est instable, et dans les troisième et quatrième cas que le système est marginalement stable.

Étudions à présent le lien entre la stabilité du système continu et la stabilité du système discret. Le système discret est stable à condition que

$$-1 < 1 + a\Delta t < 1\,, \tag{13}$$

ce qui équivaut à

$$-2 < a\Delta t < 0\,. \tag{14}$$

Pour rappel, le système continu est quant à lui stable si et seulement si $a < 0$. On en déduit que la stabilité du système continu est une condition nécessaire mais pas suffisante à la stabilité du système discret (dans le cas du schéma d'Euler explicite). Pour que le système discret soit stable, il faut également que le pas de temps choisi soit suffisamment fin, suivant la condition

$$\Delta t < \frac{2}{-a}. \tag{15}$$

On admettra un cas particulier du théorème de Lax, qui dit que, si le système discret est cohérent et stable, alors la solution du problème discret décrit par l'équation 7 tend vers la solution du problème continu décrit par l'équation 1 quand le pas de discrétisation Δt tend vers 0.

1.5 Stabilité d'un système linéaire discret

Étudions le système linéaire discret de dimension $d \in \mathbb{N}^*$, non commandé, décrit par

$$x_{n+1} = A x_n, \tag{16}$$

où $\forall n \in \mathbb{N}$, $x_n \in \mathbb{R}^d$ et A est une matrice réelle carrée de taille $d \times d$.

Dans le cadre de ce cours, on décrira un système (ou par abus de langage la matrice A associée) comme

— **stable** si $\lim_{n\to\infty} \|x_n\| = 0$;
— **instable** si $\lim_{n\to\infty} \|x_n\| = \infty$;
— **marginalement stable** si $\lim_{n\to\infty} \|x_n\| = \|x_0\|$.

Supposons que A est diagonalisable [1]. Notons

$$A = P^{-1} \Lambda P, \tag{17}$$

où P est une matrice de passage et Λ une matrice diagonale. Dans ce cas, le système linéaire discret peut se réécrire

$$P x_{n+1} = \Lambda P x_n. \tag{18}$$

En notant $\xi = P x$, on a donc

$$\xi_{n+1} = \Lambda \xi_n. \tag{19}$$

En notant $\xi^1, \xi^2, ..., \xi^d$ les composantes de ξ et $\lambda_1, \lambda_2, ..., \lambda_d$ les valeurs propres de A, on a donc

$$\begin{cases} \xi^1_{n+1} = \lambda_1 \xi^1_n \\ \xi^2_{n+1} = \lambda_2 \xi^2_n \\ \quad\vdots \\ \xi^d_{n+1} = \lambda_d \xi^d_n. \end{cases} \tag{20}$$

On constate que chacune de ces équations scalaires est stable si et seulement si chaque valeur propre de A est de module strictement inférieur à 1. La solution du système est donnée par

$$\xi_n = \Lambda^n \xi_0, \tag{21}$$

ou encore

$$x_n = A^n x_0. \tag{22}$$

1. Dans le cas où A n'est pas diagonalisable, on sait tout de même (par la décomposition de Jordan-Chevalley) que A est la somme d'une matrice diagonalisable et d'une matrice nilpotente qui commutent entre elles. Ainsi, la dynamique à long terme est déterminée par la partie diagonalisable de la décomposition, et on retombe sur le cas diagonalisable.

Finalement, on conclut que $\|x_n\| \to 0$ pour $n \to \infty$, c'est à dire que le système est stable, si (et seulement si) les valeurs propres de A sont toutes de module strictement inférieur à 1.

Imaginons à présent que l'on puisse introduire une commande dans le système, le transformant en

$$x_{n+1} = Ax_n + u_n. \tag{23}$$

La quantité u_n est appelée la commande du système. Si le but est de ramener l'état x du système à 0, on remarque qu'il suffit de prendre $u_n = -Ax_n$ pour ramener d'un coup le système à 0.

Ceci est potentiellement très coûteux, et n'est même pas toujours possible. De plus, une telle méthode impose de connaître l'état x_n du système pour choisir la commande à appliquer. Nous reviendrons au problème de la commande dans la section 3.

2 Filtrage de Kalman

2.1 Position du problème

La plupart des systèmes modernes sont commandés. Pour savoir quelle commande appliquer, il peut être utile d'estimer l'état dans lequel se trouve le système. Par exemple, un avion typique est commandé grâce au manche, au palonnier et à la manette des gaz. Une voiture typique est commandée grâce au volant, à la pédale d'accélération, et à la pédale de frein. Le pilote d'un avion veut généralement savoir, entre autres, quel est son vecteur vitesse, quelle est son altitude, et quelle est son orientation dans l'espace pour pouvoir piloter l'avion.

Pour pouvoir estimer l'état du système, on peut utiliser des capteurs. Par exemple, des tubes de Pitot peuvent servir à mesurer la vitesse d'un avion. On peut également utiliser notre connaissance de la physique du système. Par exemple, on sait qu'augmenter les gaz augmenter l'altitude de croisière de l'avion en régime permanent [2]. Enfin, on peut utiliser notre connaissance de l'état antérieur du système : si à un instant donné, l'avion vole à dix kilomètres d'altitude, il paraît très improbable qu'il vole à seulement cent mètres d'altitude trois secondes plus tard.

Le filtre de Kalman est une façon de concilier au mieux [3] l'information apportée par les capteurs et notre connaissance de la physique du système.

