Bernstein-Bézier Polynomial Basis Presentation

April 24, 2018

Pierre Jacquet
pierre.jacquet@inria.fr

First year PhD Student
Inria - Magique 3D - DIP

Pau, FRANCE

Outline

Pierre Bézier [1]

Dates: 1910-1999
Nationality: French
Institution: Arts et Métiers
Renault

Describes the Curves in 1962

[1] https://macquebec.com/courbes-bezier/
[2] http://rocbor.net/Product/surf/PaulDeFagetDeCasteljau.html

Pierre Bézier [1]

Dates: 1910-1999
Nationality: French
Institution: Arts et Métiers

Renault

Describes the Curves in 1962

Paul de Faget de Casteljau [2]

Dates: 1930 - ——
Nationality: French
Institution: Ecole Normale Supérieure

Citroën

Describes the Curves in 1958

[1] https://macquebec.com/courbes-bezier/
[2] http://rocbor.net/Product/surf/PaulDeFagetDeCasteljau.html

Multiple Applications:

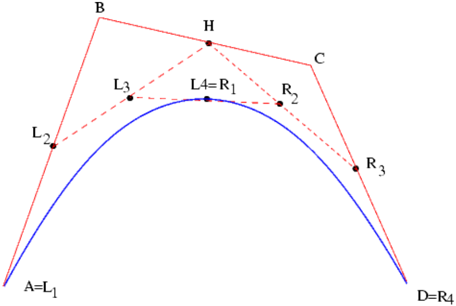


Figure: Quadratic Bézier-Curve[1]

[1] https://fr.wikipedia.org/wiki/Courbe_de_Bézier

Multiple Applications:

- Computer-Aided Design (CAD)

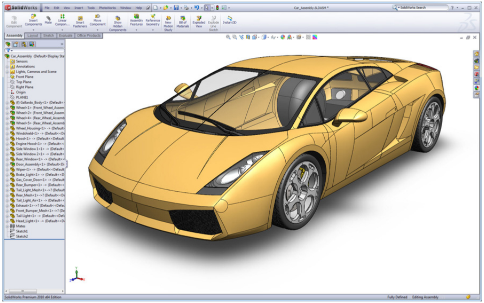


Figure: Car model on SolidWorks [2]

[2] https://www.quora.com/What-are-the-best-cad-softwares-for-automobile-designing

Bézier Curves
Applications

Multiple Applications:

- Computer-Aided Design (CAD)
- Typeface

Figure: Exemple of typeface [3]

[3] https://4design.xyz/courbes-de-bezier-regulieres-illustrator

Multiple Applications:

- Computer-Aided Design (CAD)
- Typeface
- Medical Imaging

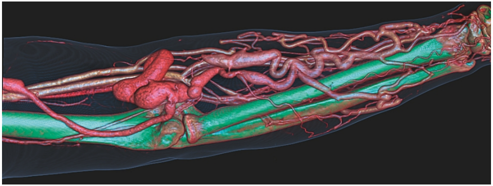


Figure: Medical Imaging [4]

[4] https://www.siemens.com/innovation/en/home/pictures-of-the-future/health-and-well-being/medical-imaging-dossier.html

Multiple Applications:

- Computer-Aided Design (CAD)
- Typeface
- Medical Imaging
- Easing Functions (cognitive science)

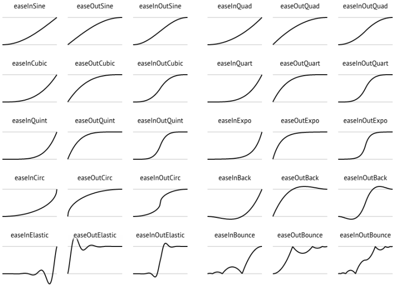


Figure: Easing Functions [5]

[5] http://easings.net/fr

Multiple Applications:

- Computer-Aided Design (CAD)
- Typeface
- Medical Imaging
- Easing Functions (cognitive science)
- Data Interpolations

