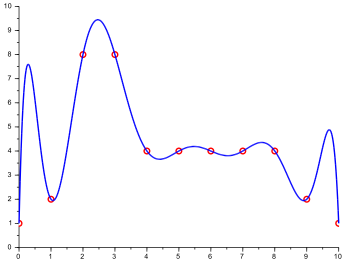
Chapter 5

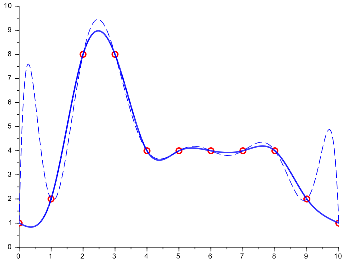
Spline interpolation

What is a spline?

- A *spline function* is a piecewise function.
  The idea is to combine simple functions (typically low degree polynomials) so as to form a smooth composite function (i.e., of class C^k).
  We distinguish *algebraic splines* (connection of polynomials or simple functions under differential continuity constraints) from the *functional splines* obtained by minimization of a functional representing an energy.
  The associated curve is called a *spline curve*.
- An *interpolating spline* is a spline curve which is constrained to pass through prescribed points (x_i, y_i).



Lagrange interpolation
one polynomial of degree 10



C^2 cubic spline interpolation
10 polynomials of degree 3

Introduction —

Definition of a space of spline functions

Interpolation problem

$\downarrow \quad \downarrow$

algebraic approach $\downarrow \quad \downarrow$ *functional approach*

physical approach

$\downarrow \quad \downarrow$

Space of Spline functions

$\downarrow \quad \downarrow$

construction Interpolation data

(algebraic) $\swarrow \quad \swarrow$

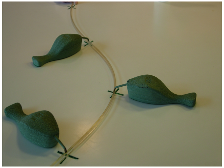
$\downarrow \quad \downarrow$ $\swarrow \quad \swarrow$

Interpolating solution

Which spline will we consider ?

	C^2 cubic splines *Natural splines*	*Tension splines* *(Exponential splines)*
Data constraints :	Lagrange data (x_i, y_i)	Lagrange data (x_i, y_i)
Interpolating solution :	*piecewise cubic*	*piecewise* $1, x, \exp(x), \exp(-x)$
Smoothness :	C^2	C^2
Approach(es) :	*algebraic & functional*	*functional*

- **C^2 cubic splines – natural splines**

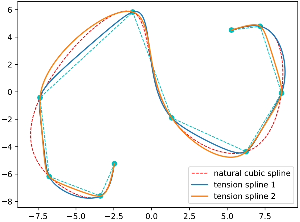


https ://secure.boeingimages.com/archive/

$$\min_{h\in \Omega_2} \int_a^b \left| \frac{h''(x)}{[1+h'(x)^2]^{3/2}} \right|^2 dx$$

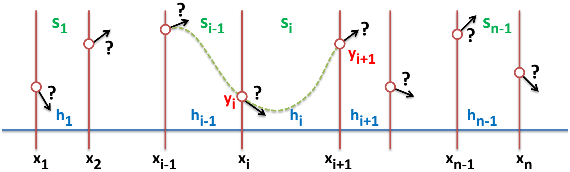
$$\min_{h\in \Omega_2} \int_a^b h''(x)^2 \, dx.$$

- **Tension splines**

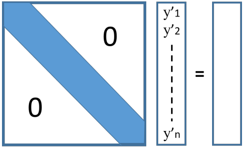


$$\min_{f\in \Omega_{2,4}} \underbrace{\int_a^b f''(x)^2 \, dx}_{\text{smoothness}} + \underbrace{\sum_{i=1}^{N-1} \sigma_i^2 \int_{x_i}^{x_{i+1}} f'(x)^2 \, dx}_{\text{tension}}$$

- **Spline space & Algebraic construction**



$s_1''(x_1) = 0$ $\quad$ $s_{i-1}''(x_i) = s_i''(x_i)$, $i = 2, \ldots, n-1$ $\quad$ $s_{n-1}''(x_n) = 0$



Tridiagonal system

Definitions

For the remainder of this chapter we consider the Lagrange interpolation of a sequence of points (x_i, y_i), $i = 1, \dots, n$, with

$$a = x_1 < x_2 < \cdots < x_{n-1} < x_n = b.$$

Be aware that the first interpolation point has index equal to 1 (and not 0).
Interpolation points x_i are called the *nodes* of the spline.

Let $k, m \in \mathbb{N}$ and let $\Pi_k^m(x_i)$ — or simply Π_k^m — be the set of real functions f (the *spline functions*) defined on $[a, b]$ by :

- $f \in C^k([a, b])$,
- the restriction f_i of f on each interval $[x_i, x_{i+1}]$ is a polynomial of degree $\leq m$.

We will verify in the following that Π_k^m is a vector space. A first analysis on the number of *degrees of freedom* shows that :

- any element of Π_k^m is composed of $n-1$ pieces of polynomials of degree $\leq m$, which provides $(n-1)(m+1)$ free parameters,
- contact constraints at inner nodes $x_2, x_3, \dots, x_{n-1}$ lead to $(n-2)(k+1)$ conditions : connection of the derivatives from the order 0 to the order k,

which suggests that Π_k^m is of dimension $(n-1)(m+1) - (n-2)(k+1)$, i.e., *“number of free parameters less number of constraints”*

Characterization

The previous analysis is *fundamental* but it does not constitute a demonstration as some constraints can be not linearly independent : see exercise "*An instructive example*" in chapter about Hermite interpolation.

Proposition 5.1

1. $\Pi_m^m \subset \Pi_{m-1}^m \subset \cdots \subset \Pi_k^m \subset \cdots \subset \Pi_1^m \subset \Pi_0^m$ *and* $\Pi_k^m \subset \Pi_k^{m+1} \subset \cdots$
2. $\Pi_k^m = \mathbb{R}_m[x]$ *for* $k \geq m$
3. Π_k^m *is a vector space.*
4. $\dim \Pi_k^m = (n-1)(m+1) - (n-2)(k+1)$

Proof (part 1)

1. *Clear.*
2. *Two polynomials of same degree* m *with* C^m *contact at a common point* ξ *are identical. Indeed, Taylor expansion of order* m *of each polynomial at point* ξ *gives the result.*
3. *Any polynomial of degree* $\leq m$*, and in particular the zero polynomial, belongs to* Π_k^m*, so that* Π_k^m *is non empty. Finally,* Π_k^m *is a linear subspace of* $C^k([a,b])$.

