Moyenne Mobile et moyenne Glissante

Préambule :

Dans la presse courante, on nous présente de plus en plus de graphique avec un petit astérisque « moyenne glissante ». Cet article essaie de clarifier ce concept et aborde l'utilisation de ces moyennes dans le cas du traitement de signal. Il est en effet primordial d'avoir des outils numériques qui agissent sur nos signaux à l'identique de nos vieux réseaux de capacité et résistances.

Il est possible de manipuler le concept à l'aide du fichier Excel joint qui permet d'effectuer des simulations de comportement sur une structure de donnée.

Plan de cet article :

I Moyenne glissante sur p valeurs…
II Moyenne mobile exponentielle
III Comparaison des deux moyennes.
IV Réalisation de filtre numérique
 Filtrage au premier ordre
 Filtrage au deuxième ordre
 Généralisation
VI Codage en C++ des moyennes
VII Utilisation de ces fonctions moyennes hors domaine de l'électronique et du traitement du signal.

I Moyenne glissante sur p valeurs…

Par définition nous avons pour une fonction la définition de la moyenne glissante :

$$Mg_x = \frac{1}{\delta x} \int_{x-\delta x}^{x} f(x)\,dx$$

Pour un système discret de valeurs échantillonnées, pour une moyenne glissante sur p valeurs, la formule ci-dessus revient à

$$Mg_n = \frac{1}{p}\left(\sum_{j=n-p+1}^{j=n} U_j\right) = \frac{1}{p}\left(p.Mg_{n-1} - U_{n-p} + U_n\right) \text{ avec } Mg_p = \frac{1}{p}\left(\sum_{j=1}^{j=p} U_j\right)$$

En utilisant la formule « récurrente » $Mg_n = \frac{1}{p}\left(p.Mg_{n-1} - U_{n-p} + U_n\right)$ le calcul de cette moyenne nécessite au minimum les opérations suivantes : 1 multiplication, 1 division, 2 addition/soustraction.

Il convient d'ajouter toutes les opérations de décalage de la pile de valeur stockées et la mise en mémoire de la nouvelle valeur .

Inconvénient :

- Cette moyenne nécessite le stockage en mémoire des p valeurs antérieures ce qui sur un microcontrôleur basique rempli très rapidement la mémoire RAM disponible.
- Du point de vue des valeurs prises en compte, La méthode est « brutale ». On passe de 100% de la valeur prise en compte pour un échantillon à 0% en un pas de calcul !!

II Moyenne mobile exponentielle

Contrairement à la méthode précédente, cette moyenne ne nécessite pas le stockage de valeurs antérieures et présente une plus grande « douceur » de traitement des valeurs.

Première formulation qui rappelle celle de la moyenne glissante sur p valeurs :

$M_0 = \frac{1}{p}\left(\left(p-1\right)M_{-1} + V_0\right)$ avec $M_{-1} = V_0$ nous obtenons $M_0 = V_0$

Entamons la récurrence :

$$M_1 = \frac{1}{p}\left(\left(p-1\right)M_0 + V_1\right)$$

$$M_2 = \frac{1}{p}\left(\left(p-1\right)M_1 + V_2\right)$$

$$M_3 = \frac{1}{p}\left(\left(p-1\right)M_2 + V_3\right)$$

…

$$M_n = \frac{1}{p}\left(\left(p-1\right)M_{n-1} + V_n\right)$$

Le calcul de la moyenne se déduit de la précédente par la somme d'opérations suivantes : 1multiplication, 1 division, 1 addition

