



R my Malgouyres,  
Algorithmes pour la synth se d'images  
et l'animation 3D,  
DUNOD, 2002

## TP n  6 : Courbes Splines

### Objectifs :

Le but de ce TP est de compl ter la hi rarchie de classes vue au TP1 et au TP2. On impl mentera ici des classes repr sentant les courbes splines uniforme non rationnelles, courbes splines no uniformes avec strat gie centrip te de choix du vecteur nodal, et courbes  $\beta$ -splines.

### 1 Les classes $B$ -splines uniformes

Concevoir une hi rarchie de classes permettant de repr senter diff rentes courbes  $B$ -splines. La classe de base, que l'on pourra appeler `BSpline`, h ritera de la classe `Courbe2D` d finie au TP1. On impl mentera ensuite une classe, que l'on pourra appeler `BSplineUniforme` permettant de repr senter un courbe  $B$ -spline non rationnelle avec vecteur nodal uniforme.

Pour la fonction calculant des points sur la courbe, on effectuera un calcul des valeurs  $N_{i,k}(t)$  des fonctions de la base de splines.

Donnera la possibilit    l'utilisateur de d finir des courbes  $B$ -splines uniformes ouvertes ou ferm es.

### 2 La classe `BSplines Centrip tes`

On int grera au programme pr c dent une classe `BSplineCentripete`, permettant de repr senter des courbes  $B$ -splines avec strat gie centrip te pour le choix du vecteur nodal.

Pour changer, la fonction calculant les points de la courbe sera bas e sur l'algorithme de Boor.

Donnera la possibilit    l'utilisateur de d finir des courbes  $B$ -splines centrip tes ouvertes ou ferm es.

### 3 La classe `BetaSpline`

On impl mentera une classe `BetaSpline`, permettant de repr senter une courbe  $\beta$ -spline cubique.

La fonction calculant des points sur la courbe sera bas e sur une  quation matricielle. On effectuera le plus possible d'op ration matricielles sous forme de pr calculs dans le constructeur.