

MEC6212: GENERATION de MAILLAGES

Travail pratique: Fonctions Splinaires

22 janvier 2023

Énoncé

À partir d'un nombre de points distincts (x_i, y_i) , on veut construire un interpolant par morceaux de type spline cubique :

1. Écrire 3 fonctions Matlab pour construire différentes variantes de cet interpolant, appelées :

$[P,PTS]= \text{geoSPLINE31}(CNTRL,nbCNTRL)$: spline parabolique, courbure extrapolée aux extrémités ;

$[P,PTS]= \text{geoSPLINE32}(CNTRL,nbCNTRL)$: spline avec pente imposée aux extrémités ;

$[P,PTS]= \text{geoSPLINE33}(CNTRL,nbCNTRL)$: spline cyclique.

2. Le protocole d'appel :

$[P,PTS]= \text{geoSPLINExx}(CNTRL,nbCNTRL)$

En entrée :

- ***CNTRL*** : tableau qui contient les coordonnées (x,y) des $nbCNTRL$ points de collocation
- ***nbCNTRL*** : nombre de points de collocation

En sortie :

- ***P*** : tableau qui contient les coordonnées (x,y) des PTS points interpolés à l'intérieur de chaque morceau de la spline
- ***PTS*** : nombre de points interpolés

Cette fonction sera appelée d'un programme montré à la Figure 1.

```

switch SEG(iSEG,1)%typeSEG
case 10%----- Bezier
    [P,ptsSEG] = geoBEZIER(PLN(PLN1:PLN2,1:2),SEG(iSEG,3));
    SEG(iSEG,4)= IPT+1;
    SEG(iSEG,5)= ptsSEG;%-----nombre de PT sur le segment
    for j=1:ptsSEG-1
        IPT = IPT + 1;
        GEO(iPT,1) = P(j,1);
        GEO(iPT,2) = P(j,2);
    end
case 20%----- spline
    [P,ptsSEG] = geoSPLINEXX(PLN(PLN1:PLN2,1:2),SEG(iSEG,3));
    SEG(iSEG,4) = IPT+1;
    SEG(iSEG,5) = ptsSEG;%----- nombre de PT sur le segment
    for j=1:ptsSEG-1
        IPT = IPT + 1;
        GEO(iPT,1) = P(j,1);
        GEO(iPT,2) = P(j,2);
    end
case 30%----- polynome de Lagrange
    nbCNTSEG = finCNTSEG -iniCNTSEG +1;
    [P,ptsSEG]=geoLAGRANGE(PLN(iniCNTSEG:finCNTSEG,1:2),SEG(iSEG,3));
    for j=1:ptsSEG-1
        GEO(iPT,1) = P(j,1);
        GEO(iPT,2) = P(j,2);
        IPT = IPT + 1;
    end
otherwise
end

```

FIGURE 1 – Programme appellent pour différents types d'interpolants

1 Représentation

On utilisera une représentation,

1. paramétrique plutôt qu'explicite. Le paramètre u est arbitraire, et $0 \leq u \leq (nbCTRL - 1)$ donne une paramétrisation en fonction du numéro k du point, qui est proche de la paramétrisation intrinsèque basée sur la longueur de la courbe ;
2. explicitée sur l'ensemble de la courbe, c-à-d localement sur chaque morceau $0 \leq u \leq 1$, mais globalement $0 \leq U \leq nbCTRL$.
3. La répartition des points calculés, P, peut se faire avec un nombre fixé de points par morceau ou bien sur l'ensemble de la courbe.

Dans ce travail, on variera la répartition de $PTS = 1, 2, 3, \dots$ sur chaque morceau de la courbe afin de vérifier la qualité du rendu graphique à l'écran.

2 Structure matricielle

La condition C^2 pour un raccordement de la courbure au point de contrôle \vec{P}_i est donnée par,

$$\vec{P}'_{i-1} + 4\vec{P}'_i + \vec{P}'_{i+1} = 3(\vec{P}_{i+1} - \vec{P}_{i-1})$$

Appliquée à l'ensemble des noeuds intérieurs de la courbe, ces $(n - 2)$ équations expriment la relation entre les valeurs des vecteurs tangents et celles des vecteurs positions aux noeuds de la spline.

$$\begin{array}{ccccccc}
 & & & & & & 2 \leq i \leq n - 1 \\
 \vec{P}'_1 & +4\vec{P}'_2 & +\vec{P}'_3 & & & & = 3(\vec{P}_3 - \vec{P}_1) \\
 & \vec{P}'_2 & +4\vec{P}'_3 & +\vec{P}'_4 & & & = 3(\vec{P}_4 - \vec{P}_2) \\
 & & \vec{P}'_3 & +4\vec{P}'_4 & +\vec{P}'_5 & & = 3(\vec{P}_5 - \vec{P}_3) \\
 & & & \ddots & \ddots & \ddots & = \vdots \\
 & & & & \vec{P}'_{n-2} & +4\vec{P}'_{n-1} & +\vec{P}'_n = 3(\vec{P}_n - \vec{P}_{n-2})
 \end{array}$$

On complète ce système avec l'une ou l'autre des conditions données dans l'énoncé :

- spline parabolique, courbure extrapolée aux extrémités ;
- spline avec pente imposée aux extrémités ;
- spline cyclique.

