Automatique continue, automatique discrète, informatique industrielle : le triangle des Bermudes ?

Paul Caspi

VERIMAG-CNRS
2, avenue de Vignate
F-38610, Gières
caspi@imag.fr

RÉSUMÉ. *Dans la plupart des systèmes automatiques complexes, automatismes continus et discrets sont étroitement mêlés. Pour l'implantation sur (réseaux de) calculateurs numériques, les parties continues reposent sur la théorie bien établie des systèmes échantillonnés périodiques ou multi-périodiques alors que la théorie des systèmes à événements discrets est plutôt préconisée pour les parties discrètes. Devant la difficulté de mélanger ces deux types d'implantations, l'expérience montre que les praticiens ont tendance à privilégier les échantillonnages périodiques, quitte à pallier l'absence de théorie satisfaisante d'échantillonnage des systèmes à événements discrets par des recettes « maison ». Dans cette présentation, nous nous interrogeons sur la signification de ce hiatus théorique qui nous semble révéler un certain manque de communication entre automatique continue, automatique discrète et informatique industrielle.*

ABSTRACT. *In most complex control systems, continuous and discrete event controls are closely mixed. When these systems are to be implemented on (networks of) computers, continuous parts can be dealt with according to the theory of sampled (and multiply sampled) control systems while discrete parts rely on the theory of discrete event control systems. But these two approaches don't easily combine with each other and experience shows that, in practice, people tend to equally sample both parts. In the absence of a convenient sampling theory for discrete event systems, people rely on "in-house" tricks to ensure systems robustness. In this presentation, we try to partially fill this theoretical gap which seems to indicate some lack of communication between continuous control, discrete event control and computer people.*

MOTS-CLÉS : *systèmes continus, discrets, hybrides, échantillonnage, robustesse.*

KEYWORDS: *continuous systems, discrete event systems, hybrid systems, sampling, robustness.*

1. Introduction

Depuis plus de vingt ans, les réalisations analogiques ou en composants discrets des systèmes de contrôle-commande ont cédé la place aux réalisations sur calculateurs numériques, voire sur des réseaux locaux de calculateurs et des exemples de grande qualité peuvent être cités comme les commandes de vol électriques des Airbus à partir de l'A320 [BRI 94], les contrôles-commandes des centrales nucléaires Framatome réalisées par Merlin-Gérin [BER 88] ou les systèmes de métros automatiques de Matra-Transport [BEH 98]. Tous ces exemples ont en commun, outre d'être des systèmes critiques exigeant une haute sûreté de fonctionnement, un certain nombre de caractéristiques :

– automatismes continus et à événements discrets y sont intimement mêlés,

– ils sont fondés sur des principes d'architecture logicielle et matérielle que nous avons qualifiés de *quasi-synchrones* [CAS 99], présentant les aspects suivants :

- échantillonnage périodique des acquisitions et calculs de chaque calculateur,

- absence de synchronisation entre les périodes des différents calculateurs, par contraste avec la technique dite « time-triggered architecture » [KOP 97],

- communications entre calculateurs susceptibles de réalisations diverses (lignes séries, bus de terrain, etc...) mais qui aboutissent au principe fonctionnel de la mémoire partagée.

Outre ces exemples, ces caractéristiques se retrouvent sans doute dans un grand nombre de systèmes de contrôle-commande, sinon dans une majorité d'entre eux.[1] Il est donc intéressant et important d'en étudier les fondements théoriques. C'est ce que nous avons fait au cours du projet européen Crisys (1997-2001) avec, entre autres partenaires, Schneider Electric, Airbus France, et le laboratoire d'automatique de Grenoble (H.Alla et R.David) et ce sont essentiellement les constatations que nous avons faites à ce sujet que nous présentons ici. Ces constatations sont les suivantes :

Si la théorie de l'échantillonnage périodique est bien établie et depuis longtemps en ce qui concerne les systèmes continus (voir par exemple [AST 84]), il n'en est pas de même pour les systèmes à événements discrets, et cela pose des problèmes d'indéterminismes et d'aléas qui sont mal pris en compte par les outils de validation fonctionnelle actuels (simulation, test, vérification formelle). Cette constatation est développée en section 2. Dans cette situation, les praticiens ont recours à des recettes « maison » que nous décrivons en section 3 et que nous essayons de théoriser ensuite. Nous commençons par présenter une théorie naïve de l'échantillonnage des signaux et systèmes continus en section 4. Ensuite, en section 5, nous présentons une approche empirique de l'échantillonnage des signaux discrets. Enfin, nous proposons une fusion des deux précédents points de vue en section 6.[2]

---

1. Bien que nous ne puissions étayer cette affirmation par des statistiques précises.

2. Ces travaux ont été soutenus par le projet européen CRISYS, par le réseau d'excellence ARTIST et par les contrats AIRBUS-VERIMAG.

Nous concluons enfin en nous étonnant du peu de considération que ces problèmes ont rencontré par le passé, ce qui est pour nous un signe d'un manque de communication à l'intersection des disciplines de l'automatique continue, de l'automatique discrète et de l'informatique industrielle.