Considérons le système dynamique discret suivant :

$$x_{n+1} = Ax_n + Bu_n + \phi_n \tag{24}$$
$$y_{n+1} = Cx_{n+1} + \psi_{n+1}. \tag{25}$$

À chaque instant n, le vecteur x_n est l'état du système (par exemple, la température d'un moteur) ; le vecteur u_n est la commande (par exemple, le débit de carburant dans le moteur) ; le vecteur y_n est la mesure de l'état du système (par exemple, la mesure renvoyée par un thermomètre), ϕ_n est un bruit dit « bruit d'état » qui mesure notre méconnaissance de la physique du problème (par exemple, il peut s'agir des fluctuations du vent) et ψ_n est un bruit dit « bruit de mesure » (par exemple, le bruit électronique du capteur). Les bruits ϕ et ψ sont supposés blancs, gaussiens, centrés, stationnaires et indépendants l'un de l'autre. On notera Φ la covariance de ϕ et Ψ la covariance de ψ.

2.2 Structure d'estimateur linéaire récurrent

Notre but est de produire un estimateur aussi bon et simple que possible de l'état x du système décrit par les équations 24 et 25. Cet estimateur, pour être valable, devra dépendre à chaque

2. Eh oui, augmenter les gaz, *en régime permanent*, augmente l'altitude de croisière de l'avion, et non sa vitesse !
3. Le sens exact de « mieux » sera défini par la suite.

instant au minimum de l'état estimé à l'instant précédent (si à un instant donné, l'avion vole à dix kilomètres d'altitude, il paraît très improbable qu'il vole à seulement cent mètres d'altitude trois secondes plus tard), de la mesure renvoyée par le capteur (il serait dommage de se priver de la mesure des sondes Pitot si on cherche à estimer la vitesse) et de la commande imposée (si on tire sur le manche, on s'attend à monter). Ainsi, notre estimateur sera de la forme

$$\hat{x}_{n+1} = F(\hat{x}_n, y_{n+1}, u_n). \tag{26}$$

Par souci de simplicité, nous choisissons une forme linéaire pour notre estimateur :

$$\hat{x}_{n+1} = A_f \hat{x}_n + B_f u_n + K_{n+1} y_{n+1}, \tag{27}$$

où les matrices A_f, B_f et K_{n+1} sont à déterminer.

2.3 Estimateur (asymptotiquement) sans biais

Pour déterminer les matrices A_f, B_f et K_{n+1}, nous allons imposer des conditions à l'erreur commise par l'estimateur $\hat{x}$.

Définissons l'erreur d'estimation par

$$e = \hat{x} - x. \tag{28}$$

Un estimateur parfait serait un estimateur tel que $\forall n \in \mathbb{N}, e_n = 0$. Malheureusement, vu que l'erreur commise est un processus stochastique (et que nous n'aurons pas accès en pratique à la valeur de l'état x), il ne sera généralement pas possible d'annuler exactement l'erreur. Commençons par écrire l'équation d'évolution de l'erreur.

$$\begin{aligned}
e_{n+1} &= \hat{x}_{n+1} - x_{n+1} && && (29)\\
&= A_f \hat{x}_n + B_f u_n + K_{n+1} y_{n+1} - x_{n+1} && \text{(éq. 27)} && (30)\\
&= A_f \hat{x}_n + B_f u_n + K_{n+1} C x_{n+1} + K_{n+1}\psi_{n+1} - x_{n+1} && \text{(éq. 25)} && (31)\\
&= A_f \hat{x}_n + B_f u_n + (K_{n+1}C - I)x_{n+1} + K_{n+1}\psi_{n+1} && \text{(factorisation)} && (32)\\
&= A_f \hat{x}_n + B_f u_n + (K_{n+1}C - I)(A x_n + B u_n + \phi_n) + K_{n+1}\psi_{n+1} && \text{(éq. 24)} && (33)\\
&= A_f \hat{x}_n + B_f u_n + (K_{n+1}C - I)A\hat{x}_n - (K_{n+1}C - I)Ae_n && && \\
&\quad + (K_{n+1}C - I)B u_n + (K_{n+1}C - I)\phi_n + K_{n+1}\psi_{n+1} && \text{(éq. 28)} && (34)\\
e_{n+1} &= (I - K_{n+1}C)Ae_n + (A_f + K_{n+1}CA - A)\hat{x}_n + (B_f + K_{n+1}CB - B)u_n && && \\
&\quad + (K_{n+1}C - I)\phi_n + K_{n+1}\psi_{n+1} && \text{(factorisation)} && (35)
\end{aligned}$$

On a bien obtenu une équation de récurrence qui décrit l'évolution de l'erreur. On voit bien que l'erreur dépend des réalisations de ϕ et ψ, et qu'il ne sera donc pas possible de l'annuler. En revanche, on peut se concentrer sur son espérance. Comme ϕ et ψ sont centrés, on peut écrire

$$\forall n \in \mathbb{N}, E[\phi_n] = 0 \tag{36}$$

et

$$\forall n \in \mathbb{N}, E[\psi_{n+1}] = 0, \tag{37}$$

où on a noté E l'espérance.

On peut alors écrire l'évolution de l'espérance de l'erreur :

$$E[e_{n+1}] = (I - K_{n+1}C)AE[e_n] + (A_f + K_{n+1}CA - A)\hat{x}_n + (B_f + K_{n+1}CB - B)u_n. \tag{38}$$

On ne peut pas annuler strictement l'erreur, mais, vu que l'espérance de l'erreur suit une dynamique du premier ordre, on peut au moins s'assurer que l'erreur converge vers 0. Concrètement, on cherche à déterminer des conditions suffisantes pour que

$$\lim_{n\to\infty} E[e_n] = 0, \tag{39}$$

On voit qu'il suffit pour cela que les trois conditions suffisantes soient simultanément réunies :

- $B_f + K_{n+1}CB - B = 0$;
- $A_f + K_{n+1}CA - A = 0$;
- la matrice $(I - K_{n+1}C)$ est stable.