Proof (part 2)

4. *Let* $f = (f_1, \ldots, f_i, \ldots, f_{n-1}) \in \Pi_k^m$ *with* $f_i(x) = \sum_{j=0}^m a_j^i x^j$, $x \in [x_i, x_{i+1}]$.
Element f *is thus determined by the collection of coefficients* a_j^i *with* $1 \le i \le n-1$ *and* $0 \le j \le m$, *under* C^k *connection constraints at each inner node* $x_2, \ldots, x_{n-1}$, *which are expressed by the* $k+1$ *conditions* $f_{i-1}^{(l)}(x_i) = f_i^{(l)}(x_i)$, $l = 0, 1, \ldots, k$.
We introduce the linear map φ *from* $\mathbb{R}^{(n-1)(m+1)}$ *to* $\mathbb{R}^{(n-2)(k+1)}$, *defined by*

$$\varphi : \quad \left(a_0^1, a_1^1, \ldots, a_m^1, \ldots\ldots, a_0^{n-1}, a_1^{n-1}, \ldots, a_m^{n-1}\right) \quad \longmapsto \quad \begin{pmatrix} f_1(x_2) - f_2(x_2) \\ f_1'(x_2) - f_2'(x_2) \\ \vdots \\ f_1^{(k)}(x_2) - f_2^{(k)}(x_2) \\ \vdots \\ f_{n-2}(x_{n-1}) - f_{n-1}(x_{n-1}) \\ f_{n-2}'(x_{n-1}) - f_{n-1}'(x_{n-1}) \\ \vdots \\ f_{n-2}^{(k)}(x_{n-1}) - f_{n-1}^{(k)}(x_{n-1}) \end{pmatrix}.$$

The rank of this linear mapping is maximum. Precisely, the rank of φ *can be determined by extracting an invertible triangular square matrix of maximal size* $(n-2)(k+1)$ *from the matrix* $\mathrm{mat}(\varphi)$. *Thus,* $\ker\varphi$ *that can be identified with* Π_k^m *is of dimension* $(n-1)(m+1) - (n-2)(k+1)$, *which gives the result.*

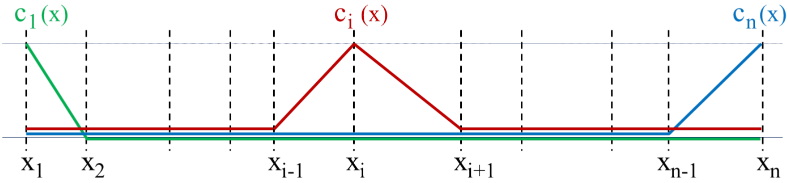
Example 1 : the space Π_0^1

Π_0^1 is the set of continuous piecewise linear functions on $[a, b]$, associated with the sequence of nodes x_i.
A function $s \in \Pi_0^1$ is completely determined by the set of its values at the nodes x_i.

Π_0^1 is a vector space of dimension n. It accepts as a basis the set of the *hat functions* c_i defined by

$$c_i(x_j) = \delta_{ij}, \quad i, j \in \{1, 2, \ldots, n\},$$

where δ_{ij} is the Kronecker symbol.



Thus, any continuous piecewise linear interpolating spline $s \in \Pi_0^1$ can be expressed in the following way

$$s(x) = \sum_{i=1}^{n} y_i \, c_i(x) \,, \quad x \in [a, b] = [x_1, x_n].$$

Example 2 : the space Π_0^2

Π_0^2 is the set of continuous piecewise quadratic functions on $[a,b]$, associated with the sequence of nodes x_i.

We now detail the construction of the linear mapping φ introduced in the proof of the proposition 5.1. Let $f = (f_1, f_2, \dots, f_{n-1}) \in \Pi_0^2$ with $f_i(x) = a_i + b_i\,x + c_i\,x^2$. The continuity constraint C^0 at each inner node $x_i,\ i = 2, \dots, n-1$ is given by

$$f_{i-1}(x_i) = f_i(x_i) \quad \Leftrightarrow \quad (a_{i-1} + b_{i-1}\,x_i + c_{i-1}\,x_i^2) - (a_i + b_i\,x_i + c_i\,x_i^2) = 0$$

which leads to the linear mapping $\varphi : \mathbb{R}^{3(n-1)} \to \mathbb{R}^{n-2}$ defined by

$$(a_1, b_1, c_1, \dots, a_{n-1}, b_{n-1}, c_{n-1}) \longmapsto \begin{pmatrix} a_1 + b_1\,x_2 + c_1\,x_2^2 - a_2 - b_2\,x_2 - c_2\,x_2^2 \\ a_2 + b_2\,x_3 + c_2\,x_3^2 - a_3 - b_3\,x_3 - c_3\,x_3^2 \\ \vdots \\ a_{n-2} + b_{n-2}\,x_{n-1} + c_{n-2}\,x_{n-1}^2 - a_{n-1} - b_{n-1}\,x_{n-1} - c_{n-1}\,x_{n-1}^2 \end{pmatrix}$$

from which we deduce the matrix mat(φ) :

$$\begin{bmatrix} 1 & x_2 & x_2^2 & -1 & -x_2 & -x_2^2 & 0 & \cdots & & & \cdots & 0 \\ 0 & 0 & 0 & 1 & x_3 & x_3^2 & -1 & -x_3 & -x_3^2 & 0 & \cdots & 0 \\ \vdots & & & & & & & & & & & \vdots \\ 0 & 0 & \cdots & & \cdots & 0 & 1 & x_{n-1} & x_{n-1}^2 & -1 & -x_{n-1} & -x_{n-1}^2 \end{bmatrix}$$

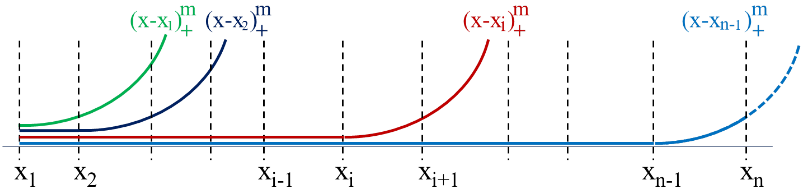
By extracting the $n-2$ first columns congruent to 1 modulo 3, we verify that φ is of maximal rank $n-2$, so that $\dim\,\Pi_0^2 = \dim\,\ker(\varphi) = 3(n-1) - (n-2) = 2n-1$.

The functions $(x - x_i)_+^m$

For $m \geq 1$, consider the $n-1$ real functions defined on the interval $[a, b]$ by

$$x \longmapsto (x - x_i)_+^m = \begin{cases} 0 & \text{if } x \leq x_i \\ (x - x_i)^m & \text{if } x \geq x_i \end{cases} \qquad \text{for} \quad i = 1, 2, \ldots, n-1.$$

For $i = n$, the function $(x - x_n)_+^m$ is identically zero on $[a, b]$ and is therefore useless.