2. Echantillonnage des systèmes à événements discrets

Dans cette section, nous illustrons les problèmes soulevés par l'échantillonnage périodique des systèmes à événements discrets.

2.1. *Echantillonnage et déterminisme*

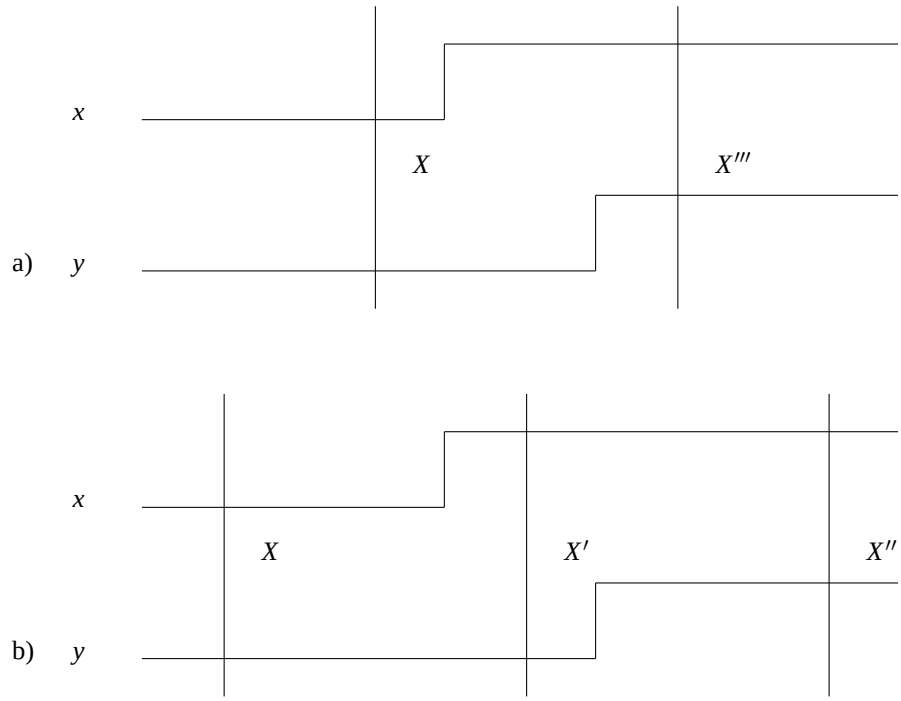
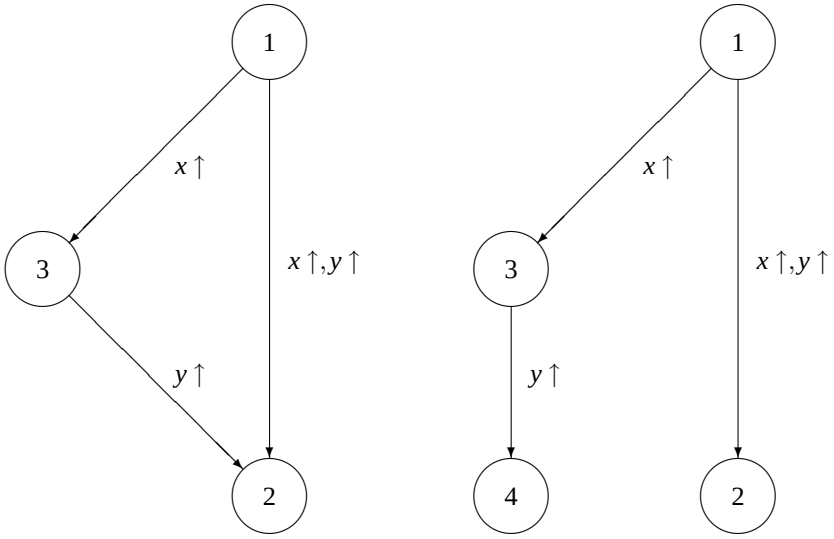


Figure 1. *L'échantillonnage n'est pas déterministe*

La figure 1 montre deux échantillonnages périodiques possibles d'un même couple de signaux booléens (x, y). Dans le premier cas, x and y passent en même temps logique de 0 à 1, alors que dans le second cas il y a un passage intermédiaire.

Comme la phase de l'échantillonnage ne peut en général pas être contrôlée, il en résulte une situation indéterministe.



a) une course non critique b) une course critique

Figure 2. *Exemples de courses*

2.2. *Courses critiques*

Supposons maintenant que le couple (x,y) soit une entrée d'un automate. La figure 2 illustre le phénomène de course. En 2a, l'aléa d'échantillonnage précédent se traduit par une divergence transitoire de l'état de l'automate. En revanche, en 2b, la divergence est plus longue et peut éventuellement conduire à des écarts d'état permanents. C'est une course critique. Il est clair que les concepteurs de systèmes doivent absolument prendre en compte ce phénomène et le contrôler.