Ceci nous fixe deux des trois matrices de notre filtre :

$$A_f = (I - K_{n+1}C)A, \tag{40}$$

$$B_f = (I - K_{n+1}C)B, \tag{41}$$

et on vérifiera en pratique que $(I - K_{n+1}C)$ est stable.

Si on injecte ces conditions dans l'équation 27, on obtient

$$\hat{x}_{n+1} = (I - K_{n+1}C)A\hat{x}_n + (I - K_{n+1}C)Bu_n + K_{n+1}y_{n+1}. \tag{42}$$

2.4 Influence du gain du filtre

On peut développer et refactoriser cette expression pour obtenir une forme barycentrique :

$$\hat{x}_{n+1} = A\hat{x}_n + Bu_n + K_{n+1}[y_{n+1} - C(A\hat{x}_n + Bu_n)]. \tag{43}$$

On remarque sur cette expression que la valeur de K permet de régler un compromis entre la fidélité au modèle numérique donné par $A\hat{x}_n + Bu_n$ et la fidélité à la valeur y_{n+1} lue sur le capteur. L'influence du choix de K est étudiée en travaux pratiques.

2.5 Condition d'optimalité du gain du filtre

Il reste un paramètre libre : le gain K_{n+1} du filtre. Vu que nous avons déjà fixé une condition sur l'espérance, nous allons travailler désormais sur la dispersion. Comme l'espérance de l'erreur est (asymptotiquement) nulle, nous allons travailler sur la grandeur

$$P = E[e \times e^\dagger]. \tag{44}$$

Dans le cas où l'erreur est d'espérance nulle, P est la covariance de l'erreur d'estimation. On rappelle que P est une matrice symétrique positive. Sa trace est telle que

$$\mathrm{tr}(P) = E[e^\dagger e]. \tag{45}$$

Cherchons le gain K_{n+1} qui rend P extrêmale. Une condition nécessaire d'extrêmalité est

$$\forall n \in \mathbb{N}, \frac{\partial P_{n+1}}{\partial K_{n+1}} = 0. \tag{46}$$

2.6 Évolution de la dynamique de l'erreur

Reprenons l'équation 35 qui décrit l'évolution de l'erreur, et injectons les conditions sur A_f et B_f que nous avons déterminées. Cette équation devient

$$e_{n+1} = (I - K_{n+1}C)Ae_n + (K_{n+1}C - I)\phi_n + K_{n+1}\psi_{n+1}. \tag{47}$$

Comme on en aura besoin, calculons aussi la dérivée de cette évolution par rapport à K_{n+1}.

$$\frac{\partial e_{n+1}}{\partial K_{n+1}} = -CAe_n + C\phi_n + \psi_{n+1}. \tag{48}$$

2.7 Équation d'évolution de K

Revenons à la condition de dispersion minimale.

$$\frac{\partial P_{n+1}}{\partial K_{n+1}} = 0 \tag{49}$$

$$\frac{\partial E[e_{n+1}e_{n+1}^\dagger]}{\partial K_{n+1}} = 0 \tag{50}$$

$$E\left[\frac{\partial e_{n+1}}{\partial K_{n+1}} e_{n+1}^\dagger + e_{n+1}\frac{\partial e_{n+1}^\dagger}{\partial K_{n+1}}\right] = 0. \tag{51}$$

Pour obtenir cette dernière équation, il suffit d'avoir

$$E\left[\frac{\partial e_{n+1}}{\partial K_{n+1}} e_{n+1}^\dagger\right] = 0. \tag{52}$$

Injectons les équations 47 et 48. On obtient

$$E\left\{(-CAe_n + C\phi_n + \psi_{n+1})[(I-K_{n+1}C)Ae_n + (K_{n+1}C-I)\phi_n + K_{n+1}\psi_{n+1}]^\dagger\right\} = 0 \tag{53}$$

$$E\left\{(-CAe_n + C\phi_n + \psi_{n+1})[e_n^\dagger A^\dagger(I-K_{n+1}C)^\dagger + \phi_n^\dagger(K_{n+1}C-I)^\dagger + \psi_{n+1}^\dagger K_{n+1}^\dagger]\right\} = 0. \tag{54}$$

À présent, on développe cette expression. Comme ϕ et ψ sont des bruits centrés indépendants entre eux, les espérances de beaucoup des termes croisés sont nulles. Il reste

$$-CAE[e_n e_n^\dagger]A^\dagger(I-K_{n+1}C)^\dagger + CE[\phi_n\phi_n^\dagger](K_{n+1}C-I)^\dagger + E[\psi_{n+1}\psi_{n+1}^\dagger]K_{n+1}^\dagger = 0. \tag{55}$$

On reconnaît les matrices de covariance de e, de ϕ et de ψ, notées P, Φ et Ψ.

$$-CAP_nA^\dagger(I-K_{n+1}C)^\dagger + C\Phi(K_{n+1}C-I)^\dagger + \Psi K_{n+1}^\dagger = 0. \tag{56}$$

En développant et en transposant, il vient

$$K_{n+1}CAP_nA^\dagger C^\dagger + K_{n+1}C\Phi C^\dagger + K_{n+1}\Psi = AP_nA^\dagger C^\dagger + \Phi C^\dagger. \tag{57}$$

Finalement, à condition que l'inverse existe, on obtient

$$K_{n+1} = \left(AP_nA^\dagger + \Phi\right)C^\dagger \times \left(CAP_nA^\dagger C^\dagger + C\Phi C^\dagger + \Psi\right)^{-1}. \tag{58}$$

On n'a pas obtenu la valeur du gain K à proprement parler, mais une équation qui permet de déterminer K_{n+1} à partir de P_n. Il faudrait donc trouver la valeur de P_n...