Proposition 5.2

These $n-1$ functions form a linearly independent family of the space Π_{m-1}^m

The key point for the proof is to check the C^{m-1} continuity at point x_i for each function $(x - x_i)_+^m$

These functions are useful for constructing basis of vector spaces of spline functions, but they are inappropriate for numerical and algorithm applications.

Example 3 : the space Π_{m-1}^m

Proposition 5.3

Π_{m-1}^m *is a vector space of dimension* $n+m-1$ *and accepts as a basis the following family of functions*

$$\left\{1,\, x,\, x^2,\, \ldots,\, x^m,\, (x-x_2)_+^m,\, (x-x_3)_+^m,\, \ldots,\, (x-x_{n-1})_+^m\right\}.$$

Proof

By proposition 5.1 we just need to prove that this family is linearly independent. For this purpose, consider a linear combination of functions of this family equal to zero, that is

$$\sum_{i=0}^{m} a_i\, x^i + \sum_{j=2}^{n-1} b_j\, (x-x_j)_+^m \;=\; 0$$

On the interval $[x_1, x_2]$*, functions* $(x-x_j)_+^m$, $j = 2, \ldots, n-1$*, are all zero, from which we deduce that coefficients* a_i *must be all zero.*
Then, evaluating this linear combination at points x_k *for* $k = 3, \ldots, n$*, proves that coefficients* b_j *are also all zero. Which concludes the proof.*

C^2 **cubic splines :** space Π_2^3 and interpolation

By proposition 5.3 the dimension of the vector space Π_2^3 is $n+2$ = number of nodes $+\,2$ and Π_2^3 accepts the following basis

$$\left\{1,\, x,\, x^2,\, x^3,\, (x-x_2)_+^3,\, (x-x_3)_+^3,\, \ldots,\, (x-x_{n-1})_+^3\right\}$$

We now wish to impose Lagrange interpolation conditions at nodes $x_1, ..., x_n$.

- Precisely, given a family of n real values $y_1, \ldots, y_n$, we consider the problem of determining a spline function f in Π_2^3 satisfying the n interpolation constraints $f(x_i) = y_i$ for $i = 1, \ldots, n$.
- The analysis of this interpolation problem shows that we have $n+2$ free parameters (the dimension of Π_2^3) and n constraints, so that we have 2 degrees of freedom remaining.
- These 2 remaining degrees of freedom are fixed by imposing additional constraints to the interpolating spline.
- The most common choice is the one presented below and leads to the definition of *natural splines.* However, other variants exist to fix these two degrees of freedom (see next slide)

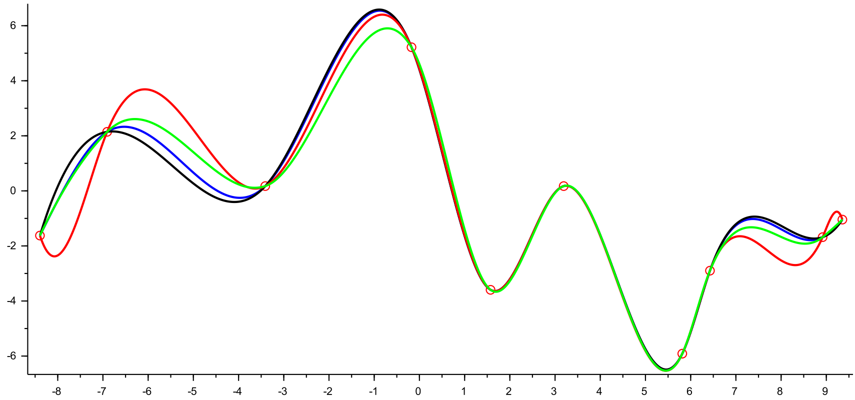
C^2 **cubic splines :** space Π_2^3 and interpolation

The following *scilab* script proposes (for the same dataset) 4 different interpolating C^2 cubic splines, including natural splines : only the end-conditions are different.

```
// acquisition of a strictly increasing sequence of n data points xi
// with associate constraints yi
tt = linspace(xi(1),xi(n),1000);

// NATURAL SPLINE : s''(xi(1)) = s''(xi(n)) = 0
// derivatives d1(i) = s'(xi(i)) at nodes xi(i)
d1 = splin(xi,yi,"natural");
yy = interp(tt,xi,yi,d1);
plot(tt,yy,'b-','linewidth',2)

// SPLINE "NOT A KNOT" : continuity C^3 in xi(2) and xi(n-1) :
d2 = splin(xi,yi,"not_a_knot");
yy2 = interp(tt,xi,yi,d2);
plot(tt,yy2,'k-','linewidth',2)

// SPLINE "CLAMPED" : Derivatives at ends are fixed :
// s'(xi(1)) = der_1 and s'(xi(n)) = der_n
der_1 = -5; der_n = -5;
d3 = splin(xi,yi,"clamped",[der_1 der_n]);
yy3 = interp(tt,xi,yi,d3);
plot(tt,yy3,'r-','linewidth',2)

// SPLINE "FAST" : pseudo-spline obtained by local interpolation
d4 = splin(xi,yi,"fast");
yy4 = interp(tt,xi,yi,d4);
plot(tt,yy4,'g-','linewidth',2)
```

C^2 **cubic splines :** space Π_2^3 and interpolation

The following *scilab* script proposes (for the same dataset) 4 different interpolating C^2 cubic splines, including natural splines : only the end-conditions are different.



"NATURAL SPLINE" (blue) — SPLINE "NOT A KNOT" (black) — SPLINE "CLAMPED" (red) — SPLINE "FAST" (green)

Definition

The space S of *natural splines* is the set of C^2 cubic splines of the space Π_2^3 whose second derivative is zero at points x_1 and x_n, that is

$$S = \left\{ f \in \Pi_2^3 ,\, f''(x_1) \,=\, f''(x_n) = 0 \right\}$$

Notice that such a natural spline defined on the interval $[a,b]$ can be extended to $\mathbb{R}$ in a C^2 function whose restriction to intervals $]-\infty,\, x_1]$ and $[x_n, +\infty[$ is affine (geometrically, two half-lines).