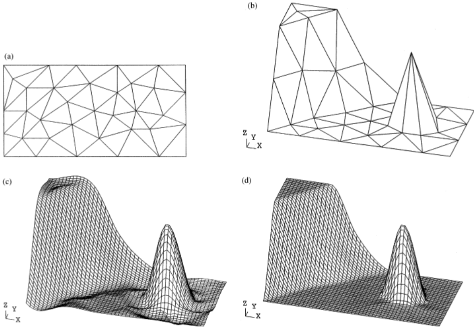


Figure: Data Interpolations [6]

[6] Range restricted scattered data interpolation using convex combination of cubic Bézier triangles, E.S.Chan B.H.Ong

Multiple Applications:

- Computer-Aided Design (CAD)
- Typeface
- Medical Imaging
- Easing Functions (cognitive science)
- Data Interpolations
- “Photoshop”...

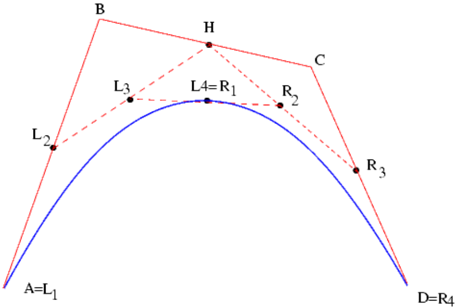


Figure: Wonderfull Photoshop[1]

[1] Elvira’s Photoshop

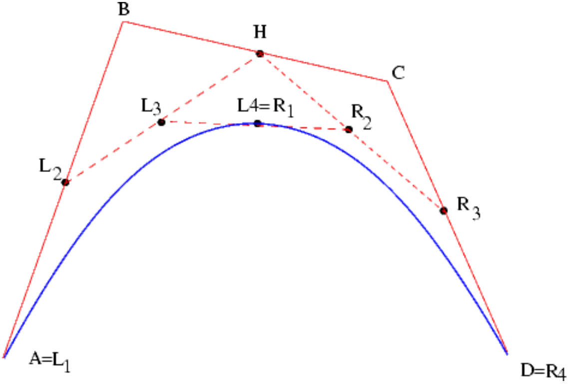


Figure: Quadratic Bézier Curve

(See construction on Geogebra)

Why the Bézier Curves are so popular ?

- Intuitive barycentric formulation
- Flexible Behaviour
- Easy extension to 2D or 3D
- Low Memory cost
- Various applications

Why the Bézier Curves are so popular ?

- Intuitive barycentric formulation
- Flexible Behaviour
- Easy extension to 2D or 3D
- Low Memory cost
- Various applications

and ...

Why the Bézier Curves are so popular ?

- Intuitive barycentric formulation
- Flexible Behaviour
- Easy extension to 2D or 3D
- Low Memory cost
- Various applications

and ...

- Easily represented in **Bernstein Polynomial Basis**

Sergei Natanowitsch Bernstein [1]

Dates: 1880-1968
Nationality: Russian
Institution: Sorbonne / Supéléc
Kharkov University

Introduce his polynomial basis in 1912

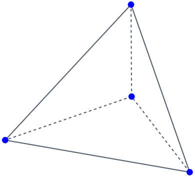
$\Rightarrow$ prove the Weierstrass approximation theorem

[01] https://fr.wikipedia.org/wiki/Serge%C3%AF_Natanovitch_Bernstein

Bernstein formulation:

$$B_{ijkl}^{N} = C_{ijkl}^{N} \lambda_0^i \lambda_1^j \lambda_2^k \lambda_3^l \quad \text{with:} \quad C_{ijkl}^{N} = \frac{N!}{i!j!k!l!}$$

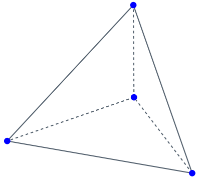
$$\text{and :} \quad i + j + k + l = N$$