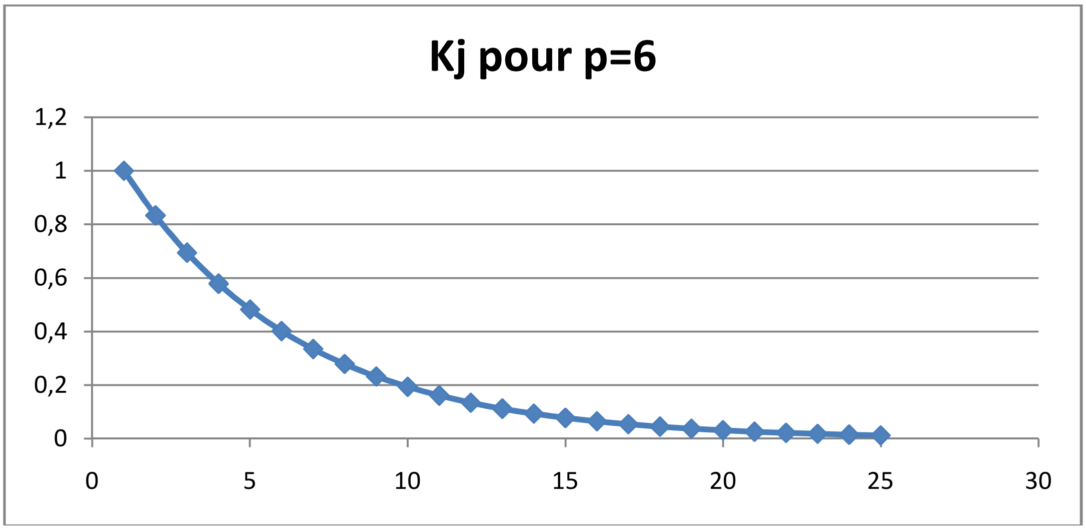
Nous pouvons montrer que $M_3 = \frac{1}{p}\left(\left(p-1\right)\frac{1}{p}\left(\left(p-1\right)\frac{1}{p}\left(\left(p-1\right)V_0 + V_1\right)_1 + V_2\right) + V_3\right)$ soit

$M_3 = \frac{1}{p}V_3 + \frac{\left(p-1\right)}{p^2}V_2 + \frac{\left(p-1\right)^2}{p^3}V_1 + \frac{\left(p-1\right)^3}{p^4}V_0$ Nous avons donc la formule générale :

$M_n = \frac{1}{p}\left(\sum_{j=0}^{j=n}\frac{\left(p-1\right)^j}{p^j}V_{n-j}\right) = \frac{1}{p}\left(\sum_{j=0}^{j=n}k_jV_{n-j}\right)$ avec $k_j = \frac{\left(p-1\right)^j}{p^j} = \left(\frac{p-1}{p}\right)^j$

Exemple de coefficients k_j avec p=6 :

On remarque que les termes éloignés de plus de 14 valeurs de la valeur actuelle ne comptent plus que pour moins du 1/10 é me de leurs valeur. On remarque aussi la « douceur » de l'évolution de l'évincement des valeurs en fonction de leur éloignement de la valeur courante.



Pour étudier plus en avant cette « moyenne », nous allons lui préférer une autre formulation : $M_n = (1-\alpha) M_{n-1} + \alpha . V_n$ avec $\alpha \in]0,1[$ le degré de décroissance. En posant $\alpha = \frac{1}{p}$ nous pouvons montrer que les deux formulations sont identiques.

Il est possible de montrer que cette moyenne correspond à un filtre passe bas du premier ordre de « pulsation *de coupure* » en Radiant par échantillons de

$$\omega_c = -\ln(1-\alpha) = -\ln\left(1-\frac{1}{p}\right) = -\ln\left(\frac{p-1}{p}\right)$$

Exemple : Pour un signal échantillonné à 1kHz et un $\alpha = 0.25$; $p = 4$ nous obtenons une pulsation de coupure $\omega_c = -\ln(1-0.25) = -\ln(0.75) = 0.287\, rd / \partial t$ avec $\partial t = 1E^{-3} s$ nous avons $\omega_c = 280\, rd / s$ soit la fréquence de coupure $f_c = \frac{\omega_c}{2\pi} = 45\, Hz$

Les calculs précédents reviennent à déterminer la formule $f_c = \frac{-\ln(1-\alpha)}{2\pi} f_{echantillonage}$ Le tableau ci-dessous donne les fréquences de coupure en Hz dans une série de cas cohérent avec l'utilisation d'un microcontrôleur réalisant une prise de donnée synchrone.