2.3. *Courses critiques et redondances*

L'attention des concepteurs vis-à-vis des aléas d'échantillonnage est toujours importante et nous ne voulons pas ici privilégier certaines situations plutôt que d'autres. Néanmoins les méthodes de redondance illustrent bien les écueils que cela réserve.

Dans les systèmes critiques, il est fréquent que les mêmes calculs soient répliqués sur plusieurs calculateurs à des fins de détections ou de tolérance aux fautes. L'idée est que des désaccords entre calculateurs permettront soit de détecter des fautes soit de les tolérer par des méthodes de votes majoritaires. Supposons que l'automate de la figure 2b soit ainsi répliqué sur plusieurs calculateurs, dont les horloges d'échantillon-

nage ne soient pas synchronisées. Pour les mêmes entrées vues à la figure 1, certains calculateurs vont emprunter le chemin $1 \rightarrow 2$ tandis que d'autres passeront par par le chemin $1 \rightarrow 3 \rightarrow 4$. On voit que, même en l'absence de faute, des désaccords durables peuvent survenir, détruisant ainsi toute propriété de tolérance aux fautes. Ces phénomènes, bien connus des gens de systèmes répartis, rappellent bien évidemment les problèmes dits de généraux byzantins [PEA 80].

2.4. *Aléas d'échantillonnage et validation*

Ces difficultés ont pour effet de rendre la mise au point et la validation de systèmes échantillonnés très difficile. Remarquons tout d'abord que les outils classiques de construction et simulation de systèmes, tels que Simulink prennent très mal en compte ces phénomènes : l'échantillonnage en Simulink est un échantillonnage « idéal » qui ne souffre pas des imperfections des horloges physiques disponibles : non maîtrise du déphasage, dérives, etc... Si on fait l'effort de modéliser ces aléas, ce que nous avons fait dans Crisys en utilisant l'outil Scade/Lustre d'Esterel-Technologies[3] [CAS 01b], on s'aperçoit alors que cela est de peu d'utilité car les événements de courses critiques se produisent très rarement et il faudrait donc des temps de simulation exorbitants pour parvenir à quelque chose de concluant.[4]

Les mêmes difficultés se rencontrent bien évidemment si, au lieu de simulations, on utilise des méthodes de vérification formelle. Le pendant de temps de simulation très longs est alors l'explosion combinatoire, qui rend les outils extrêmement inefficaces.

3. Méthodes utilisées en pratique

Pour se prémunir contre ces phénomènes, les concepteurs ont recours à des recettes souvent « maison » ou empruntées à la théorie des circuits asynchrones [BRZ 95]. Ces méthodes peuvent se classer en deux catégories : adjonctions de retards ou adjonctions de causalités.[5]

---

3. *http ://www.esterel-technologies.com*
4. Cela n'est pas étonnant, compte-tenu de l'analogie déjà notée avec les phénomènes byzantins. Les gens de la communauté dite de la tolérance aux fautes savent que ceux-ci se produisent très rarement et qu'on ne doit en tenir compte que dans les systèmes très critiques. En revanche, dans ces derniers, des algorithmes spécifiques et démontrés une fois pour toutes doivent être employés.
5. Dans le cadre de Crisys, des méthodes de découplage et de changement de variables d'état ont aussi été étudiées [YED 00a, YED 00b], que nous ne rappellerons pas ici.

3.1. *Adjonctions de délais*

Reprenions l'exemple de la figure 1 et supposons que nous ayons par ailleurs les informations suivantes :

– les signaux x et y sont à *variabilité uniformément bornée (VUB)*, c'est-à-dire qu'ils ne varient pas trop vite ou, plus précisément, qu'il existe un temps minimum T_x (T_y) entre deux changement de valeur du signal,

– on connaît aussi une borne maximum τ_x (τ_y) de l'incertitude avec laquelle ces signaux sont perçus par le calculateur.

– $\sup(\tau_x, \tau_y) < \frac{\inf(T_x, T_y)}{2}$

La technique d'insertion de délai est alors la suivante :

– Dans le cas de la figure 1a, il n'y a rien à faire : le calculateur voit les deux signaux changer simultanément.

– Dans le cas de figure 1b, en revanche, le calculateur voit d'abord le changement de valeur de x. L'idée est alors d'attendre le délai τ_y ou le changement de y avant de prendre en compte le changement de x. Ainsi, les changements de valeurs de x et y seront pris en compte simultanément, sauf s'ils diffèrent de plus de l'incertitude temporelle associée à leurs variations. En effet, dans ce dernier cas, on peut être certain que l'on n'intervertit pas l'ordre des changements.

On voit ici que l'on a donc inséré un délai variable en fonction des valeurs d'entrées reçues, ce qui requiert une certaine intelligence, spécifique du logiciel. Nous avons montré en [CAS 00] que cette technique est assez puissante pour résoudre les problèmes d'aléas d'échantillonnage dans certains systèmes redondants.