2.8 Équation de Riccati

Pour cela, repartons de la définition de P :

$$
\begin{aligned}
P_{n+1} &= E\left[e_{n+1}e_{n+1}^\dagger\right] && (59)\\
&= E\left\{\left[(I-K_{n+1}C)Ae_n + (K_{n+1}C-I)\phi_n + K_{n+1}\psi_{n+1}\right]\right.\\
&\qquad \left.\times\left[e_n^\dagger A^\dagger(I-K_{n+1}C)^\dagger + \phi_n^\dagger(K_{n+1}C-I)^\dagger + \psi_{n+1}^\dagger K_{n+1}^\dagger\right]\right\} && \text{(éq. 47)} \quad (60)\\
&= E\left[(I-K_{n+1}C)Ae_ne_n^\dagger A^\dagger(I-K_{n+1}C)^\dagger\right.\\
&\qquad \left.+ (K_{n+1}C-I)\phi_n\phi_n^\dagger(K_{n+1}C-I)^\dagger + K_{n+1}\psi_{n+1}\psi_{n+1}^\dagger K_{n+1}^\dagger\right] && (\phi \text{ et } \psi \text{ sont centrés}) \quad (61)\\
&= (I-K_{n+1}C)AP_nA^\dagger(I-K_{n+1}C)^\dagger\\
&\qquad + (K_{n+1}C-I)\Phi(K_{n+1}C-I)^\dagger + K_{n+1}\Psi K_{n+1}^\dagger && (\text{définition de } P, \Phi \text{ et } \Psi) \quad (62)\\
&= AP_nA^\dagger + K_{n+1}CAP_nA^\dagger C^\dagger K_{n+1}^\dagger - K_{n+1}CAP_nA^\dagger - AP_nA^\dagger C^\dagger K_{n+1}^\dagger\\
&\qquad + K_{n+1}C\Phi C^\dagger K_{n+1}^\dagger - K_{n+1}C\Phi - \Phi C^\dagger K_{n+1}^\dagger + K_{n+1}\Psi K_{n+1}^\dagger && \text{(développement)} \quad (63)\\
P_{n+1} &= AP_nA^\dagger + K_{n+1}\left[CAP_nA^\dagger C^\dagger + C\Phi C^\dagger + \Psi\right]K_{n+1}^\dagger && \text{(factorisation)}\\
&\qquad - K_{n+1}\left[CAP_nA^\dagger + C\Phi\right] - \left[AP_nA^\dagger C^\dagger + \Phi C^\dagger\right]K_{n+1}^\dagger. && (64)
\end{aligned}
$$

On peut simplifier cette expression en injectant la valeur de la première occurrence de K_{n+1}, donnée par l'équation 58. En effet, le terme en inverse dans cette équation est exactement l'inverse du premier facteur de K_{n+1} ! Il reste donc

$$\begin{aligned} P_{n+1} = AP_nA^\dagger + \left(AP_nA^\dagger + \Phi\right) C^\dagger \times I \times K_{n+1}^\dagger \qquad\qquad\qquad\qquad \text{(compensation)} \\ - K_{n+1}\left[CAP_nA^\dagger + C\Phi\right] - \left[AP_nA^\dagger C^\dagger + \Phi C^\dagger\right] K_{n+1}^\dagger. \end{aligned} \tag{65}$$

On voit que les deux termes en $K_{n+1}^\dagger$ se compensent exactement. On obtient donc finalement l'équation

$$P_{n+1} = (I - K_{n+1}C)\left(AP_nA^\dagger + \Phi\right), \tag{66}$$

qui est une équation de Riccati. On n'a donc pas obtenu une expression figée de la valeur de K, ni de la valeur de P, mais une relation de récurrence croisée entre l'évolution du gain du filtre K et la covariance de l'erreur d'estimation P.

2.9 Synthèse

Résumons ce qui précède. On a considéré un système dynamique, donné par les équations 24 et 25 de la forme

$$x_{n+1} = Ax_n + Bu_n + \phi_n \tag{67}$$
$$y_{n+1} = Cx_{n+1} + \psi_{n+1}, \tag{68}$$

où l'état x du système est l'inconnue à estimer, u est une commande choisie et connue, y est une mesure donnée par un capteur, et ϕ et ψ sont des bruits blancs gaussiens centrés stationnaires indépendants entre eux. On cherche un estimateur $\hat{x}$ de x, en imposant par simplicité que cet estimateur soit récurrent linéaire. L'équation 27 donne la forme de cet estimateur. En imposant la condition que l'erreur, définie comme $e = \hat{x} - x$, s'annule en espérance, on obtient la forme d'estimateur donnée par l'équation 43.

Si on décide de choisir le gain K de sorte à minimiser la covariance $P = E\left[ee^\dagger\right]$ de l'erreur, on obtient un système de récurrence croisée à chaque étape, donnée par les équations 58 et 66.

En conclusion, on a établi l'algorithme récursif suivant pour estimer l'état du système. C'est l'algorithme du filtre de Kalman (dans une version discrète simplifiée).

Étape 1 : On initialise $\hat{x}$ à $\hat{x}_0$. On peut par exemple lire la valeur y_0 renvoyée par le capteur.

Étape 2 : On initialise P à P_0. On peut par exemple prendre la valeur de la covariance Ψ du bruit du capteur.