	α	0,5	0,25	0,125	0,0625	0,03125	0,015625
	p	2	4	8	16	32	64
Fréquence d'échantillonnage en Hz	5	0,55	0,23	0,11	0,05	0,03	0,01
	10	1,10	0,46	0,21	0,10	0,05	0,03
	50	5,52	2,29	1,06	0,51	0,25	0,13
	100	11,03	4,58	2,13	1,03	0,51	0,25
	500	55,16	22,89	10,63	5,14	2,53	1,25
	1000	110,32	45,79	21,25	10,27	5,05	2,51
	5000	551,59	228,93	106,26	51,36	25,26	12,53
	10000	1103,18	457,86	212,52	102,72	50,53	25,06

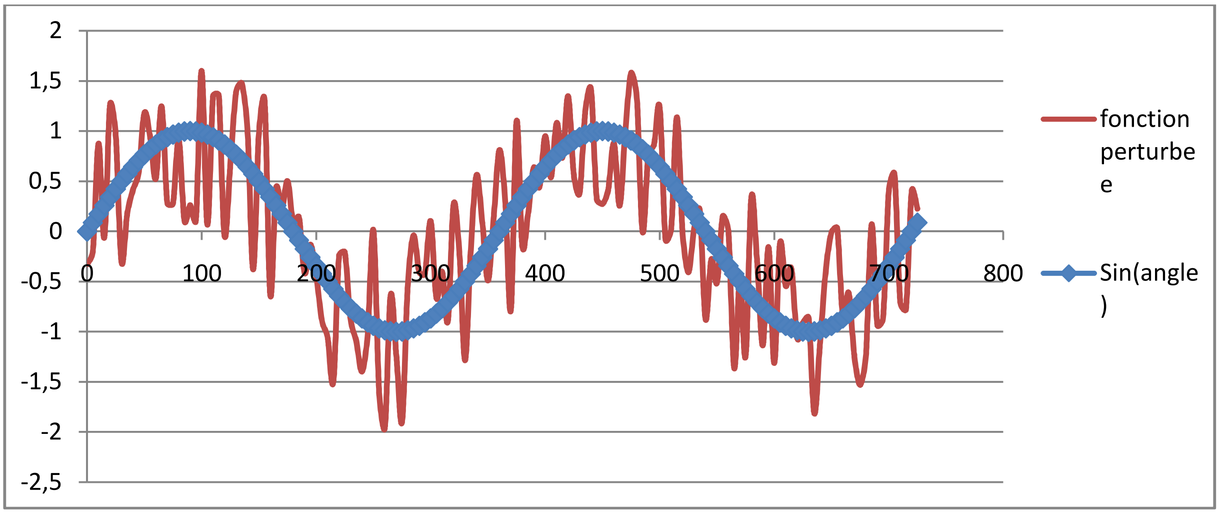
III Comparaison des deux moyennes.

A l'aide d'Excel nous allons créer une fonction échantillonnée volontairement très bruitée :