3.2. *Adjonctions de causalités*

En revanche, ces techniques ne sont pas assez puissantes pour éviter les problèmes de courses critiques en général. Prenons par exemple le cas des systèmes redondants décrit en 2.3. Il est à peu près clair que, quelque soit le délai qu'on impose entre les changements de x et y pour les considérer comme non simultanés, lorsque les changements de x et y ont lieu aux environs de ce délai, certains calculateurs pourront considérer, du fait des incertitudes de l'échantillonnage, que ce délai est écoulé alors que d'autres considéreront le contraire et le système de vote divergera pareillement.

L'autre technique pour résoudre ce problème de course critique est d'interdire les courses, c'est-à-dire d'interdire les changements simultanés des deux signaux. Cela peut se faire au moyen d'insertions de causalités : si on sait que y ne peut changer que si on a précédemment positionné un autre signal z à une certaine valeur, si cette valeur n'est pas positionnée, le changement ne pourra pas se produire et aucune ambiguïté ne pourra résulter de l'échantillonnage.

Cette technique peut paraître très élémentaire mais elle est en fait très employée car elle s'apparente à celle des protocoles de communication : typiquement, dans un protocole, je n'attends une réponse que si j'ai posé une question et toute réponse qui me parviendrait sans que j'aie posé la question ne pourrait provenir que d'une erreur. Cette technique s'apparente aussi aux techniques de construction de circuits asynchrones à partir de « Signal Transition Graphs » [BRZ 95].

4. Echantillonnage des signaux continus

Dans cette section, nous partons d'une théorie naïve de l'échantillonnage des signaux continus pour la généraliser aux signaux discrets et hybrides.

D'après [AST 84], la théorie usuelle de l'échantillonnage de Nyquist-Shannon n'est pas exactement celle utilisée pour la commande temps-réel ; en effet, la reconstruction correspondante n'est pas causale. En réalité, les périodes d'échantillonnage considérées en temps-réel correspondent aux calculs d'erreurs des méthodes d'intégration à pas fixe utilisées et sont en général plus courtes que celles prévues par Nyquist-Shannon.

4.1. *Signaux et systèmes*

D'autre part, nous considérons des systèmes qui doivent fonctionner sur des horizons de temps très longs comme, par exemple, des centrales nucléaires ou des commandes d'avions devant voler pendant des heures. Ainsi, nous considérons des signaux sur un horizon non borné. Dans ces conditions un signal x est pour nous simplement une fonction de $\Re$ vers $\Re$ (fixée à une valeur de repos avant une date de début d'activité) et un système S une fonction transformant causalement des signaux en d'autres signaux, c'est-à-dire telle $S(x)(t)$ n'est fonction que des $x(t')$ avec $t' \leq t$.

4.2. *Echantillonnage*

Qu'est-ce qu'un signal échantillonnable ? C'est intuitivement un signal que l'on peut approximer par une fonction constante par morceaux avec une erreur d'approximation que l'on peut contrôler. Cela peut se reformuler ainsi : pour toute erreur ε positive, il existe un pas d'échantillonnage η positif, tel que deux échantillons quelconques $x(t)$, $x(t')$ voisins de moins de η en temps diffèrent de moins de ε en valeur.

Exprimé ainsi, on a reconnu la définition classique de la *continuité uniforme* (cf. figure 3) :

$$\forall \varepsilon > 0, \exists \eta > 0, \forall t, t', \mid t - t' \mid \leq \eta \Rightarrow \mid x(t) - x(t') \mid \leq \varepsilon$$

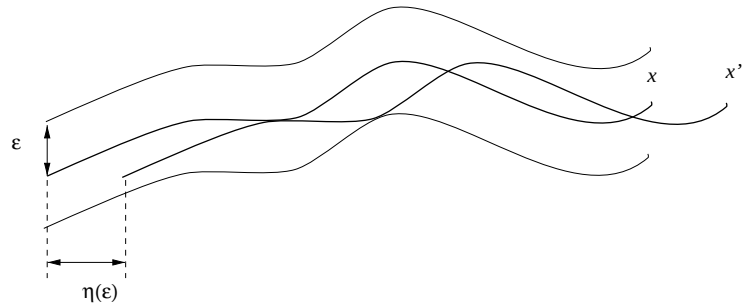


Figure 3. *Un signal uniformément continu*

On peut aussi dire qu'il existe une fonction des erreurs vers les délais η_x positive, associée au signal x, telle que :

$$\forall \varepsilon > 0, \forall t, t', |t - t'| \leq \eta_x(\varepsilon) \Rightarrow |x(t) - x(t')| \leq \varepsilon$$

4.3. *Redatation et échantillonnage*

Pour se placer dans un cadre plus fonctionnel, pour des raisons qui apparaîtront ultérieurement, nous introduisons ici des fonctions de redatation :

Une fonction de redatation r est une fonction non décroissante de $\Re$ dans $\Re$. Une telle fonction peut donc redater un signal par :

$$x' = x \circ r$$

Ce procédé très général permet, en particulier d'exprimer l'échantillonnage. Par exemple, un échantillonneur périodique de période T correspond à la fonction de redatation :

$$r(t) = E(t/T)$$

où E est la partie entière inférieure(voir figure 4). Ainsi, le signal échantillonné x' reste bloqué jusqu'au prochain saut de la fonction de redatation.