Bernstein formulation:

$$B_{ijkl}^{N} = C_{ijkl}^{N}\lambda_0^i\lambda_1^j\lambda_2^k\lambda_3^l \quad \text{with: } C_{ijkl}^{N} = \frac{N!}{i!j!k!l!}$$

$$\text{and : } \quad i+j+k+l = N$$



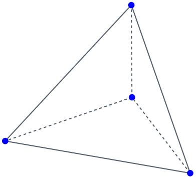
In 1D it gives :

$$B_i^N(t) = C_i^N(1-t)^{N-i}t^i$$

Bernstein formulation:

$$B^N_{ijkl} = C^N_{ijkl}\lambda_0^i\lambda_1^j\lambda_2^k\lambda_3^l \quad \text{with:} \quad C^N_{ijkl} = \frac{N!}{i!j!k!l!}$$

$$\text{and :} \quad i + j + k + l = N$$



In 1D it gives :

$$B_i^N(t) = C_i^N(1-t)^{N-i}t^i$$

NB: Which is equal to the probability of i successes in trials of N random process with individual probability of t success in each trial.

Bernstein Polynomials
Basis Function Properties

Bernstein Basis Properties :

- Highly correlated to probabilities
- Partition of unity : $\sum_{k=0}^{n} B_k^N(t) = 1$
- Positivity : $\forall t \in [0,1], B_k^n(t) \leq 0$
- Symmetry : $B_k^N(t) = B_{N-k}^N(1-t)$
- Describes Bézier curves (See construction on Geogebra)
- De Casteljau Algorithm

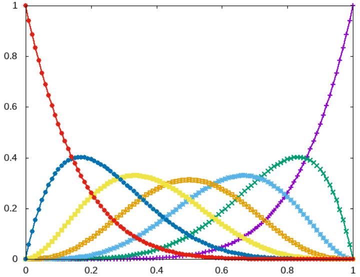


Figure: $P[X^6]$ Bersntein basis

Bernstein Basis Properties :

- Highly correlated to probabilities
- Partition of unity : $\sum_{k=0}^{n} B_k^N(t) = 1$
- Positivity : $\forall t \in [0, 1]$, $B_k^n(t) \leq 0$
- Symmetry : $B_k^N(t) = B_{N-k}^N(1 - t)$
- Describes Bézier curves (See construction on Geogebra)
- De Casteljau Algorithm

and...

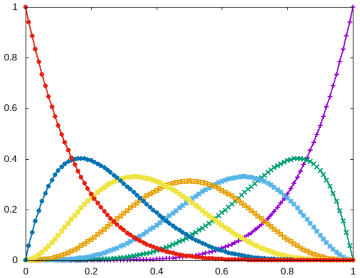


Figure: $P[X^6]$ Bersntein basis

Bernstein Basis Properties :

- Highly correlated to probabilities
- Partition of unity : $\sum_{k=0}^{n} B_k^N(t) = 1$
- Positivity : $\forall t \in [0,1]$, $B_k^n(t) \leq 0$
- Symmetry : $B_k^N(t) = B_{N-k}^N(1-t)$
- Describes Bézier curves (See construction on Geogebra)
- De Casteljau Algorithm

and...

- Other interessant proprieties for DGM formulations

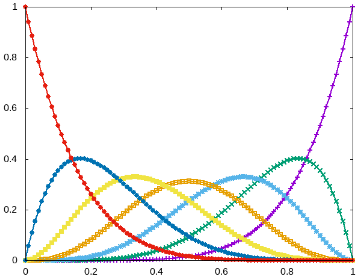
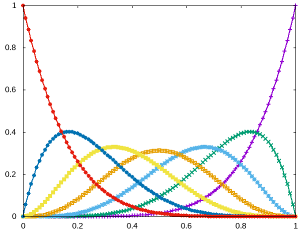


Figure: $P[X^6]$ Bersntein basis

Bernstein formulation:

$$B^N_{ijkl} = C^N_{ijkl}\lambda_0^i\lambda_1^j\lambda_2^k\lambda_3^l \quad \text{with: } C^N_{ijkl} = \frac{N!}{i!j!k!l!}$$