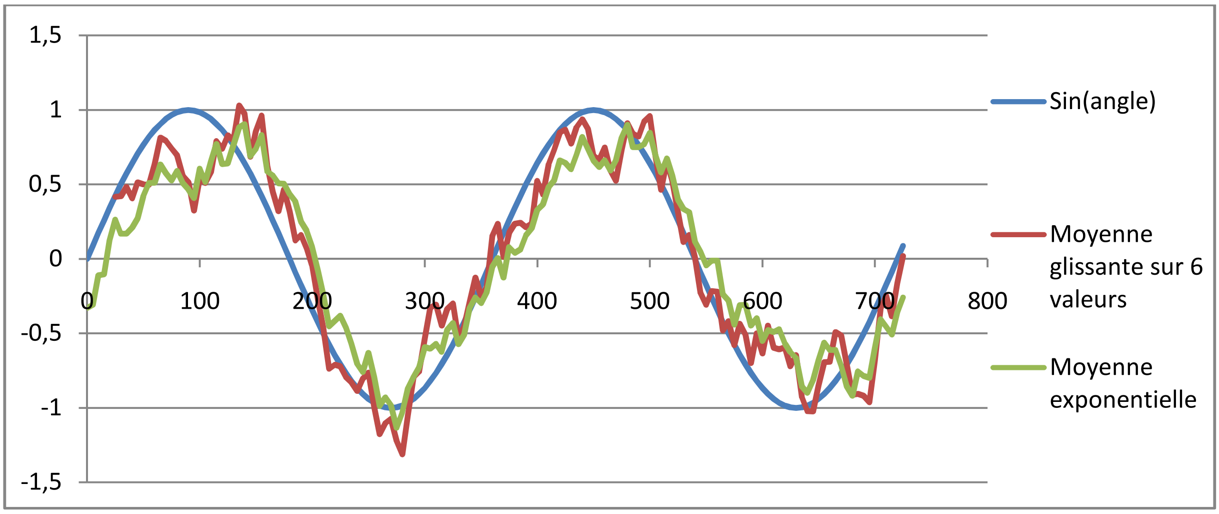
$V_n = \sin\left(\frac{2n.\pi}{pas}\right) + \beta.alea(-1,1)$. Ou alea est une fonction qui retourne une valeur aléatoire réelle entre les deux valeurs données. Le coefficient $\frac{1}{\beta}$ est représentatif du rapport signal bruit.

Un exemple effectué à l'aide d'Excel (ficher ***filtre.xlsx*** onglet