Étape 3 : On fait évoluer K selon

$$K_{n+1} = \left(AP_nA^\dagger + \Phi\right) C^\dagger \times \left(CAP_nA^\dagger C^\dagger + C\Phi C^\dagger + \Psi\right)^{-1}. \tag{69}$$

Étape 4 : On fait évoluer P selon

$$P_{n+1} = (I - K_{n+1}C)\left(AP_nA^\dagger + \Phi\right). \tag{70}$$

Étape 5 : On fait évoluer $\hat{x}$ selon

$$\hat{x}_{n+1} = A\hat{x}_n + Bu_n + K_{n+1}[y_{n+1} - C(A\hat{x}_n + Bu_n)]. \tag{71}$$

2.10 Régime permanent et équation de Riccati algébrique

Dans un grand nombre de cas, la convergence de K et de P vers des valeurs limites est très rapide. Aussi, on peut préférer fixer une bonne fois pour toute la valeur de K et celle de P, sans avoir à les recalculer à chaque itération. Supposons que les suites K et P admettent chacune une valeur limite, notée respectivement K et P en surchargeant la notation. Dans ce cas, les équations de récurrence croisée 58 et 66 se simplifient. L'équation 58 devient une équation algébrique :

$$K = \left(APA^\dagger + \Phi\right) C^\dagger \times \left(CAPA^\dagger C^\dagger + C\Phi C^\dagger + \Psi\right)^{-1}. \tag{72}$$

L'équation 66 devient

$$P = (I - KC)\left(APA^\dagger + \Phi\right). \tag{73}$$

En la développant et en injectant la valeur de K donnée par l'équation 72, on obtient une équation de Riccati algébrique discrète :

$$APA^\dagger + \Phi - P - \left(APA^\dagger + \Phi\right) C^\dagger \times \left(CAPA^\dagger C^\dagger + C\Phi C^\dagger + \Psi\right)^{-1} C \left(APA^\dagger + \Phi\right) = 0. \tag{74}$$

Il existe des méthodes — qui ne seront pas traitées dans ce cours — de résolution des équations de Riccati algébriques. La mise en pratique sous Octave est traitée dans l'annexe B.

3 Commande par retour d'état optimale des systèmes linéaires avec coût quadratique

3.1 Position du problème

On cherche à stabiliser un système linéaire dont la dynamique est connue. De plus, on cherche à le faire en trouvant un compromis entre qualité de la stabilisation et coût de la stabilisation, le tout avec des coûts quadratiques. Formellement, on considère le système dynamique linéaire suivant :

$$x_{n+1} = Ax_n + Bu_n \tag{75}$$
$$y_{n+1} = Cx_{n+1} \tag{76}$$

et on se pose le problème de trouver la valeur optimale du paramètre L tel que la commande $u = -Lx$ stabilise le système sous contrainte de minimiser le critère

$$J = \sum_{n=0}^{\infty} y_n^\dagger Q y_n + u_n^\dagger R u_n, \tag{77}$$

où Q et R sont des matrices symétriques définies positives.

Le choix de formes quadratiques est guidé par deux considérations. La première, d'ordre mathématique, est que les formes quadratiques sont bien connues, en particulier dans le contexte de l'optimisation. La seconde, d'ordre plus physique, est que les formes quadratiques peuvent souvent être interprétées comme des énergies.

Le choix du retour linéaire de l'état est guidé par la simplicité mathématique, et par le fait que, hors toutes contraintes, il suffit de prendre $L = A$ pour stabiliser directement le système.

Les matrices Q et R définissent un produit scalaire. On remarquera qu'il suffit de choisir $C = I$ dans le critère précédent avoir un critère de coût qui dépend directement de l'état et non de l'observable.

3.2 Évaluation du critère

Tirons les premières conséquences du choix de la forme du retour d'état.

$$x_{n+1} = Ax_n + Bu_n \tag{78}$$
$$x_{n+1} = Ax_n - BLx_n \qquad (\text{car}\, u = -Lx) \tag{79}$$
$$x_n = (A - BL)^n x_0 \qquad (\text{terme général d'une suite géométrique}). \tag{80}$$

Pour stabiliser le système, il est donc nécessaire que la matrice $A - BL$ soit stable. Injectons la forme de x dans le critère de coût J.

$$J = \sum_{n=0}^{\infty} y_n^\dagger Q y_n + u_n^\dagger R u_n \qquad (\text{par définition de } J) \tag{81}$$
$$= \sum_{n=0}^{\infty} x_n^\dagger C^\dagger Q C x_n + x_n^\dagger L^\dagger R L x_n \qquad (\text{par définition de } y \text{ et choix de } u) \tag{82}$$
$$= \sum_{n=0}^{\infty} x_n^\dagger (C^\dagger Q C + L^\dagger R L) x_n \qquad (\text{factorisation}) \tag{83}$$
$$J = x_0^\dagger \sum_{n=0}^{\infty} (A - BL)^{\dagger n} (C^\dagger Q C + L^\dagger R L)(A - BL)^n x_0 \qquad (\text{injection de l'éq. 80}). \tag{84}$$

On notera dans la suite

$$\mathscr{P} = \sum_{n=0}^{\infty} (A - BL)^{\dagger n} (C^\dagger Q C + L^\dagger R L)(A - BL)^n, \tag{85}$$

de telle sorte que

$$J = x_0^\dagger \mathscr{P} x_0. \tag{86}$$

La matrice $\mathscr{P}$ est clairement symétrique définie positive (on met à part les cas dégénéré tels que $Q = 0$ et $R = 0$). Attention, ce $\mathscr{P}$ est évidemment différent du P défini comme la covariance de l'erreur d'estimation dans le cas du filtrage de Kalman.

À présent que l'on a déterminé l'expression du critère J à minimiser, il faut le minimiser effectivement. Pour cela, nous allons introduire les équations de Lyapounoff.

