UVSQ ISTY Institut des Sciences et Techniques des Yvelines université PARIS-SACLAY CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES	***TRAITEMENT DU SIGNAL*** Partie 3 : La transformée en z

La transformée en z est un outil pour la représentation et l'étude des signaux discrétisés. Ces signaux sont donc échantillonnés (partie 2) puis discrétisés (partie 2).

1. Introduction :

La transformée en Z est un outil fondamental pour l'analyse des **signaux et systèmes discrets**. Elle constitue l'équivalent discret de la transformée de Laplace.
Elle est utilisée notamment en :

- traitement du signal numérique (DSP)
- robotique
- systèmes embarqués (Arduino, ESP32)
- automatique numérique

2. Passage du continu au discret :

Un signal continu s(t) échantillonné donne : **s[k] = s(kT_e)**
avec :

- **T_e** : période d'échantillonnage
- **k** : indice entier

On obtient une **suite discrète** :

{s[0], s[1], s[2], ...}

3. Définition de la transformée en Z :

La transformée en Z d'un signal discret s[k] est définie par :

$$Z\{s[k]\}=S(z)=\sum_{k=0}^{+\infty} s[k].z^{-k}$$

4. Lien fondamental avec Laplace

La variable complexe z est reliée à p (Laplace) par : $z=e^{pT_e}$
Interprétation :

- Laplace → domaine continu
- Z → domaine discret

Correspondance :

- Re(p) < 0 ⇔ |z| < 1 (stabilité)

5. Plan complexe en z

Dans le plan complexe :

- |z| = 1 → régime sinusoïdal
- |z| < 1 → décroissance
- |z| > 1 → divergence

Le cercle unité joue un rôle central.

6. Région de convergence

La transformée en Z n'existe que si la série converge.
Exemple :

S(z) = ∑ $z^{\{-k\}}$ converge si |z| > 1

7. Transformées usuelles

Échelon :

u[k] = 1 → $S(z)=\frac{1}{(1-z^{-1})}$

Exponentielle :

$s[k]=a^k \rightarrow S(z)=\frac{1}{(1-a.z^{-1})}$

8. Propriétés

- Linéarité
- Retard : $Z\{x[k-1]\}=z^{-1}X(z)$
- Avance : $Z\{x[k+1]\}=z.X(z)$
- Convolution → produit

9. Systèmes discrets

Un système discret est décrit par une **équation aux différences** :
Exemple :

$y[k]=a.y[k-1]+b.x[k]$

Transformée en Z :

$Y(z)=a.z^{-1}.Y(z)+b.X(z)$

Fonction de transfert :

$H(z)=Y(z)/X(z)=\frac{b}{(1-a.z^{-1})}$

10. Stabilité

Un système est stable si :
Tous les pôles de H(z) sont à l'intérieur du cercle unité : |z| < 1

11. Interprétation fréquentielle

Sur le cercle unité :
$z=e^{j\omega T_e}$

Permet d'obtenir la réponse fréquentielle.

12. Exemples

Filtre passe-bas : $y[k]=0.8\,y[k-1]+0.2\,x[k]$

Moyenne glissante : $y[k]=(1/N)\sum x[k-i]$

Exemple de filtre sous Arduino:

```
float y = 0; float alpha = 0.8;
```

```
void loop() {
      float x = analogRead(A0);
      y = alpha*y + (1-alpha)*x;
      }
```

13. Conclusion

La transformée en Z est l'outil central pour :

- analyser les systèmes numériques
- concevoir des filtres
- comprendre la stabilité

Elle est l'équivalent discret de la transformée de Laplace.

Quelques exemples :

Exemple 1 : Calculez la transformée en z d'un échelon : s(t)=1

$$F(z)=\sum_{k=-0}^{+\infty} s(k.T_e).z^{-k}=\sum_{k=-0}^{+\infty} z^{-k}=1+z^{-1}+z^{-2}+z^{-3}+....$$

On est devant une suite géométrique de raison z^{-1} donc : $F(z)=\frac{1}{1-z^{-1}}$

Exemple 2 : Calculez la transformée en z d'un signal: $s(t)=e^{-at}$:

$F(z)=\sum_{k=-0}^{+\infty} s(k.T_e).z^{-k}$ et $s(k.t_e)=e^{-a.k.T_e}=(e^{-a.T_e})^k=A^k$avec $A=e^{-a.Te}$=cst

donc $F(z)=\sum_{k=-0}^{+\infty} s(k.T_e).z^{-k}=\sum_{k=-0}^{+\infty} A^k.z^{-k}=\sum_{k=-0}^{+\infty}\left(\frac{A}{z}\right)^k=1+\frac{A}{z}+\left(\frac{A}{z}\right)^2+....$

On est devant une suite géométrique de raison $\frac{A}{z}$ donc : $F(z)=\frac{1}{1-\frac{A}{z}}=\frac{1}{1-A.z^{-1}}=\frac{1}{1-z^{-1}.e^{-a.T_e}}$

Propriétés de la transformée en z :