Fonction joint) permet de montrer la puissance de cette formulation.

On a généré une fonction sinus auquel on a ajouté une perturbation aléatoire dans un rapport signal bruit de 1.



La figure suivante contient les résultats des calculs des moyennes sur la fonction perturbée ci-dessus



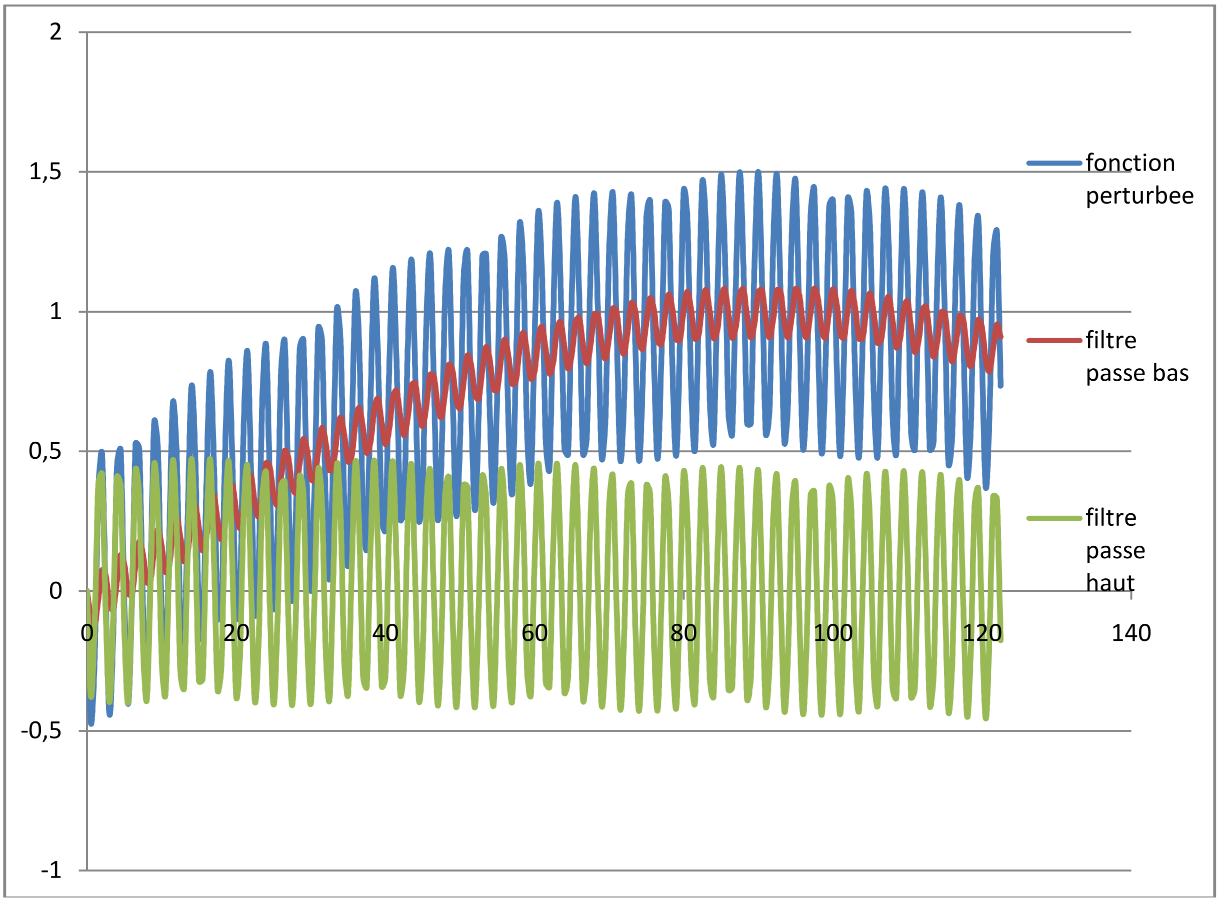
On constate que :