3.3 Théorème de Lyapounoff et équations de Lyapounoff

Nous désirons trouver une commande qui stabilise le système, aussi la matrice $A - BK$ doit être stable. Vu que le système est stable,

$$\forall x_n \neq 0, \|x_{n+1}\| < \|x_n\|. \tag{87}$$

Ceci est vrai pour n'importe quelle norme. En particulier, pour la norme issue du produit scalaire défini par la matrice symétrique définie positive $\mathscr{P}$,

$$\forall x_n \neq 0, x_{n+1}^\dagger \mathscr{P} x_{n+1} < x_n^\dagger \mathscr{P} x_n. \tag{88}$$

En injectant la valeur de x_{n+1} en fonction de x_n, on trouve

$$\forall x_n \neq 0, x_n^\dagger (A - BL)^\dagger \mathscr{P} (A - BL) < x_n^\dagger \mathscr{P} x_n. \tag{89}$$

Finalement, en factorisant le tout,

$$\forall x_n \neq 0, x_n^\dagger \left[\mathscr{P} - (A - BL)^\dagger \mathscr{P} (A - BL)\right] x_n > 0. \tag{90}$$

Comme cette inégalité stricte est valable pour tout x_n non nul, on a démontré que $\mathscr{P} - (A - BL)^\dagger \mathscr{P} (A - BL)$ est définie positive[4]. Remarquons que ces implications sont en fait des équivalences : on peut remonter les calculs en supposant que $\mathscr{P} - (A - BL)^\dagger \mathscr{P} (A - BL)$ est définie positive pour démontrer qu'alors $A - BK$ est stable.

Finalement, on a démontré une version du théorème de Lyapounoff discret : le système défini par

$$x_{n+1} = (A - BL) x_n \tag{91}$$

est stable si et seulement si
il existe $\mathscr{P}$ symétrique définie positive telle que $\mathscr{P} - (A - BL)^\dagger \mathscr{P} (A - BL)$ est symétrique définie positive, c'est à dire si et seulement si il existe un matrice M symétrique définie positive telle que

$$(A - BL)^\dagger \mathscr{P} (A - BL) - \mathscr{P} + M = 0. \tag{92}$$

Une telle équation, de forme

$$X^\dagger \mathscr{P} X - \mathscr{P} + M = 0, \tag{93}$$

où $\mathscr{P}$ et M sont symétriques définies positives, et appelée une équation de Lyapounoff discrète.

4. Au passage, cette matrice est de plus clairement symétrique. C'est donc la matrice d'un produit scalaire.

3.4 Résolution de l'équation de Lyapounoff

On cherche à résoudre en $\mathscr{P}$ l'équation de Lyapounoff suivante :

$$(A-BL)^{\dagger}\mathscr{P}(A-BL)-\mathscr{P}+M=0. \tag{94}$$

Si on résout cette équation en dimension 1, on doit résoudre

$$(A-BL)^{2}\mathscr{P}-\mathscr{P}+M=0, \tag{95}$$

et on trouve immédiatement

$$\mathscr{P}=\frac{M}{1-(A-BL)^{2}}. \tag{96}$$

Pour généraliser cette formule aux dimensions supérieures, on écrit l'inverse sous forme de série :

$$\mathscr{P}=\sum_{n=0}^{\infty}(A-BL)^{n}\times M\times(A-BL)^{n}. \tag{97}$$

On va donc choisir comme solution candidate (à tester) de l'équation en dimension quelconque la matrice

$$\mathscr{S}=\sum_{n=0}^{\infty}(A-BL)^{\dagger n}M(A-BL)^{n}. \tag{98}$$

Calculons

$$(A-BL)^{\dagger}\mathscr{S}(A-BL)-\mathscr{S}+M \tag{99}$$

pour tester cette candidate.

$$\begin{aligned}
(A-BL)^{\dagger}\mathscr{S}(A-BL)-\mathscr{S}+M &= (A-BL)^{\dagger}\sum_{n=0}^{\infty}(A-BL)^{\dagger n}M(A-BL)^{n}(A-BL) \\
&\qquad -\sum_{n=0}^{\infty}(A-BL)^{\dagger n}M(A-BL)^{n}+M \qquad (100)\\
&= \sum_{n=0}^{\infty}(A-BL)^{\dagger n+1}M(A-BL)^{n+1} \\
&\qquad -\sum_{n=0}^{\infty}(A-BL)^{\dagger n}M(A-BL)^{n}+M \qquad (101)\\
&= \sum_{n=1}^{\infty}(A-BL)^{\dagger n}M(A-BL)^{n} \\
&\qquad -M-\sum_{n=1}^{\infty}(A-BL)^{\dagger n}M(A-BL)^{n}+M \qquad (102)\\
&= 0. \qquad (103)
\end{aligned}$$

Notre solution candidate est donc bien une solution de cette équation.

Conclusion : l'équation

$$(A-BL)^{\dagger}\mathscr{P}(A-BL)-\mathscr{P}+M=0 \tag{104}$$

a pour solution

$$\mathscr{P}=\sum_{n=0}^{\infty}(A-BL)^{\dagger n}M(A-BL)^{n}. \tag{105}$$

3.5 Établissement de l'équation de Lyapounoff du problème

On a démontré que, comme $A - BL$ est stable, l'équation de Lyapounoff

$$(A-BL)^\dagger \mathscr{P}(A-BL) - \mathscr{P} + M = 0, \tag{106}$$

où $\mathscr{P}$ et M sont symétriques définies positives, est vérifiée. De plus, on a démontré que

$$\mathscr{P} = \sum_{n=0}^{\infty} (A-BL)^{\dagger n} M (A-BL)^n. \tag{107}$$

En comparant cette formule à l'équation 85, on trouve que

$$M = C^\dagger Q C + L^\dagger R L. \tag{108}$$

Ainsi, l'équation de Lyapounoff suivie par $\mathscr{P}$ est

$$(A-BL)^\dagger \mathscr{P}(A-BL) - \mathscr{P} + C^\dagger Q C + L^\dagger R L = 0. \tag{109}$$