Proposition 5.4 (natural splines)

1. *The set S of natural splines is a vector space of dimension n.*
2. *Given n real values $y_1, y_2, \ldots, y_n$, there exists a unique natural spline s interpolating the data (x_i, y_i), that is, satisfying the constraints*

$$s(x_i) = y_i, \quad i = 1, 2, \ldots, n.$$

Proof

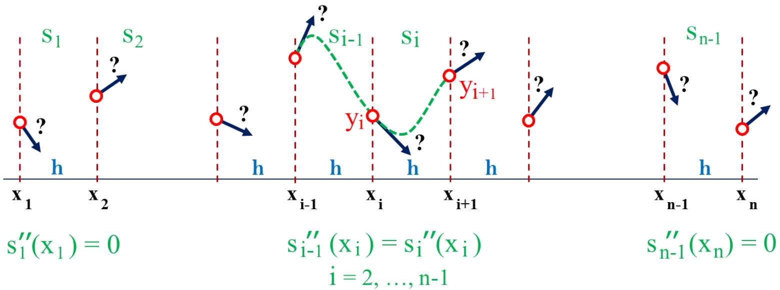
1. *The vector space structure of S is clear.*
   Let $f \in \Pi_2^3$*, we have*
   $$\begin{array}{rcl} f(x) & = & a_0 + a_1\,x + a_2\,x^2 + a_3\,x^3 + \sum_{i=2}^{n-1} \delta_i\,(x - x_i)_+^3 \\ f''(x) & = & 2\,a_2 + 6\,a_3\,x + \sum_{i=2}^{n-1} 6\,\delta_i\,(x - x_i)_+ \end{array}$$
   Conditions $f''(x_1) = 0$ *and* $f''(x_n) = 0$ *are expressed in the form*
   $$\begin{array}{rcl} 2a_2 + 6\,a_3\,x_1 & = & 0 \\ 2a_2 + 6\,a_3\,x_n + \sum_{i=2}^{n-1} 6\,\delta_i\,(x_n - x_i) & = & 0 \end{array}$$
   Consider then the mapping φ *from* $\mathbb{R}^{n+2}$ *to* $\mathbb{R}^2$ *defined by*
   $$(a_0, a_1, a_2, a_3, \delta_2, \delta_3, \ldots, \delta_{n-1}) \longmapsto \left(2\,a_2 + 6\,a_3\,x_1,\ 2\,a_2 + 6\,a_3\,x_n + \sum_{i=2}^{n-1} 6\,\delta_i\,(x_n - x_i)\right)$$
   We verify that this application is linear and of maximum rank. The vector space S of natural splines identify with the kernel of this linear mapping φ and the rank-nullity theorem gives the result.

2. *The proof of this second point* (existence and uniqueness of an interpolating natural spline) *is essentially constructive and is detailed in the following section.*
   We will consider two approaches : the classical algebraic approach and a functional approach. Both uniform and non-uniform cases will be considered.

Uniform case (1)

In order to simplify the presentation, we first consider the *uniform* case of evenly spaced nodes : $x_{i+1} - x_i = h, \ i = 1, \ldots, n-1$.

- Our objective is to construct a natural spline $s \in S$ interpolating the data (x_i, y_i) for $i = 1, \ldots, n$, and to prove that such a natural spline is unique.
- For this purpose, we assign to each node x_i an arbitrary derivative y_i' and we consider the unique Hermite-interpolating C^1 cubic spline s associated with the Hermite data $(x_i, y_i, y_i'), \ i = 1, \ldots, n$, as defined previously.
  As usual, let s_i be the restriction of s to the interval $[x_i, x_{i+1}]$ which is the unique cubic Hermite interpolating polynomial of data (x_i, y_i, y_i') and $(x_{i+1}, y_{i+1}, y_{i+1}')$.



Uniform case (2)

- The problem is now the following : *determine a set of derivative values* $y'_1, y'_2, \ldots, y'_n$ *such that this* C^1 *cubic spline* s *belongs to* S, which leads to the following requirements

$$s''_1(x_1) = 0, \qquad s''_{i-1}(x_i) = s''_i(x_i),\ i = 2, \ldots, n-1, \qquad s''_{n-1}(x_n) = 0.$$

- With proposition 4.3 (*second derivatives at extremities*) stated in chapter on *Hermite Interpolation*, these conditions can be expressed as follows

$$\begin{aligned} 2\,y'_1 + y'_2 &= \frac{3}{h}\,(y_2 - y_1) \\ y'_{i-1} + 4\,y'_i + y'_{i+1} &= \frac{3}{h}\,(y_{i+1} - y_{i-1}) \quad \text{for } i = 2, \ldots, n-1, \\ y'_{n-1} + 2\,y'_n &= \frac{3}{h}\,(y_n - y_{n-1}) \end{aligned}$$

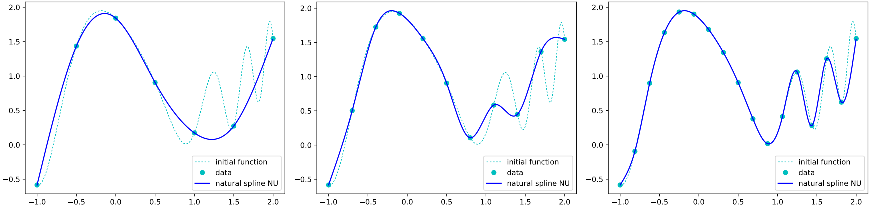
Uniform case (3)

- Finally, these conditions lead to the following tri-diagonal linear system whose unknowns are the derivatives y_i' :

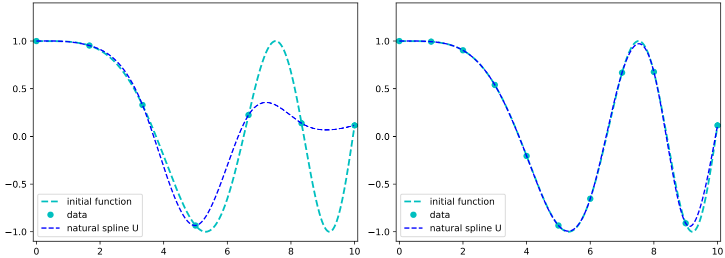
$$\begin{pmatrix} 2 & 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 1 & 4 & 1 & 0 & & & & \vdots \\ 0 & 1 & 4 & 1 & 0 & & & \vdots \\ \vdots & 0 & 1 & 4 & 1 & 0 & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & 0 & 1 & 4 & 1 & 0 \\ \vdots & & & & 0 & 1 & 4 & 1 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} y_1' \\ y_2' \\ y_3' \\ \vdots \\ \vdots \\ y_{n-2}' \\ y_{n-1}' \\ y_n' \end{pmatrix} = \frac{3}{h} \begin{pmatrix} y_2 - y_1 \\ y_3 - y_1 \\ y_4 - y_2 \\ \vdots \\ \vdots \\ y_{n-1} - y_{n-3} \\ y_n - y_{n-2} \\ y_n - y_{n-1} \end{pmatrix}$$

- The matrix of this $n \times n$ linear system is strictly diagonally dominant which ensures the existence and the uniqueness of a solution (*the uniform natural spline*), and thus concludes the proof of proposition 5.4 in the uniform case.
- The effective resolution of this tri-diagonal linear system can be achieved using *the Thomas algorithm* (see preliminaries).