- Hors mis les problèmes de départ (dues aux calculs des premiers pas), la moyenne effectue bien un filtre passe pas du signal perturbé.

- A l'identique d'un filtre analogique passe bas réalisé à l'aide d'une capacité et d'une résistance en parallèle, le filtre numérique crée un déphasage retard par rapport à la fonction initiale
- A condition de ne pas avoir un rapport signal bruit trop petit (on peu le simuler sur le fichier Excel en modifiant le coefficient Beta) on conserve à peu près l'amplitude du signal initial.
- La moyenne glissante exponentielle donne des résultats analogues à la moyenne glissante sur fenêtres fixe. Elle nécessite du point de vue mémoire moins de ressources de stockage en RAM et moins de calculs (au moins une opération en virgule flottante de moins et nous verrons plus loin durant le codage que ces deux moyennes ne se ressemblent pas en terme de complexité informatique). Ces deux dernières caractéristiques sont relativement intéressantes dans le cas de traitement temps réel de signal.

IV Réalisation de filtre numérique

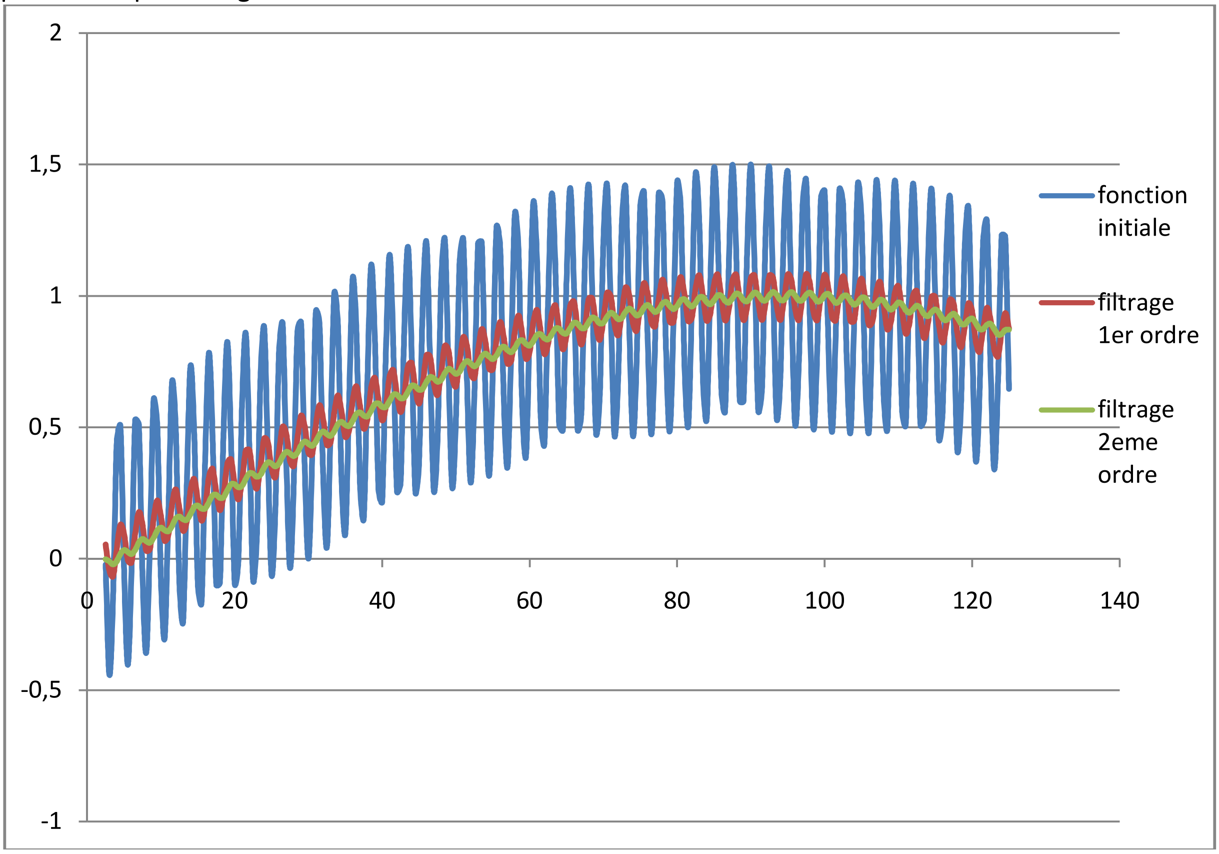
Filtrage du premier ordre :
Sur le même fichier Excel, sur l'onglet filtre, on a généré une fonction sinus perturbée par une harmonique (courbe bleu).
Un filtre numérique passe bas du premier ordre réalisé à l'aide e la moyenne exponentielle (courbe rouge).
Un filtre passe haut calculée par la différence entre la valeur filtrée passe bas et la valeur de la fonction donne la courbe verte.



En jouant sur le paramètre p du fichier on modifie la fréquence de coupure du filtre. Ceci se matérialise par un degré d'ondulation résiduel de la courbe rouge. (Essayez sur le fichier Excel !)

Filtrage du deuxième ordre :

Un filtre du deuxième ordre sur un signal revient à effectuer une deuxième fois une procédure de filtrage du premier ordre avec un même filtre que le premier (même caractéristiques). Il est facile de le matérialiser sur le fichier Excel joint et de jouer sur les « fréquences de coupures » en jouant sur le paramètre p de l'onglet filtre.



Généralisation :

Dans le cadre de l'informatique « embaquée » ou de gestion de procès, nous recevons une image des phénomènes physiques via des capteurs et chaine d'acquisition qui convertissent la nature du phénomène en une quantité numérique représentative. Cette quantité est perturbée par divers phénomènes physiques extérieurs ainsi que par la chaine d'acquisition elle-même. Il convient donc souvent de filtrer le signal avant de l'exploiter pour le « dépoussiérer » de ses perturbations.

En règle général, les perturbations sont de nature ondulatoires haute fréquences par rapport au signal à traiter et un filtre passe bas permet habituellement de récupérer une image exploitable du phénomène.

De manière numérique, la réalisation d'un filtre passe bas (et comme montré d'un filtre passe haut) est relativement simple en utilisant la moyenne exponentielle. La réalisation d'un filtre d'ordre supérieur ou à courbes complexes est aussi très simple.