4.4. *Redatation et continuité uniforme*

Un autre intérêt de la redatation est de permettre d'exprimer la continuité uniforme de manière fonctionnelle.

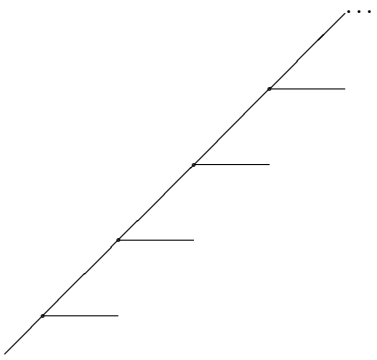


Figure 4. *Une redatation périodique*

Définition 4.1 (Signaux uniformément continus) *x est uniformément continu s'il existe une fonction positive η_x des erreurs vers les délais, telle que, pour tout $\varepsilon > 0$ et pour toute fonction de redatation r,*

$$||r - id||_\infty \leq \eta_x(\varepsilon) \Rightarrow ||x - x \circ r||_\infty \leq \varepsilon$$

où id est la fonction identité et où $||\ ||_\infty$ est la norme $\mathscr{L}_\infty$ (qui se ramène à la norme du sup pour les fonctions continues : $\sup_{t \in \mathscr{R}} |x(t)|$*).*

4.5. *Des signaux aux systèmes*

Cette approche s'étend très naturellement aux systèmes en disant qu'un système S est uniformément continu (cf. figure 5) s'il existe une fonction positive η_S des erreurs vers les erreurs telle que :

$$\forall \varepsilon > 0, \forall x, x', ||x - x'||_\infty \leq \eta_S(\varepsilon) \Rightarrow ||(S\,x) - (S\,x')||_\infty \leq \varepsilon$$

On peut alors proposer le théorème suivant [CAS 02] :

Théoreme 4.1 *Un système S, stationnaire, uniformément continu, alimenté par une entrée x uniformément continue, produit une sortie $S(x)$ uniformément continue avec :*

$$\eta_{Sx} = \eta_x \circ \eta_S$$

Ce théorème permet de propager un calcul d'erreur de type analyse numérique et de choisir des périodes d'échantillonnage adaptées aux précisions désirées. Il faut remarquer aussi que d'autres approches sont aussi possibles par exemple fondées sur la notion voisine de *stabilité entrée-sortie* [SAS 99]

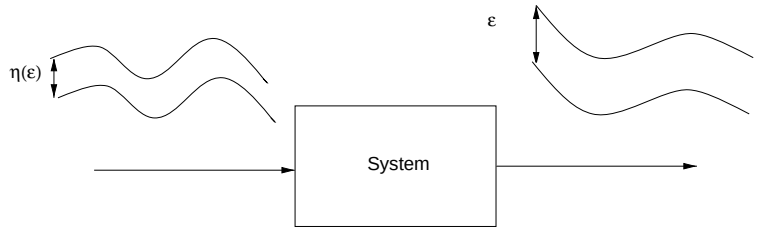


Figure 5. *Un système uniformément continu*

4.6. *Généralisation*

L'intérêt de l'approche fonctionnelle est de pouvoir se généraliser à n'importe quelle distance entre signaux. On dira ainsi qu'un signal x est uniformément continu pour la distance d s'il existe une fonction positive η_x telle que :

$$\forall \varepsilon > 0, \forall r, ||r - id||_\infty \leq \eta_x(\varepsilon) \Rightarrow d(x, x \circ r) \leq \varepsilon$$

De même, on dira qu'un système S est uniformément continu pour la distance d s'il existe une fonction positive η_S telle que :

$$\forall \varepsilon > 0, \forall x, x', d(x, x') \leq \eta_S(\varepsilon) \Rightarrow d((S\,x), (S\,x')) \leq \varepsilon$$

Et, dans ce contexte généralisé, le théorème 4.1 est encore vérifié.

5. Echantillonnage des signaux discrets

Nous considérons maintenant des signaux booléens et nous cherchons des concepts analogues à la continuité uniforme, permettant de caractériser l'échantillonnage de ces signaux. Naturellement, de même que nous nous étions limités précédemment aux signaux continus, nous devons maintenant limiter la classe des signaux que nous considérons, par exemple à la classe des signaux booléens continus par morceaux, c'est-à-dire tels qu'il existe une suite finie ou divergente d'instants $\{t_0, \ldots t_n, \ldots\}$ tels que le signal soit constant dans chaque intervalle ouvert $]t_n, t_{n+1}[$.

Dans ce contexte, nous pouvons essayer de formaliser la méthode d'adjonction de délais vue en 3.1.