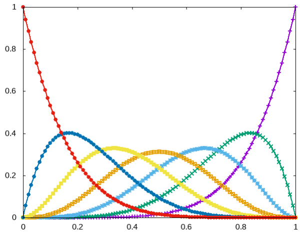
P[X^6] Bernstein basis

Bernstein formulation:

$$B^N_{ijkl} = C^N_{ijkl}\lambda_0^i\lambda_1^j\lambda_2^k\lambda_3^l \quad \text{with: } C^N_{ijkl} = \frac{N!}{i!j!k!l!}$$

Easy Derivative expression:

$$\frac{\partial B^N_\alpha}{\partial \lambda_p} = NB^{N-1}_{\alpha - e_p} \quad \text{with: } \alpha = (i, j, k, l)$$



$P[X^6]$ Bernstein basis

Bernstein formulation:

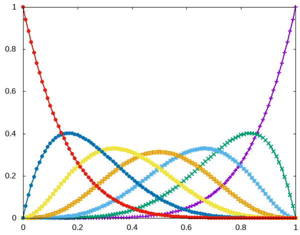
$$B^N_{ijkl} = C^N_{ijkl}\lambda_0^i\lambda_1^j\lambda_2^k\lambda_3^l \quad \text{with: } C^N_{ijkl} = \frac{N!}{i!j!k!l!}$$

Easy Derivative expression:

$$\frac{\partial B^N_\alpha}{\partial \lambda_p} = NB^{N-1}_{\alpha-e_p} \quad \text{with: } \alpha = (i,j,k,l)$$

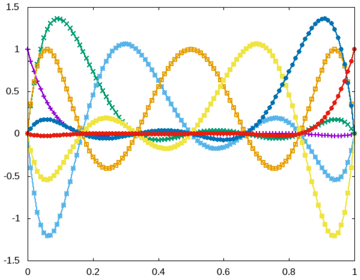
Sparse Degree Elevation operator:

$$B^{N-1}_\alpha = \sum_{p=0}^{d} \frac{\alpha_p+1}{N} B^N_{\alpha+e_p}$$

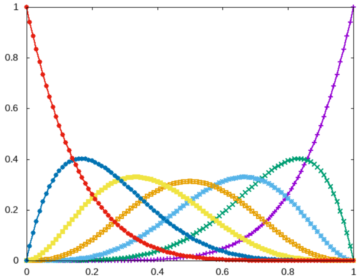


P[X^6] Bernstein basis

Unique boundary condition values :



(a) P(X^6) Lagrange basis



(b) P(X^6) Bernstein basis

$\Rightarrow$ Same Flux Management

First order Acoustic Wave equation discretisation:

Continuous Problem:

$$\begin{cases} \dfrac{\partial p}{\partial_t} + \kappa_0 div(\overrightarrow{u}) = 0 \\ \rho_0 \dfrac{\partial \overrightarrow{u}}{\partial_t} + \overrightarrow{grad}(p) = 0 \end{cases}$$

Discrete Problem:

$$\begin{cases} M\dfrac{\partial P}{\partial_t} + \sum_d S_d U_d = M^F F_p \\ M\dfrac{\partial U_d}{\partial_t} + S_d P = M^F F_u \end{cases}$$

Application on DGs
Discontinuous Galerkin formulation

First order Acoustic Wave equation discretisation:

Continuous Problem:

$$\begin{cases} \dfrac{\partial p}{\partial_t} + \kappa_0 div(\overrightarrow{u}) = 0 \\ \rho_0 \dfrac{\partial \overrightarrow{u}}{\partial_t} + \overrightarrow{grad}(p) = 0 \end{cases}$$

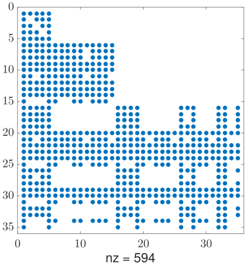
Discrete Problem:

$$\begin{cases} M\dfrac{\partial P}{\partial_t} + \sum_d S_d U_d = M^F F_p \\ M\dfrac{\partial U_d}{\partial_t} + S_d P = M^F F_u \end{cases}$$