Attention le « numérique » même s'il semble LA solution, peut provoquer mathématiquement des réactions non maitrisée. Il est bon de rappeler que le phénomène d'échantillonnage provoque une perturbation harmonique de la fréquence ½ de la valeur de l'échantillonnage et que les erreurs numériques dues à la résolution du convertisseur analogique numérique provoquent un bruit de fond relativement aléatoire. Ces phénomènes peuvent perturber sérieusement les mesures et traitements. On ne parlera pas des phénomènes de saturation numériques qui s'ils ne sont pas maitrisés dans les algorithmes peuvent provoquer des « retournement de situations » non désirés (ex : pour un mot de 8 bits, 255+1 donne 0 !!! et 0-1 peut donner 255 …) Il me semble que le vol 501 d'Ariane 5 pourtant un élément de haute technologie a subi une erreur de ce type

Dixit wikipédia :

Après 37 secondes de vol, les fortes accélérations produites par l'évolution de la fusée provoquent un dépassement d'entiers dans le calculateur du système de guidage inertiel principal, qui se met aussitôt hors service. Le système de guidage de secours, identique au système principal, subit la même avarie et s'arrête à la même seconde. Le pilote automatique, qui s'appuie justement sur les informations provenant de ces systèmes de guidage inertiels, n'a alors plus aucun moyen de contrôler la fusée. L'échec de la mission est inéluctable.

. Le calculateur de la centrale inertielle est équipé de protections, qui évitent qu'une erreur informatique se produise en cas de mesure trop élevée (plantage par dépassement de capacité). Pour certaines valeurs, il est physiquement impossible d'atteindre la limite, ou alors il existe une large marge de sécurité, et il a donc été décidé de ne pas mettre de protection, les concepteurs estimant que ces emplacements mémoire ne pourraient jamais être saturés par une valeur trop grande. C'est pourtant le dépassement d'une valeur non protégée qui a provoqué l'incident

VI Codage en C++ des moyennes

L'algorithme correspondant à ces moyennes peut s'écrire comme suit

\\ moyenne fenetre
Variables : moyenneCourante, table[tailleTable]
Parametres entree : valeurCourante, init

DÉBUT
   SI {init==true} ***ALORS***
      POUR {compteur} = {0} ***JUSQU'À*** {taille table-1} ***INCRÉMENT*** {1} ***FAIRE***
      {table[compteur]=valeurCourante}
      FINPOUR
      {moyennecourante=valeurCourante}
   SINON
      {moyenneCourante=0}
      POUR {compteur} = {0} JUSQU'À {tailleTable - 2} INCRÉMENT {1} FAIRE
      {moyenneCourante=moyenneCourante=table[compteur]}
      FINPOUR
      {table[tailleTable-1]=valeurCourante}
      {moyenneCourante=(moyenneCourante+valeurCourante)/tailleTable}
   FINSI
FIN

\\ moyenne exponentielle
Variables : moyenneCourante]
Parametres entree : valeurCourante, init

DÉBUT
   SI {init==true} ***ALORS***

{MoyenneCourante=valeurCourante}
SINON
{moyenneCourante=alpha*moyenneCourante+(1-alpha)*valeurCourante}
FINSI
FIN

