

Bonjour à tous. Aujourd'hui, nous allons reprendre les cours d'une manière un peu spéciale, comme vous devez le savoir, à cause de ce virus. J'espère que vous avez passé de bonnes vacances, et que la reprise va se faire dans de bonnes conditions. Aujourd'hui, on va commencer par quelques révisions. Comme vous le voyez, ces révisions et ces cours vont se faire en vidéo, et ces vidéos, je vais les découper en petits morceaux qui vont correspondre, pour aujourd'hui, à des chapitres du cours, et pour les semaines à venir, non pas en chapitres, mais en différentes sections des chapitres du cours, d'une part parce qu'elles seront plus faciles à manipuler par vous, et d'autre part parce que leur poids en sera diminué, et leur taille aussi.

Aujourd'hui, on va commencer par des révisions, et on va commencer par les chapitres du départ, c'est-à-dire les chapitres d'introduction qui étaient les chapitres 1 et 2 de votre livre. Ce qu'on avait vu dans les chapitres 1 et 2, principalement, c'est qu'il y avait deux méthodes qui permettent de contrôler (de commander) un système, et ces deux méthodes, on avait vu que la première, c'est ce qu'on appelle la commande en chaîne directe. Et cette commande en chaîne directe, elle consiste en quoi? Alors déjà, on va revenir sur un point important : ce point important, c'est qu'est-ce que c'est qu'une commande? Qu'est-ce qu'on entend par commander un système, contrôler un système? Alors pour ça, on a un processus : ce processus, c'est la partie du système qui va modifier la matière d'œuvre. Ça veut dire que dans tout le système, la partie qui est la plus visible, entre guillemets, c'est le processus, puisque c'est lui qui agit. Ce processus, il a une sortie, et une entrée. Mais cette sortie, on veut qu'elle vaille une certaine valeur, ou qu'elle atteigne certaines performances. Donc ça veut dire : j'ai une sortie que je souhaite obtenir, une sortie souhaitée. Et pour l'obtenir, la question, c'est : qu'est-ce que je vais mettre en entrée du processus pour obtenir la sortie souhaitée? Et si je veux obtenir la sortie que je souhaite, eh bien je vais justement commander ce système pour obtenir la sortie souhaitée à la sortie du système.

Et donc pour commander ce système en chaîne directe, eh bien avant ce processus, qu'est-ce qu'on va trouver? Eh bien on va trouver les différents composants qu'on avait étudiés dans le premier livre. Ces différents composants, ce sont le pré-actionneur, l'actionneur, le transmetteur de puissance, et l'effecteur. Et donc l'ensemble de ces composants va permettre de commander le système. Mais ce qu'on avait vu, c'est que si je commande le système de cette manière-là, en chaîne directe, il y a un problème. Ce problème, c'est que je sais ce que je mets en entrée, mais je n'ai aucune garantie que ce que je vais obtenir en sortie est effectivement ce que je souhaite. Parce qu'il n'y a aucun moment où je vais venir observer ce qui se passe en sortie ici. Donc cette méthode de commande-là, on peut l'améliorer avec une deuxième méthode qu'on appelle la commande en boucle fermée.

Cette commande en boucle fermée, elle consiste en quoi? Eh bien elle consiste à dire : j'ai mon processus, ce processus a une sortie que je souhaite pouvoir contrôler, et pour commander ce système, qu'est-ce que je vais faire? Je vais rajouter, comme pour la commande en chaîne directe, un pré-actionneur, un actionneur, un transmetteur de puissance, un effecteur, et je vais rajouter quelques éléments en plus. Le premier élément, je vais le rajouter à la sortie : je vais venir regarder avec mes yeux ce qui se passe en sortie du système. Pour ça, je vais mesurer ce que j'ai en sortie avec un capteur. Ensuite, alors ça, je vous l'avais déjà dit, si ce que je mesure, par exemple, c'est un signal sinusoïdal qui a cette forme-là, en pratique, ce qu'on observe, c'est que le capteur ne va pas me

fournir ça : le capteur va me fournir un signal qui ressemble à ça. Pourquoi ? Parce que le capteur est dans un environnement, il interagit avec son environnement, et ça veut dire qu'il est perturbé par son environnement. Donc j'ai des perturbations ici au niveau du capteur, de même que je vais avoir des perturbations sur le processus, des perturbations sur l'effecteur, sur le transmetteur de puissance, sur l'actionneur, et sur le pré-actionneur. Donc tous les composants du système sont perturbés à cause de leur interaction avec l'environnement. Bon, et donc le problème qui se pose, c'est que j'aimerais bien récupérer le signal qui est là, j'aimerais avoir ce qui se passe en sortie. Donc pour faire ça, qu'est-ce qu'on avait dit ? On avait dit qu'on rajoute ici ce qu'on appelle un conditionneur, et le but de ce conditionneur, c'est de me restituer le même type de signal que ce que j'avais en entrée du capteur. Et pour obtenir ça, il me suffit de prendre le signal fourni par le capteur, celui-là, et d'en filtrer le bruit, c'est-à-dire d'en filtrer les hautes fréquences qui se trouvent ici. Donc ce conditionneur, ce n'est rien d'autre qu'un filtre passe-bas qui va laisser passer les basses fréquences et qui va filtrer les hautes fréquences. Bon, une fois que j'ai ça, ce qu'on va chercher à faire, c'est comparer ce que j'ai en sortie, ce que j'ai ici, c'est une image de la sortie, et je veux comparer cette image de la sortie à ce que j'impose en consigne. Donc ici, je vais avoir ma consigne, qu'en général on note $x_c(t)$ alors que la sortie était notée $x(t)$. Bon, et donc je veux comparer cette consigne à ce que j'ai en sortie. Et là, on a encore un problème, c'est que la consigne, *a priori*, ça peut être n'importe quoi : ça peut être une température, une tension, une intensité, un déplacement, une vitesse, n'importe quoi, alors que ce que vous allez avoir ici, l'image de la sortie, a été fournie par un capteur après avoir été filtrée. Ça veut dire que ce qu'on a ici, c'est une tension, c'est-à-dire des volts. Et je n'ai aucun moyen de comparer des volts avec des mètres, des ampères, ou une autre unité. Donc si je veux pouvoir comparer l'image de la sortie à ma consigne, eh bien il faut que je commence par convertir ma consigne en quelque chose que je peux comparer à l'image de la sortie, et pour ça on utilise un adaptateur, et cet adaptateur va me fournir une image de la consigne, en volts, que je pourrai comparer à l'image de la sortie. Ensuite je compare les deux, j'obtiens en sortie ce qu'on a noté $\varepsilon(t)$, qu'on appelle le signal d'erreur, et donc $\varepsilon(t)$ est la différence entre $u_a(t)$, qui est l'image de la consigne, et $u_r(t)$, qui est l'image de la sortie. Et enfin, dernier composant qu'on va rajouter, auquel on ne s'est pas intéressé pour le moment, avant les vacances, mais qu'on verra au chapitre 9, c'est le correcteur : le correcteur va nous permettre d'améliorer les performances du système si elles ne sont pas suffisamment bonnes.

Donc voilà pour le chapitre 2. On va passer ensuite au chapitre 3.