Uniform case : examples



Interpolation of the function $f(x) = \sin(x^2 - 2x + 1) + \left[\cos(-x^3 - x)\right]^2$ *at 7, 11 and 17 uniform points.*



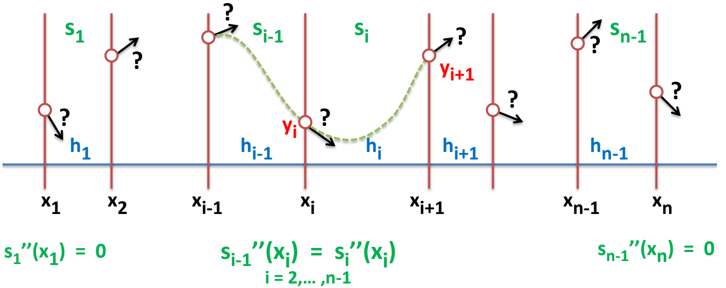
Interpolation of the function $f(x) = \cos(-x^2/9)$ *at 7 and 11 uniform points.*

Non uniform case (1)

We now consider the *non-uniform* case (the general case) of a strictly increasing sequence of nodes x_i with : $x_{i+1} - x_i = h_i, \ i = 1, \ldots, n-1$.

- The method is identical to that used in the uniform case.
- We consider natural end-conditions at points x_1 and x_n and C^2 contact at inner nodes

$$s_1''(x_1) = 0, \qquad s_{i-1}''(x_i) = s_i''(x_i), \ i = 2, \ldots, n-1, \qquad s_{n-1}''(x_n) = 0.$$



Non uniform case (2)

- C^2 contact at inner nodes : $s''_{i-1}(x_i) = s''_i(x_i)$ for $i = 2, \dots, n-1$

$$s''_{i-1}(x_i) = \frac{2}{h_{i-1}^2}\left(3\,y_{i-1} - 3\,y_i + 2\,h_{i-1}\,y'_i + h_{i-1}\,y'_{i-1}\right)$$

$$s''_i(x_i) = \frac{2}{h_i^2}\left(3\,y_{i+1} - 3\,y_i - 2\,h_i\,y'_i - h_i\,y'_{i+1}\right)$$

  lead to conditions

$$\boxed{h_i\,y'_{i-1} + 2\,(h_{i-1} + h_i)\,y'_i + h_{i-1}\,y'_{i+1} = 3\left(\frac{h_{i-1}}{h_i}\,y_{i+1} + \frac{h_i^2 - h_{i-1}^2}{h_i h_{i-1}}\,y_i - \frac{h_i}{h_{i-1}}\,y_{i-1}\right), \quad 2 \le i \le n-1}$$

- Natural end-condition in x_1 :

$$s''_1(x_1) = \frac{2}{h_1^2}\left(3\,y_2 - 3\,y_1 - 2\,h_1\,y'_1 - h_1\,y'_2\right) = 0$$

  leads to $\boxed{2\,y'_1 + y'_2 = \frac{3}{h_1}\,(y_2 - y_1)}$

- Natural end-condition in x_n :

$$s''_{n-1}(x_n) = \frac{2}{h_{n-1}^2}\left(3\,y_{n-1} - 3\,y_n + 2\,h_{n-1}\,y'_n + h_{n-1}\,y'_{n-1}\right) = 0$$

  leads to $\boxed{y'_{n-1} + 2\,y'_n = \frac{3}{h_{n-1}}\,(y_n - y_{n-1})}$

Non uniform case (3)

- Finally, these conditions lead to the following tri-diagonal linear system whose unknowns are the derivatives y_i'

$$\begin{pmatrix} 2 & 1 & 0 & \cdots & \cdots & \cdots & 0 & 0 \\ h_2 & 2(h_1+h_2) & h_1 & 0 & & & & 0 \\ 0 & h_3 & 2(h_2+h_3) & h_2 & 0 & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & & & \vdots \\ \vdots & & 0 & h_i & 2(h_{i-1}+h_i) & h_{i-1} & 0 & \vdots \\ \vdots & & & & \ddots & \ddots & \ddots & 0 \\ 0 & & & & 0 & h_{n-1} & 2(h_{n-2}+h_{n-1}) & h_{n-2} \\ 0 & 0 & \cdots & \cdots & \cdots & 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} y_1' \\ y_2' \\ y_3' \\ \vdots \\ y_{n-2}' \\ y_{n-1}' \\ y_n' \end{pmatrix} = \cdots$$

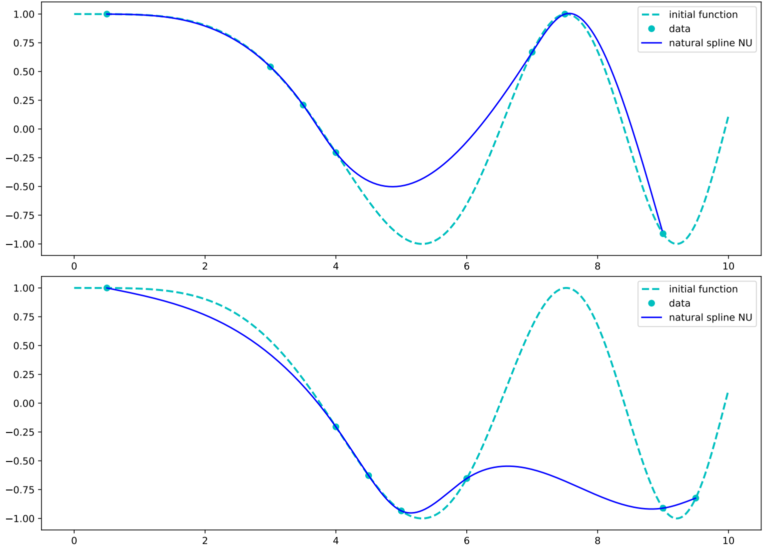
Non uniform case (4)

- Finally, these conditions lead to the following tri-diagonal linear system whose unknowns are the derivatives y_i'

$$
\cdots = 3 \begin{pmatrix} \frac{1}{h_1}(y_2 - y_1) \\ \frac{h_1}{h_2}(y_3 - y_2) + \frac{h_2}{h_1}(y_2 - y_1) \\ \vdots \\ \frac{h_{i-1}}{h_i}(y_{i+1} - y_i) + \frac{h_i}{h_{i-1}}(y_i - y_{i-1}) \\ \vdots \\ \frac{h_{n-2}}{h_{n-1}}(y_n - y_{n-1}) + \frac{h_{n-1}}{h_{n-2}}(y_{n-1} - y_{n-2}) \\ \frac{1}{h_{n-1}}(y_n - y_{n-1}) \end{pmatrix} .
$$

- Again, the matrix of this $n \times n$ linear system is strictly diagonally dominant which ensures the existence and the uniqueness of a solution (the *non-uniform natural spline*) and thus concludes the proof of proposition 5.4 in the non-uniform case .
- And again, the effective resolution of this tri-diagonal linear system can be achieved using *the Thomas algorithm* (see preliminaries).