Pour cela, nous introduisons la fonction $dc_{t_1,t_2}(x)$ qui compte le nombre de discontinuités du signal x dans un intervalle de temps $[t_1, t_2]$:

$$dc_{t_1,t_2}(x) = card\{t \mid x(t^-) \neq x(t^+) \wedge t_1 \leq t \leq t_2\}$$

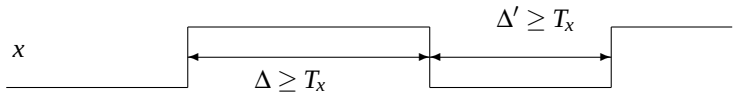


Figure 6. *Variabilité uniformément bornée*

où, comme d'habitude, $x(t^-), (x(t^+))$ est la limite à gauche (à droite) de x en t.

Cela nous permet de caractériser les signaux booléens échantillonnables, comme ceux (cf. figure 6) qui ne varient pas trop vite :

Définition 5.1 (Variabilité uniformément bornée (VUB)) *Un signal booléen x est à variabilité uniformément bornée s'il existe T_x positif, tel que*

$$\forall t, t', |t - t'| \leq T_x \Rightarrow dc_{t,t'}(x) \leq 1$$

5.1. *VUB et échantillonnage*

Nous avons vu précédemment comment les délais se transforment en erreurs. Qu'en est-il pour les signaux discrets ? On peut facilement proposer la relation suivante :

$$\forall x, r, ||r - id||_\infty \leq \eta \Rightarrow T_{x \circ r} \leq T_x - \eta$$

On voit ainsi comment la période de stabilité d'un signal se réduit sous l'effet de retards et d'échantillonnages incertains.

5.2. *VUB et adjonction de délais*

Considérons deux signaux VUB, x et y avec leurs périodes de stabilité T_x et T_y associées, et deux redatations bornées

$$\begin{aligned} ||r_x - id||_\infty &\leq \tau_x \\ ||r_y - id||_\infty &\leq \tau_y \end{aligned}$$

La question des adjonctions de délais est de produire une fonction, dite « de confirmation », f telle qu'il existe un retard borné global r vérifiant :

$$f(x \circ r_x, y \circ r_y) = (x \circ r, y \circ r)$$

Autrement dit, cette fonction vise à remplacer des délais incohérents en délais cohérents.

En [SAL 01], nous avons pu montrer que la fonction f_{τ_x,τ_y} définie par :

$$f_{tau_x,\tau_y}(x,y)(t) = x'(t), y'(t) = \begin{cases} x(t), y(t) & \text{si} \begin{cases} \forall t', t - \tau_y < t' \leq t \Rightarrow x(t) = x(t') \\ \forall t', t - \tau_x < t' \leq t \Rightarrow y(t) = y(t') \end{cases} \\ x(t), y(t) & \text{si} \begin{cases} \exists t', t - \tau_y < t' \leq t \wedge x(t) \neq x(t') \\ \exists t', t - \tau_x < t' \leq t \wedge y(t) \neq y(t') \end{cases} \\ x'(t^-), y'(t^-) & \text{autrement} \end{cases}$$

vérifie la propriété désirée sous réserve de la relation :

$$\sup(\tau_x, \tau_y) < \frac{\inf(T_x, T_y)}{2})$$

Cette propriété peut être vue comme un analogue du théorème 4.1 pour les signaux discrets en ce qu'il autorise un calcul de délais dans les réseaux acycliques d'opérateurs.

6. Echantillonnage des signaux hybrides : distance de Skorokhod

Malheureusement, le concept de variabilité bornée n'est pas topologique. Nous avons montré en [CAS 02] que la distance de Skorokhod pouvait en partie pallier cette difficulté.

6.1. *Distance de Skorokhod*

Cette distance [BIL 99] a été proposée comme généralisation de la distance usuelle $\mathscr{L}_\infty$ pour prendre en compte les discontinuités dans la convergence faible de lois de probabilités. Elle a été introduite dans les systèmes hybrides en [BRO 98].

Définition 6.1 (Distance de Skorokhod)

$$d_S(x,y) = \inf_{redatation\ bijective\ r} ||r - id||_\infty + ||x - y \circ r||_\infty$$

On voit ici l'idée de cette définition : au lieu de comparer des signaux aux mêmes instants, on s'autorise des glissements d'instants (redatations), pourvu que ceux-ci soient limités et qu'on ne rate aucun instant. C'est pourquoi on se restreint aux redatations bijectives.

6.2. *Distance de Skorokhod et variabilité uniformément bornée*

En [CAS 02], nous avons pu montré le théorème suivant, qui établit un lien entre les théories de l´échantillonnage de signaux continus et celles de l'échantillonnage des signaux discrets :

Théoreme 6.1 *Un signal booléen a une variabilité uniformément bornée si et seulement si il est uniformément continu pour la distance de Skorokhod.*

L'idée de la preuve est la suivante : si un signal a une variabilité uniformément bornée, toute redatation bornée, éventuellement discontinue, peut être compensée par une autre redatation bijective, donc continue. Dans l'autre sens, si un signal n'a pas de variabilité uniformément bornée, une redatation discontinue, même bornée peut effacer des points de discontinuité, qu'une redatation bijective, donc continue, ne pourra pas compenser.