Chacune de ces conditions donne deux équations (une à chaque extrémités) qui s'ajoutent au système ci-dessus, donnant la structure générale suivante :

$$\begin{bmatrix} * & * & & & & & \\ 1 & 4 & 1 & & & & \\ & 1 & 4 & 1 & & & \\ & & 1 & 4 & 1 & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & 1 & 4 & 1 \\ & & & & & * & * \end{bmatrix} \begin{bmatrix} \vec{P}'_1 \\ \vec{P}'_2 \\ \vec{P}'_3 \\ \vdots \\ \vdots \\ \vdots \\ \vec{P}'_n \end{bmatrix} = 3 \begin{bmatrix} * \\ \vec{P}_3 - \vec{P}_1 \\ \vec{P}_4 - \vec{P}_2 \\ \vec{P}_5 - \vec{P}_3 \\ \vdots \\ \vec{P}_n - \vec{P}_{n-2} \\ * \end{bmatrix}$$

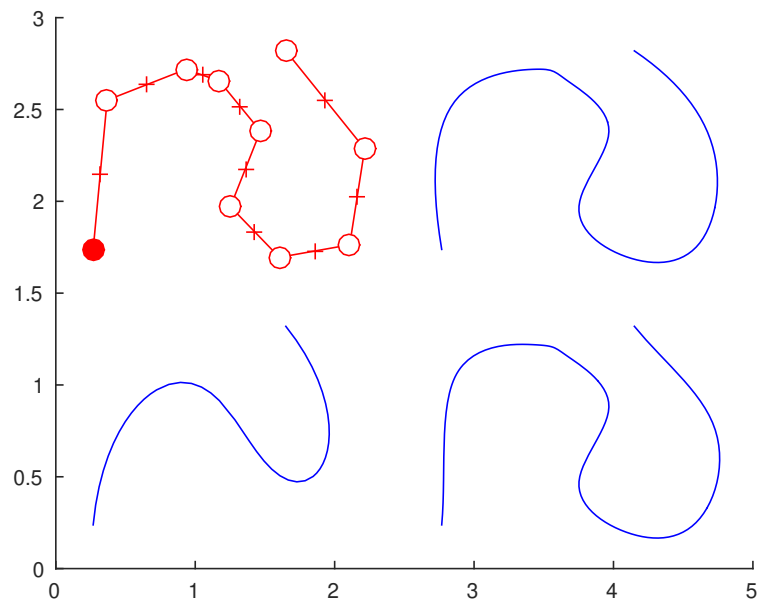
où seuls les éléments marqués par "*" varient selon le type de conditions aux extrémités.

Les fonctions demandées devront être structurées de manière générique, où le système matriciel comprend une partie centrale invariante, et une première et dernière équation pour les différentes conditions.

Finalement, la "tridiagonalité" du système devra être maintenue, à l'exception de la spline cyclique.

3 Résultats

1. A l'aide de quelques essais, valider :
 - l'approche de l'interpolation par morceaux en comparant avec le polynôme de Lagrange ;
 - la représentation paramétrique quant à la possibilité d'approximer une grande variété de courbes, notamment, des courbes qui se croisent ;
2. Analyser/critiquer le choix du paramètre $0 \leq u \leq (nbCTRL - 1)$ par rapport à une paramétrisation intrinsèque basée sur la longueur de la courbe.
3. Comparer, pour une même polygone de contrôle, les deux variantes, geoSPLINE31 et geoSPLINE32.



Modifier si nécessaire les points de contrôle pour bien faire ressortir la différence entre ces deux variantes.

4. Rédiger un document, en format **identifiant.pdf** qui résume (succinctement!) vos résultats ainsi qu'une discussion critique sur les splines comme technique d'approximation de courbes.
5. Dans un fichier **identifiant.zip**, les trois fichiers **geoSPLINExx.m** ainsi que la discussion **identifiant.pdf**, et remettre sur le site du cours à la rubrique

Travail pratique no. 1

Remise TP1