3.6 Établissement de la valeur du gain

Nous cherchons à trouver L qui minimise le critère J, c'est à dire de manière complétement équivalente la valeur de L qui minimise $\mathscr{P}$. Pour que $\mathscr{P}$ soit minimal, il faut que

$$\frac{\mathrm{d}\mathscr{P}}{\mathrm{d}L} = 0. \tag{110}$$

Nous allons dériver l'équation de Lyapounoff 109 par rapport à L. Développée, l'équation 109 s'écrit

$$A^\dagger \mathscr{P} A - L^\dagger B^\dagger \mathscr{P} A - A^\dagger \mathscr{P} B L + L^\dagger B^\dagger \mathscr{P} B L - \mathscr{P} + C^\dagger Q C + L^\dagger R L = 0. \tag{111}$$

On dérive par rapport à L, en gardant en tête que, pour la commande que nous cherchons,

$$\frac{\mathrm{d}\mathscr{P}}{\mathrm{d}L} = 0. \tag{112}$$

On obtient que

$$-B^\dagger \mathscr{P} A - A^\dagger \mathscr{P} B + L^\dagger B^\dagger \mathscr{P} B + B^\dagger \mathscr{P} B L + L^\dagger R + R L = 0. \tag{113}$$

Pour satisfaire cette condition, il suffit de choisir L telle que

$$-B^\dagger \mathscr{P} A + B^\dagger \mathscr{P} B L + R L = 0. \tag{114}$$

On choisit donc

$$L = \left(B^\dagger \mathscr{P} B + R\right)^{-1} B^\dagger \mathscr{P} A. \tag{115}$$

Exactement comme dans le cas du filtre de Kalman, on trouve la valeur du gain L en fonction de $\mathscr{P}$. Il va donc falloir trouver la valeur de $\mathscr{P}$…

3.7 Établissement de l'équation de Riccati algébrique

Reprenons l'équation 111 et factorisons par $L^\dagger$.

$$A^\dagger \mathscr{P} A + L^\dagger \left(-B^\dagger \mathscr{P} A + L^\dagger R L\right) - A^\dagger \mathscr{P} B L + B^\dagger \mathscr{P} B L - \mathscr{P} + C^\dagger Q C = 0 \tag{116}$$

$$L^\dagger \left[\left(B^\dagger \mathscr{P} B + L^\dagger R\right) L - B^\dagger \mathscr{P} A\right] - A^\dagger \mathscr{P} B L + A^\dagger \mathscr{P} A - \mathscr{P} + C^\dagger Q C = 0. \tag{117}$$

On injecte la valeur de L donnée par l'équation 115, et tout le facteur de $L^\dagger$ disparaît. On reconnaît finalement l'équation de Riccati algébrique discrète

$$A^\dagger \mathscr{P} A - \mathscr{P} - A^\dagger \mathscr{P} B \left(B^\dagger \mathscr{P} B + R\right)^{-1} B^\dagger \mathscr{P} A + C^\dagger Q C = 0. \tag{118}$$

La résolution de cette équation de Riccati fournit le retour d'état optimal (dans le cas linéaire quadratique discret à horizon infini) au sens du critère J pour stabiliser le système, ainsi que la valeur du critère de coût associé pour une condition initiale x_0 donnée.

A Réalisation d'une variable aléatoire centrée en Octave ou Matlab

En Octave, la commande `randn()` permet de produire une réalisation d'une variable aléatoire scalaire qui suit un loi gaussienne standard, c'est à dire une loi normale centrée réduite. On rappelle que si $X \sim \mathcal{N}(0,1)$, alors $\psi = \sigma X$ suit une loi normale centrée d'écart-type σ. Ainsi, par exemple, la commande `3*randn()` permet de produire une réalisation d'une variable aléatoire gaussienne de variance $3^2 = 9$.

B Résolution de l'équation de Riccati algébrique discrète en Octave ou Matlab

En Octave ou Matlab, la commande permettant de résoudre une équation de Riccati algébrique discrète est `dare`, pour *discrete algebraic Riccati equation*. En Octave, elle est fourni par le *control package*. On pourra se reporter à l'aide d'Octave ou de Matlab pour la syntaxe de cette commande. On peut remarquer que pour les changements de notation présentés dans le tableau B, la commande `dare` correspond exactement à la résolution de l'équation 74.

Variable dans le script	Notation dans ce cours
A	$A^\dagger$
B	$A^\dagger C^\dagger$
Q	Φ
R	$C\Phi C^\dagger + \Psi$
S	$\Phi C^\dagger$
E	I
X	P
G	$K^\dagger$

TABLE 1 – Correspondance entre notations du cours et commande `dare`.

C Un peu de pratique : filtre de Kalman pour un oscillateur

On considère un modèle simple d'oscillateur de raideur (éventuellement négative) variable, avec un frottement (éventuellement négatif) variable. On note $p(t)$ la position de la particule à tout instant t, et $v(t)$ sa vitesse à tout instant.

On rappelle que la vitesse est définie par l'équation différentielle suivante :

$$\frac{\mathrm{d}p(t)}{\mathrm{d}t} = v(t). \tag{119}$$

Par ailleurs, l'évolution du système suit l'équation différentielle suivante :

$$\frac{\mathrm{d}v(t)}{\mathrm{d}t} = a(t)p(t) + b(t)v(t). \tag{120}$$

Notons X_c le vecteur d'état du système, défini à tout instant t par

$$X_c(t) = \begin{pmatrix} p(t) \\ v(t) \end{pmatrix}. \tag{121}$$

1. Montrer que l'évolution de X_c obéit à une équation de la forme

$$\frac{\mathrm{d}}{\mathrm{d}t} X_c(t) = A_c(t) X_c(t), \tag{122}$$

   où $A_c(t)$ est une matrice dont on précisera la valeur des coefficients en fonction des valeurs de $a(t)$ et $b(t)$.