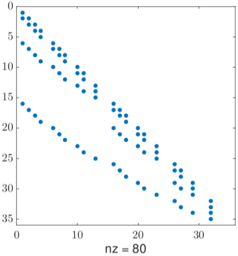
With $Sd = MD_d$ and $Lift = M^{-1}M^F$:

$$\begin{cases} \dfrac{\partial P}{\partial_t} + \sum_d D_d U_d = LiftF_p \\ \dfrac{\partial U_d}{\partial_t} + D_d P = LiftF_u \end{cases}$$

Application on DGs
Operator Evaluation



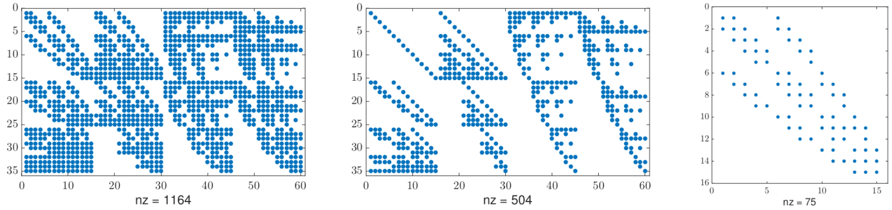
(c) 3D Lagrange D matrix



(d) 3D Bernstein D matrix

[1] Chan Jesse and Warburton T.
GPU-Accelerated Bernstein Bézier Discontinuous Galerkin Methods for Wave Problems
SIAM Journal on Scientific Computing 2017

Application on DGs
Operator Evaluation



(a) Bernstein lift matrix (b) $\boldsymbol{E}_L$ (c) $\boldsymbol{L}_0$

Bernstein Lift matrix operator decomposition

With $Lift = E_L * L_0$

[1] Chan Jesse and Warburton T.
GPU-Accelerated Bernstein Bézier Discontinuous Galerkin Methods for Wave Problems
SIAM Journal on Scientific Computing 2017

$$\begin{cases} P^{n+\frac{1}{2}} = P^{n-\frac{1}{2}} + A_p U^n + RHS_p \\ U^{n+1} = U^n + A_u P^{n+\frac{1}{2}} + RHS_u \end{cases}$$

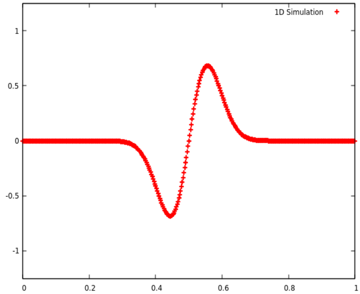
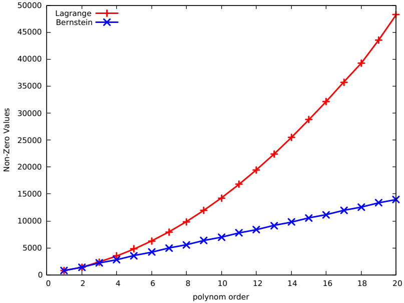
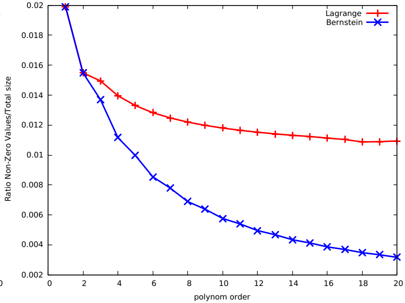


Figure: First 1D DG Simulation

Application on DGs
Few Personal Results



A_p/A_u NZVs as a function of the order



A_p/A_u NZVs Ratio ($NZVs/Size$)

Main results about Bernstein-Bézier polynomial basis :

- Plenty of utilization of Bézier curves
- Plenty of numerical properties of Bernstein polynomial basis
- Recently used in numerical simulation
- Create very Sparse Matrices
- Same Flux management as a nodal polynomial basis
- Easy elevation/reduction degree (useful for p-adaptivity)
- Easy Derivative