6.3. *Distance de Skorokhod et systèmes hybrides*

D'autre part, il est immédiat de voir que la distance de Skorokhod domine la distance $||\ ||_\infty$:

$$\forall x, y, d_S(x,y) \leq ||x-y||_\infty$$

D'où,

Théoreme 6.2 *Un signal uniformément continu l'est aussi pour la distance de Skorokhod*

On voit donc que la distance de Skorokhod peut caractériser l'échantillonnage à la fois des signaux continus et des signaux discrets. Cela veut dire aussi qu'elle peut caractériser l'échantillonnage des signaux « hybrides », c'est-à-dire des signaux continus par morceaux.

7. Conclusion

Nous avons essayé de montrer ici que le problème de co-existence de composantes continues et discrètes dans les systèmes de commande pose des problèmes compliqués de réalisation. En particulier, l'échantillonnage périodique des systèmes à événements discrets pose des problèmes d'aléas et d'asynchronismes qui obligent les praticiens à recourir à des astuces « maison ». Nous avons ensuite essayé de montrer que la distance de Skorokhod était un bon candidat pour fonder une théorie unifiée de l'échantillonnage des systèmes continus, discrets et même hybrides.

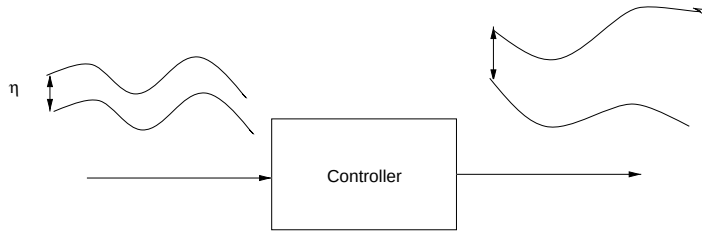


Figure 7. *Un système instable*

Cependant, beaucoup reste à faire en ce sens. En particulier, on peut noter les points suivants :

– **Systèmes multi-dimensionnels :** la question des systèmes discrets à plusieurs entrées n'est pas encore bien traitée par l'approche toplogique, malgré les indications fournies en [CAS 00].

– **Liens avec la stabilité :** même l'approche suivie en section 4.5 n'est pas totalement satisfaisante, en ce qu'elle ne s'applique qu'à des systèmes stables. Or, beaucoup de systèmes de commande, comme, par exemple les PID, ne sont pas stables en eux-mêmes(figure 7), mais ne le deviennent qu'en boucle fermée avec le procédé qu'ils sont sensés commander (figure 8). Ce problème va se poser sans doute de façon similaire dans notre approche, et pose la question de la caractérisation de la stabilité et de la stabilisation par feed-back dans le cas des systèmes hybrides. En particulier, il est tentant d'interpréter ([CAS 01a]) les techniques de protocoles et d'évitement de courses critiques en ce sens.

Mais, au delà de ces perspectives, on peut s'interroger sur la signification des constatations que nous faisons quant à l´état des communications entre les diverses communautés qui se partagent l'enseignement et la recherche en automatique continue, automatique discrète et informatique industrielle. Les problèmes que nous avons mentionnés ne sont pas nouveaux : par exemple les commandes de vol dites « électriques » sur calculateurs numériques des Airbus A320 ont été conçues dans les années 80. Cependant, il ne semble pas que ces problèmes aient reçu une attention suffisante de la part de ces communautés. Les conséquences n'en sont pas toujours heureuses :

– Les étudiants sortent des écoles avec un solide bagage dans chacune de ces trois disciplines ; mais, arrivés dans l'industrie, ils se retrouvent devant des situations imprévues, comme de devoir implanter des systèmes de commande mixtes, discrets et continus sur des calculateurs numériques et ils ne disposent d'aucune théorie satisfaisante pour ce faire. Les entreprises ont alors recours à des formations « maison » pour pallier cette situation.

– Les outils de conception, validation (simulation, mise au point) dont les concepteurs disposent ne permettent pas d'observer de façon précise et reproductible les phénomènes d'aléas d'échantillonnage qui risquent de perturber leurs conceptions.

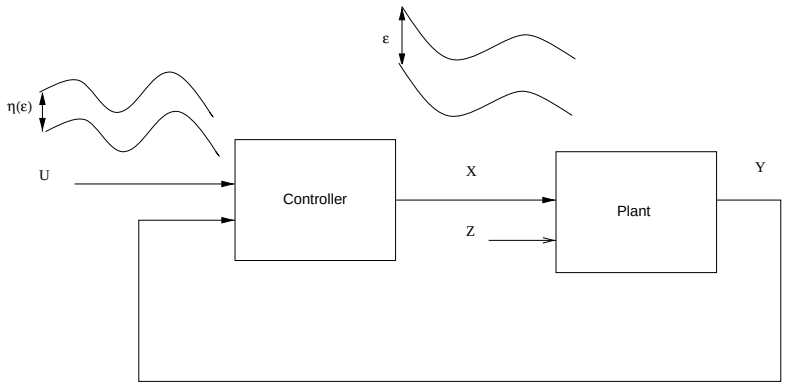


Figure 8. *Stabilisation par feed-back*

– La communication entre les divers intervenants dans la conception de systèmes complexes risque d'être obscurcie par l'absence de savoirs solides et partagés dans ce domaine. C'est le cas, par exemple, entre concepteurs et autorités de certifications pour les systèmes critiques. Soient ils ont partagé les mêmes formations, et la communication peut être biaisée, soit ils n'ont pas la même culture et certains problèmes peuvent être ignorés.