2. On choisit une discrétisation de type Euler explicite :

$$X_c(t) \mapsto X_n \tag{123}$$

   et

$$\frac{\mathrm{d}}{\mathrm{d}t} X_c(t) \mapsto (X_{n+1} - X_n)/\Delta t. \tag{124}$$

   Montrer que cette discrétisation mène à un système discret de la forme

$$X_{n+1} = A X_n, \tag{125}$$

   où on précisera la valeur de la matrice A en fonction de A_c et Δt. On pourra noter Id l'identité des matrices réelles de taille 2×2.

3. À partir de maintenant et pour toute la suite de l'exercice, on prendra

$$X_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

   on prendra

$$A_c = \begin{pmatrix} 0 & 1 \\ -0.8 & -0.3 \end{pmatrix}$$

   et on prendra $\Delta t = 0.1$. Simuler l'évolution du système pour $T = 200$ pas de temps. Afficher sur un graphique l'évolution de la position p du système au cours du temps. Est-ce bien l'allure attendue d'un oscillateur harmonique amorti ?

4. À présent, et uniquement pour cette question, on prend $\Delta t = 1$. Afficher sur un graphique l'évolution de la position p du système au cours du temps. Est-ce bien l'allure attendue d'un oscillateur harmonique amorti ? Comment expliquez-vous cette courbe ? À présent, on reprend $\Delta t = 0.1$.

5. À présent, on décide de simuler un oscillateur harmonique où s'ajoute une force aléatoire : $X_{n+1} = AX_n + \phi_n$, où ϕ_n est défini à tout instant par

$$\phi_n = \begin{pmatrix} 0 \\ b_n \end{pmatrix}, \tag{126}$$

où b_n est un bruit blanc gaussien centré stationnaire d'écart-type $\sigma = 0.04$. La matrice de covariance de ϕ est donc

$$V_\phi = \begin{pmatrix} 0 & 0 \\ 0 & \sigma^2 \end{pmatrix}. \tag{127}$$

Vérifiez que vous avez bien repris $\Delta t = 0.1$. Afficher sur un graphique l'évolution de la position p du système au cours du temps.

6. À présent, on ajoute un capteur de position : à tout instant, $y_{n+1} = C x_{n+1} + \psi_{n+1}$. Le bruit ψ est un bruit blanc gaussien centré d'écart-type 0.1, et donc de variance $V_\psi = 0.01$. La matrice C vaut $C = [1 \quad 0]$. (Attention, il s'agit bien d'une matrice ligne, de sorte que $CX = p$). On introduit l'estimateur

$$\hat{X}_{n+1} = A\hat{X}_n + K[y_{n+1} - CA\hat{X}_n] \tag{128}$$

On prend $\hat{X}_0 = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$.

Afficher sur le même graphe l'évolution au cours du temps de p, de l'estimée $\hat{p}$ de p et de la mesure y pour le cas où $K = 0$. Commenter.

7. À présent, on met en place un filtre de Kalman, c'est à dire que K est donnée à chaque instant par

$$K_{n+1} = (AP_nA^TC^T + V_\phi) \times C^T \times (CAP_nA^TC^T + CV_\phi C^T + V_\psi)^{-1} \tag{129}$$

où P suit l'équation de récurrence

$$P_{n+1} = (Id - K_{n+1}C)(AP_nA^T + V_\phi). \tag{130}$$

On initialisera P par $P_0 = Id$.

Afficher sur le même graphe l'évolution au cours du temps de p, de l'estimée $\hat{p}$ de p et de la mesure y.

8. Comparer la courbe obtenue à la question 7 à la courbe obtenue à la question 6.

9. Quelles sont la valeur finale P_T et la valeur finale K_T, toujours pour $P_0 = Id$? (Aucun calcul n'est nécessaire, il suffit d'afficher la valeur calculée par Matlab/Octave.) À présent, on prend $P_0 = 0 \times Id$, c'est à dire que l'on initialise le filtre de Kalman avec une covariance de l'erreur supposée nulle (ce qui est bien sûr faux). Afficher sur le même graphe l'évolution au cours du temps de p, de l'estimée $\hat{p}$ de p et de la mesure y. Comparer à la courbe de la question précédente.

10. Notez la valeur finale P_T et la valeur finale K_T, pour $P_0 = 0 \times Id$. Comparez aux valeurs finales de P et K pour $P_0 = Id$ que vous avez notées à la question précédente. Commenter.

11. À présent, on prend $V_\psi = 0$ et $C = Id$. Calculer analytiquement la valeur de K à chaque instant. Afficher sur le même graphe l'évolution au cours du temps de p, de l'estimée $\hat{p}$ de p et de la mesure y. Commenter.

12. À présent, on reprend $V_\psi = 0.01$ et $C = [1 \quad 0]$. Expliquez comment vous feriez pour ramener au plus vite la position vers 0 si vous avez accès à la valeur de X.

13. Et si vous n'avez pas accès à la valeur de X ? Affichez le graphe correspondant.
14. Enfin, proposez comment stabiliser le système d'une façon qui minimise le critère

$$J_\alpha = \alpha \times X^2 + u^2. \tag{131}$$

15. Tracez la solution qui minimise J_α pour $\alpha = 0, \alpha = 0.1, \alpha = 1, \alpha = 10$. Commentez.