Non-uniform case : examples

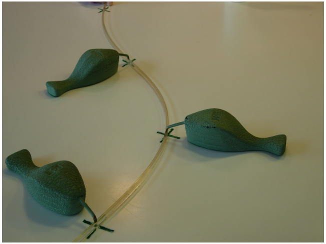


Interpolation of the function $f(x) = \cos(-x^2/9)$ at 7 non-uniform points.

Natural splines — Minimization property

Introduction

Spline originally designates a *flexible wood or plastic lath* used to generate geometric shapes (boat hull, aircraft fuselage, ...).
Such a slat, constrained to pass through frictionless "forced points" (interpolation points), assumes a smooth shape which minimizes its deformation energy (bending energy).



https ://secure.boeingimages.com/archive/

This *bending energy* can be modeled as being proportional to the *integral of the square of the curvature over the whole lath*, which thus requires the shape be of " class C^2 ".

The lath (or spline) therefore seeks to minimize its curvature at each point in order to be as "linear" as possible.

Minimization property

Thus, if Ω_2 denotes the set of C^2 real functions defined on $[a,b]$, and interpolating the data (x_i, y_i) for $i = 1, \ldots, n$, the lath takes the shape defined by the graph of a function $h_0 \in \Omega_2$ and characterized by

$$E(h_0) = \min_{h \in \Omega_2} E(h) \qquad \text{with} \quad E(h) = \int_a^b \Bigg| \underbrace{\frac{h''(x)}{\left[1 + h'(x)^2\right]^{3/2}}}_{\textit{curvature}} \Bigg|^2 \mathrm{d}x \tag{1}$$

If we assume that the deformations of the slat remain small enough to neglect the term $h'(x)^2$ compared to 1 in the denominator of the curvature, the problem simplifies as the minimization of the energy

$$E_1(h) = \int_a^b h''(x)^2 \, \mathrm{d}x \tag{2}$$

Proposition 5.5 (minimization property)

Let $s \in S$ be the unique natural spline interpolating data (x_i, y_i), $i = 1, \ldots, n$. Then

$$\forall\, h \in \Omega_2, \quad E_1(s) \leq E_1(h).$$

In other words, among all functions h of $C^2[a,b]$ satisfying the interpolation conditions $h(x_i) = y_i$, the natural spline function minimizes the energy $E_1(h)$.

Proof (minimization property)

The natural interpolating spline s is clearly in Ω_2.
Then, given $h \in \Omega_2$, we write $h = s + (h - s)$, so that

$$\int_a^b (h'')^2 = \int_a^b (s'')^2 + 2\int_a^b (s'')(h'' - s'') + \int_a^b (h'' - s'')^2,$$

where the variable x and the differential element $\mathrm{d}x$ *are omitted for readability.*
Integrating by parts the second integral of the right side, we get

$$\int_a^b (s'')(h''-s'') = \left[s''(h'-s')\right]_a^b - \int_a^b (s''')(h'-s') = -\int_a^b (s''')(h'-s') \quad \text{as } s''(a) = s''(b) = 0.$$

Moreover, the third derivative of s is piecewise constant. Denoting by s_i''' its value on $]x_i, x_{i+1}[$, we get

$$\int_a^b (s''')(h' - s') = \sum_{i=1}^{i=n-1} s_i''' \left[h - s\right]_{x_i}^{x_{i+1}} = 0$$

because of the interpolating conditions at each node x_i. Finally :

$$\int_a^b (s'')(h'' - s'') = 0 \quad \text{from which we deduce} \quad \int_a^b (h'')^2 \geq \int_a^b (s'')^2.$$

Calculus of variations

Calculus of variations is a field of mathematical analysis that uses variations, which are small changes in functions and functionals, to find maxima and minima of functionals such as the energy E_1 defined by relation (2).

A major interest of the functional approach is to allow generalizations to other types of splines.

We first give *Euler* [1]*-Lagrange equations of order one and order two* (without any proof) which are essential in such minimization problems. Then we will consider the application to *cubic splines* and to *exponential splines*.

Some applications of the calculus of variations include : (*from wikipedia*)

- The derivation of the catenary shape
- Newton's minimal resistance problem
- The brachistochrone problem
- Isoperimetric problems
- Geodesics on surfaces
- Minimal surfaces and Plateau's problem
- Optimal control

1. Leonhard Euler, 1707-1783, Swiss mathematician

Euler-Lagrange equations of order 1

Let $\alpha, \beta \in \mathbb{R}$ with $\alpha < \beta$ and let us denote $W_2 = W_2(\alpha, \beta, y_\alpha, y_\beta)$ the set of real functions $y \in C^2([\alpha, \beta])$ such that $y(\alpha) = y_\alpha$ and $y(\beta) = y_\beta$, where y_α, y_β are fixed real values. We consider the functional J defined on W_2 by

$$J(y) = \int_\alpha^\beta F\big(x, y(x), y'(x)\big)\,\mathrm{d}x$$

where $F : (x, y, y') \mapsto F(x, y, y')$ is a given C^2 function from $\mathbb{R}^3$ to $\mathbb{R}$.

A necessary condition for a function y in W_2 to realize the minimum of the functional J, is that y satisfy the equation

$$\frac{\partial F}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{\partial F}{\partial y'}\right) = 0, \tag{3}$$

named *Euler-Lagrange equation of order 1*.

Equation (3) can be more precisely written as follows

$$\frac{\partial F}{\partial y}\big(x, y(x), y'(x)\big) - \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{\partial F}{\partial y'}\big(x, y(x), y'(x)\big)\right) = 0, \quad \forall x \in (\alpha, \beta),$$

where $\frac{\partial F}{\partial x}$, $\frac{\partial F}{\partial y}$, $\frac{\partial F}{\partial y'}$, represent partial derivatives of F with respect to each of its three independent real variables x, y, y'.