Il est vrai que l'apparition de la discipline dite des « systèmes hybrides » à la fin des années 90, a amené un certain progrès en la matière. Mais l'aspect implantation (informatique industrielle) y est encore peu représenté.

C'est pourquoi il paraît important, pour l'enseignement et la recherche, d'explorer cette mer inconnue qui se trouve au centre du triangle formé par ces trois disciplines.

Remerciements

Je voudrais remercier ici Albert Benveniste de l'INRIA-IRISA et Chiheb Kossentini d'Airbus-France pour leurs contributions à ce travail. Oded Maler de Verimag, Alberto San Giovanni-Vincentelli de l'université Berkeley, Jan van Schuppen de CWI et Hervé le Berre d'Airbus-France nous ont aidés de leurs conseils et encouragements.

8. Bibliographie

[AST 84] ASTRÖM K., WITTENMARK B., *Computer Controlled Systems*, Prentice-Hall, 1984.

[BEH 98] BEHM P., DESFORGES P., MEYNADIER J., « MÉTÉOR : An Industrial Success in Formal Development », BERT D., Ed., *B'98 : Recent Advances in the Development and Use of the B Method*, vol. 1393 de *Lecture Notes in Computer Science*, Springer, 1998.

[BER 88] BERGERAND J., PILAUD E., « SAGA ; a software development environment for dependability in automatic control », *SAFECOMP'88*, Pergamon Press, 1988.

[BIL 99] BILLINGSLEY P., *Convergence of probability measures*, John Wiley & Sons, 1999.

[BRI 94] BRIÈRE D., RIBOT D., PILAUD D., CAMUS J., « Methods and specification tools for Airbus on-board systems », *Avionics Conference and Exhibition*, London, December 1994, ERA Technology.

[BRO 98] BROUCKE M., « Regularity of solutions and homotopic equivalence for hybrid systems », *Proceedings of the 37th IEEE Conference on Decision and Control*, vol. 4, 1998, p. 4283–4288.

[BRZ 95] BRZOZOWSKI J. A., SEGER C.-J. H., *Asynchronous Circuits*, Springer-Verlag, 1995.

[CAS 99] CASPI P., MAZUET C., SALEM R., WEBER D., « Formal Design of Distributed Control Systems with Lustre », *Proc. Safecomp'99*, vol. 1698 de *Lecture Notes in Computer Science*, Springer Verlag, September 1999.

[CAS 00] CASPI P., SALEM R., « Threshold and Bounded-Delay Voting in Critical Control Systems », JOSEPH M., Ed., *Formal Techniques in Real-Time and Fault-Tolerant Systems*, vol. 1926 de *Lecture Notes in Computer Science*, September 2000, p. 68–81.

[CAS 01a] CASPI P., « Embedded control : from asynchrony to synchrony and back », HENZINGER T., KIRSCH C., Eds., *First International Wokshop on Embedded Software*, vol. 2211 de *Lecture Notes in Computer Science*, 2001.

[CAS 01b] CASPI P., MAZUET C., REYNAUD-PARIGOT N., « About the design of distributed control systems : the quasi-synchronous approach », *Proc. Safecomp'01*, vol. 2187 de *Lecture Notes in Computer Science*, Springer Verlag, September 2001.

[CAS 02] CASPI P., BENVENISTE A., « Toward an Approximation Theory for Computerised Control », SANGIOVANNI-VINCENTELLI A., SIFAKIS J., Eds., *2nd International Wokshop on Embedded Software, EMSOFT02*, vol. 2491 de *Lecture Notes in Computer Science*, 2002.

[KOP 97] KOPETZ H., *Real-Time Systems Design Principles for Distributed Embedded Applications*, Kluwer, 1997.

[PEA 80] PEASE M., SHOSTAK R., LAMPORT L., « Reaching Agreement in the Presence of Faults », *Journal of the ACM*, vol. 27, n° 2, 1980, p. 228–237.

[SAL 01] SALEM-HABERMEHL R., « Répartition de programmes synchrones temps-réel », Thèse de doctorat, Université Joseph Fourier, Grenoble, octobre 2001.

[SAS 99] SASTRY S., *Nonlinear systems - Analysis, stability, and control*, N° 10 Interdisciplinary Applied Mathematics, Springer-Verlag, 1999.

[YED 00a] YEDDES M., « *Contribution à une approche robuste pour la distribution de systèmes synchrones* », Thèse de doctorat, Institut National Polytechnique de Grenoble, novembre 2000.

[YED 00b] YEDDES M., ALLA H., DAVID R., « The Partition Method for the Order-insensitivity in a Synchronous Distributed Systems », *IEEE, ISCC'2000, Antibes (France)*, July 2000.