Linéarité	$Z[a.x_k+b.y_k]=a.X(z)+b.Y(z)$
Avance	$Z(x_{k+1})=z.X(z)$
Retard	$Z(x_{k-1})=z^{-1}.X(z)$
Somme	$\sum_{k=0}^{+\infty} x_k=\lim_{z\to 1} X(z)$
Valeur finale	$x_{+\infty}=\lim_{z\to 1}(1-z^{-1})X(z)$
Valeur initiale	$x_0=\lim_{z\to+\infty} X(z)$
Convolution soit $\{c_m\}=\{x_n * y_n\}$ $c_n=\sum_{m=-\infty}^{+\infty} x_m.y_{n-m}$	$Z[c]=X(z).Y(z)$

Calcul de la transformée en z :

Il existe 3 manières de calculer cette transformée en z

1. Par la formule de définition : $F(z)=\sum_{k=-0}^{+\infty} f(k.T_e).z^{-k}$
2. Par la théorie des résidus (non abordé dans ce cours)
3. Par l'utilisation du formulaire des transformées usuelles dans l'espace z comme ci-dessous :

$X(p)$	$X(t)$	$X(z)$
1	$\delta(t)$	1
$e^{-kT_e p}$	$\delta(t-kT_e)$	z^{-k}
$\frac{1}{p}$	$\Gamma(t)=1$	$\frac{z}{z-1}$
$\frac{1}{p^2}$	t	$\frac{T_e z}{(z-1)^2}$
$\frac{1}{p+a}$	e^{-at}	$\frac{z}{z-\alpha}$ avec $\alpha=e^{-aT_e}$
$\frac{1}{p(1+\tau p)}$	$1-e^{-t/\tau}$	$\frac{(1-\alpha)z}{(z-1)(z-\alpha)}$ avec $\alpha=e^{-T_e/\tau}$

Calcul de la transformée en z inverse :

1. Par la théorie des résidus (non abordé dans ce cours)
2. Par l'utilisation du formulaire des transformées usuelles
3. par division polynomiale (calcul des n premiers échantillons)

Exercices :

1. Calculer la transformée en Z de $s[k]=(0.5)^k$
2. Déterminer la stabilité de $H(z)=\frac{1}{(1-1{,}2.z^{-1})}$
3. Implémenter un filtre passe-bas sur Arduino. On mesure une tension analogique issue d'un capteur : $x[k]=signal+bruit$ → On souhaite filtrer avec un filtre passe-bas numériques
   Filtre passe-bas numérique : $y[k]=\alpha.y[k-1]+(1-\alpha).x[k]$ avec 0 < α < 1
   (si α → 1 alors filtrage fort (lent) et si α → 0 alors filtrage faible (rapide))
   3.1. Calculez la fonction de transfert du filtre et donnez son pôle
   3.2. pour α = 0,8, déterminer son pôle et conclure sur sa stabilité
   3.3. quel est l'effet de α sur la rapidité et sur le lissage ?
   3.3. réalisez le programme Arduino

Exercices corrigés :

1. $S(z)=\frac{1}{(1-0{,}5.z^{-1})}$
2. Instable car pôle > 1
3. 3.1. $H(z)=\frac{1-\alpha}{1-\alpha.z^{-1}}$ et le pôle est en z = α. ce filtre est stable si |α| < 1
   3.2. Le système possède donc **un pôle réel unique en z=0.8**. Le filtre est donc stable
   3.3. $y[k]=\alpha.y[k-1]+(1-\alpha).x[k]$
   cas où α est petit : α = 0,2 alors $y[k]=0{,}2.y[k-1]+0{,}8.x[k]$

La sortie dépend beaucoup de la nouvelle entrée x[k].
Le filtre réagit vite aux variations du signal d'entrée.

→ Le système est **rapide**
→ Il suit mieux les changements brusques

cas où α est grand: α = 0,9 alors $y[k]=0{,}9.y[k-1]+0{,}1.x[k]$

La sortie dépend beaucoup de la valeur précédente y[k-1].
Le filtre réagit lentement aux variations du signal d'entrée.

→ Le système est **plus lent**
→ Il met plus de temps à atteindre une nouvelle valeur

Effet de α sur le lissage

Si α est petit : Le filtre prend beaucoup en compte la mesure actuelle.

→ Il lisse peu le bruit.

Si α est grand : Le filtre garde beaucoup de mémoire du passé.

→ Les variations rapides sont atténuées.

3.3. Programme Arduino :

```
#define N 10 //Définition du nombre d'échantillons utilisés pour la moyenne


float buffer[N]; //Tableau qui stocke les N dernières mesures
// C'est une mémoire glissante (historique du signal)
int index = 0; //Indice courant dans le buffer


void loop() {
 float x = analogRead(A0); //Lecture du signal analogique (0 → 1023 sur Arduino UNO): C'est x[k]


 buffer[index] = x; //On stocke la nouvelle valeur dans le buffer
 index = (index + 1) % N; //Incrément circulaire de l'indice
// Quand on arrive à N , on revient à 0 (c'est un buffer circulaire)
// % est le reste de la division : c'est le modulo (5%2=1, 8%4=0)
 float sum = 0;
 for(int i=0; i<N; i++)
       {
       sum += buffer[i];
       }
```

//dans cette boucle, on a additionne les N dernières valeurs $\sum_{i=0}^{N-1} x[k-i]$ qui est la définition du filtre moyenne glissante

```
 float y = sum / N; //Calcul de la moyenne :
```

$$y[k]=\frac{1}{N}.\sum_{i=0}^{N-1} x[k-i]$$

```
 Serial.println(y);
}
```