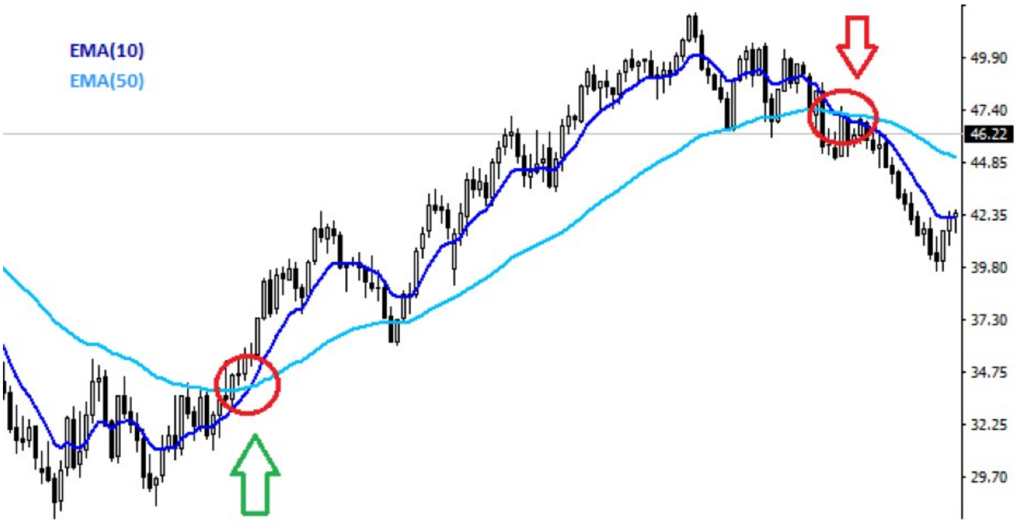
Un exemple de codage possible en C++ de ces deux moyennes

```
// Moyenne glissante « fenetre »
#define tailleFenetre 10
float moyenneGlissante(flat valeur, boolean init=false)
{
static float moyenneSto[tailleFenetre] ; // traité qu'au  premier appel de la fonction
static float valeurMoyenne=0 ;

if (init)
        {
        for(int i=0 ; i<tailleFenetre ; i++) {moyenneSto[i]=valeur ;
        valeurMoyenne=0 valeur
        return valeurMoyenne ;
        }
valeurMoyenne=1/tailleFenetre*(tailleFenetre*valeurMoyenne-moyenneSto[0]+valeur) ;
// decalage de la pile de valeur vers le bas
for (int i=1 ; i<length(moyenneSto) ; i++) {moyenneSto[i-1]=moyenneSto[i] ;}
// empilage de la valeur
moyenneSto[tailleFenetre-1]=valeur ;

return valeurMoyenne ;
}

 // ****************************************************************
// Moyenne Exponentielle
const float coefAlpha=0.2 ;
float unMoinsAlpha=1-coefAlpha ;
float moyenneExpo(float valeur,boolean init=false)

{
static float valeurMoyenne=0 ; // traité qu'au  premier appel de la fonction
if (init)
        {
        valeurMoyenne=valeur ;
        return valeurMoyenne ;
        }
valeurMoyenne=unMoinsAlpha*valeurMoyenne+coefAlpha*valeur ;
return valeurMoyenne ;
}
```

Nota : On peut effectivement essayer de simplifier le code de la moyenne « fenêtre » en utilisant des ruses et astuces, mais le résultat est tout de même sans appel… L'efficacité en nombre d'instruction de la moyenne exponentielle n'est pas à démontrer.

VII Utilisation de ces fonctions moyennes hors domaine de l'électronique et du traitement du signal.

Curieusement (peut être pas !) les fonctions de moyennes glissantes sont très utilisées dans le domaine de la finance et surtout par les « traders » qui comparent en effectuant des filtres numériques à fréquences différentes les évolutions à court et long terme de la valeur de tel ou tel action. Les positions relatives de ces deux courbes indiquent s'il est judicieux de vendre ou d'acheter les titres (pour faire du bénéfice bien évidement). En effet lorsque la cote à court terme passe au dessus de la moyenne de la cote à long terme, il est intéressant de vendre pour récupérer la surcote. Dans le cas contraire, on peut estimer que l'achat de titre peut à long terme s'avérer payant en différentiel de cote. Cette science non exacte se base sur les fréquences de coupure estimées qui suivant les opérations boursières peuvent aller de 5 minutes à plusieurs jours, semaines...



Il est à noter que le logiciel de simulation numérique *« MatLab »* outre ses outils utilisés par les ingénieurs non financiers traitant de problèmes techniques matériels, possède une myriade de fonctions financières d'analyse de tableau de variation. C'est un outil de premier rang dans le domaine de la finance… Eh oui les unités $, €, … sont certainement les plus utilisés au monde loin devant le système Métrique ou Impérial…

Cordier Yves
DDFPT
Lycée Altitude
Briançon