Example 1 : geodesic

Consider the classical problem of finding the shortest path between two points (α, y_α) and (β, y_β). For a curve joining these two points and defined by $y \in W_2$, the functional expressing its length is here

$$J(y) = \int_\alpha^\beta \sqrt{1 + y'(x)^2}\,\mathrm{d}x.$$

The function $F(x, y, y') = \sqrt{1 + y'^2}$ is of class C^2 and

$$\frac{\partial F}{\partial y} = 0, \qquad \frac{\partial F}{\partial y'} = \frac{y'}{\sqrt{1 + y'^2}}\,.$$

The Euler-Lagrange equation is thus here

$$\frac{\mathrm{d}}{\mathrm{d}x}\Big(\frac{y'(x)}{\sqrt{1 + y'(x)^2}}\Big) = 0\,,$$

which means that function $\dfrac{y'}{\sqrt{1 + y'^2}}$ is constant, so that function y' is itself constant, and y is affine, which leads to the unique solution

$$y(x) = y_\alpha + (x - \alpha)\,\frac{y_\beta - y_\alpha}{\beta - \alpha}$$

Example 2 : longitudinal tension

For later purposes, consider the simple problem of minimizing the functional

$$J(y) = \int_{\alpha}^{\beta} \Big(y'(x) \Big)^2 \, \mathrm{d}x. \tag{4}$$

The function $F(x, y, y') = \Big(y'(x) \Big)^2$ is of class C^2 and the associated Euler-Lagrange equation is

$$\frac{\mathrm{d}}{\mathrm{d}x} \Big(2\, y'(x) \Big) = 2\, y''(x) = 0\,.$$

We deduce that y' is constant so that the solution is (again) an affine segment between points (α, y_α) an (β, y_β).

Euler-Lagrange equations of order 2

With the previous notations, consider now the set W_4 of real functions $y \in C^4([\alpha, \beta])$ such that $y(\alpha) = y_\alpha$, $y'(\alpha) = y'_\alpha$ and $y(\beta) = y_\beta$, $y'(\beta) = y'_\beta$, where $y_\alpha, y'_\alpha, y_\beta, y'_\beta$ are fixed real values. We then consider the functional J_2 defined on W_4 by

$$J_2(y) = \int_\alpha^\beta F\big(x, y(x), y'(x), y''(x)\big)\,\mathrm{d}x$$

where F is now a function of class C^3 on $\mathbb{R}^4$.

A necessary condition for a function y in W_4 to realize the minimum of the functional J_2, is that y satisfy the equation

$$\frac{\partial F}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{\partial F}{\partial y'}\right) + \frac{\mathrm{d}^2}{\mathrm{d}x^2}\left(\frac{\partial F}{\partial y''}\right) = 0, \tag{5}$$

named *Euler-Lagrange equation of order 2*.

Variation approach for C^2 cubic splines

Consider now the problem of minimizing the energy $E_1(h) = \int_a^b h''(x)^2 \, \mathrm{d}x$, on the set of functions of $C^2([a,b])$, interpolating data (x_i, y_i) and of class C^4 on each interval (x_i, x_{i+1}).
The function $F(x, h, h', h'') = (h'')^2$ is of class C^3 and

$$\frac{\partial F}{\partial h} = 0, \qquad \frac{\partial F}{\partial h'} = 0\,, \qquad \frac{\partial F}{\partial h''} = 2\,h''\,.$$

The Euler-Lagrange equation of order 2, associated with the functional E_1, is thus here

$$\frac{\mathrm{d}^2}{\mathrm{d}x^2}\Big(2\,h''(x)\Big) \;=\; 2\,h^{(4)}(x) = 0,$$

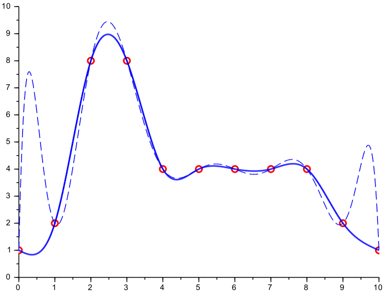
which shows that the restriction of h to each interval $[x_i, x_{i+1}]$ must be a polynomial of degree ≤ 3.
Consequently, the process is as follows (and is similar to the algebraic approach).

- ⋄ Constraints $y_i = h(x_i)$ are prescribed at each node x_i : they are the interpolating data.
- ⋄ Given n real values $y'_1, \ldots, y'_n$, there exists a unique function $h \in \Pi_1^3$, such that $h'(x_i) = y'_i$, $1 \leq i \leq n$.
- ⋄ We then determine the real values y'_i so that the function h is actually of class C^2, with second derivatives equal to zero at extremities (in case of natural splines).

Note that *cubic polynomials* are chosen “a priori” in the algebraic approach and deduced “a posteriori” in the variational approach.

Smoothing and tension

A close inspection of the C^2 cubic interpolating spline given in introduction (see right, the solid blue line) shows slight oscillations which are inherent in the nature of this spline and which also depend on the repartition of the data points to be interpolated.



As an example, imagine a car forced to pass through imposed points at a given speed must anticipate its turns and can be thus constrained to take large turns.

However, by reducing and adjusting its speed, the driver will be able to take shorter turns thus decreasing the undesirable oscillations and the length of its path. In this situation, the driver will therefore seek the trajectory allowing to minimize the energy

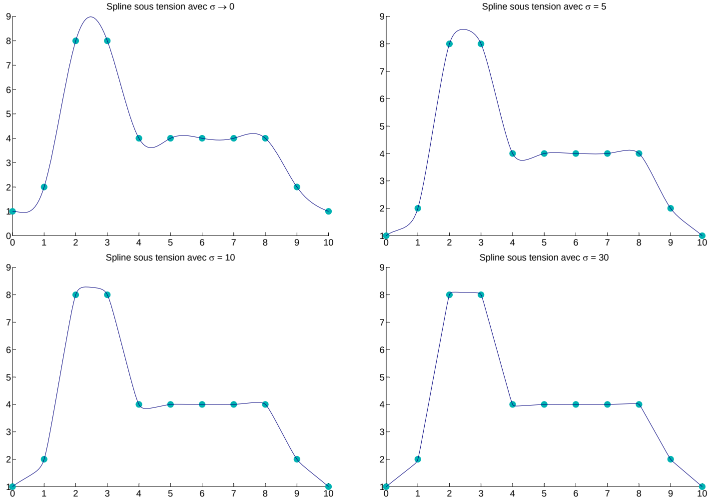
$$E_2(h) = \int_a^b \left[h''(x)^2 + \sigma^2 \, h'(x)^2\right] \mathrm{d}x = \int_a^b h''(x)^2 \, \mathrm{d}x + \sigma^2 \int_a^b h'(x)^2 \, \mathrm{d}x \qquad (6)$$

with $\sigma > 0$.

The first term represents an approximation of the bending energy while the second term induces the trajectory to be as short as possible (i.e., affine) between interpolation points (see example 2). Parameter σ behaves as a *tension parameter*.

Examples

The *tension parameter* σ controls (indirectly) the length of the curve.



For $\sigma \to 0$, we get a C^2 cubic interpolating spline.
For $\sigma \to +\infty$, we get a continuous piecewise linear spline.

General solution

The function $F(x,h,h',h'') = (h'')^2 + \sigma^2 (h')^2$ is of class C^3 and the Euler-Lagrange equation of order 2 associated with the functional E_2 is expressed by

$$-\frac{\mathrm{d}}{\mathrm{d}x}\Big(2\,\sigma^2\,h'(x)\Big) + \frac{\mathrm{d}^2}{\mathrm{d}x^2}\Big(2\,h''(x)\Big) = -2\,\sigma^2\,h''(x) + 2\,h^{(4)}(x) = 0,$$

which leads to the differential equation

$$h^{(4)} - \sigma^2\,h'' = 0. \tag{7}$$

On each interval $[x_i, x_{i+1}]$, the solution h_i is thus on the form

$$h_i(x) = \lambda_0^i + \lambda_1^i\,x + \lambda_2^i\,\exp(\sigma\,x) + \lambda_3^i\,\exp(-\sigma\,x), \tag{8}$$

which provides $4(n-1)$ free coefficients $\lambda_k^i,\ k=0,\ldots,3,\ i=1,\ldots,n-1$, allowing to satisfy the $3(n-2)$ C^2 contact conditions at each inner node $x_i,\ i=2,\ldots,n-1$, the end-conditions $h''(x_1) = 0 = h''(x_n)$, together with the n interpolating constraints $h(x_i) = y_i,\ i=1,\ldots,n.$

Finally, for each $\sigma > 0$, there exists a unique C^2 function h, of class C^4 on each interval (x_i, x_{i+1}), called *spline in tension* or *exponential spline*, interpolating the initial data. These splines have been introduced by Daniel Schweikert in 1966.

Local tension

Parameter σ influences the entire curve, that can be viewed as a disadvantage. Nevertheless, *local control* over the smoothing can be accomplished by considering the following energy to be minimized.

$$E_{l,2}(h) \;=\; \int_a^b \; h''(x)^2 \, \mathrm{d}x \;+\; \sum_{i=1}^{n-1} \sigma_i^2 \; \int_{x_i}^{x_{i+1}} \; h'(x)^2 \, \mathrm{d}x \tag{9}$$

with *tension parameters* $\sigma_i > 0$ associated with each interval.

Implementation

Considering this energy $E_{l,2}(h)$, the solution $h_i(x)$ on interval $[x_i, x_{i+1}]$ should be rewritten in the following form which has better numerical properties

$$\hat{h}_i(u) = \mu_0^i \,(1-u) + \mu_1^i \, u + \mu_2^i \, \phi_i(1-u) + \mu_3^i \, \phi_i(u),$$

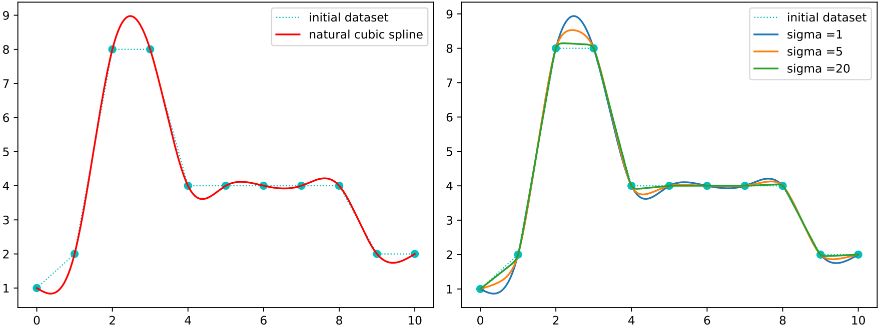
with

$$\phi_i(u) = \frac{\sinh(\omega_i \, u) - u \, \sinh(\omega_i)}{\sinh(\omega_i) - \omega_i}, \quad u = \frac{x - x_i}{h_i}, \quad h_i = x_{i+1} - x_i, \quad \omega_i = \sigma_i \, h_i,$$

where u serves as a local parameter.

Examples

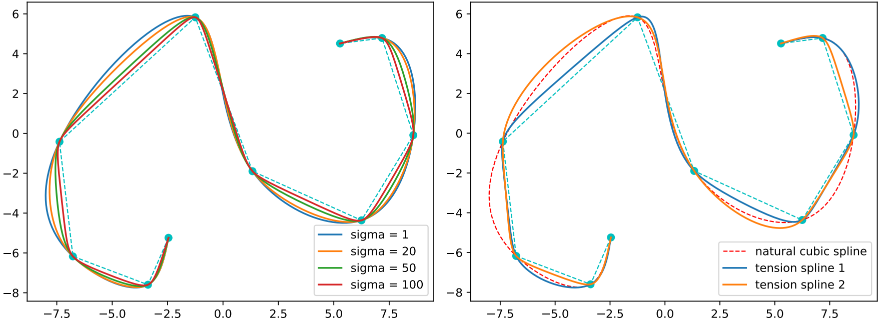
◇ *Explicite case*



Left : natural spline – Right : global tension parameter : σ_i = sigma for all i

Examples

◇ *Parametric case with chordal parameterization*



Left : global tension parameter σ_i = sigma for all i, respectively equal to 1, 20, 50, 100.
Right :

tension spline 1 : $\{\sigma_i\} = \{1, 1, 100, 50, 1, 50, 100, 1, 1\}$
tension spline 2 : $\{\sigma_i\} = \{50, 100, 100, 1, 1, 1, 100, 100, 50\}$
together with the (chordal) parametric natural spline.

Miscellaneous

Splines in tension have the major drawback of not being polynomial, which constitutes a real handicap for many applications, in particular in CAGD and geometrical modeling (in particular for numerical evaluation).

The ν-*splines* (nu-splines) introduced to remedy this disadvantage, are obtained by minimizing the following energy

$$E_\nu(h) \;=\; \int_a^b h''(x)^2 \,\mathrm{d}x \;+\; \sum_{i=0}^{n} \nu_i \left[h'(x_i)\right]^2 \tag{10}$$

where coefficients ν_i are non negative and behave as tension parameters applied only at interpolation points x_i. The solution is C^1, piecewise polynomial cubic, with continuous curvature at inner nodes (so that the solution is almost C^2).

The story of splines does not stop there. Many other kinds of splines have been introduced : β-splines, γ-splines, PH-splines, B-splines [2] (Basis splines),... What made Tom Lyche (university of Oslo) to say : ≪ *There's always more room in the spline Zoo !* ≫.

2. For French readers : note that contrary to a widespread idea, B is not here the initial of Baudelaire (French poet).

The tridiagonal spline system

Construction of uniform or non uniform natural splines leads to solve a $n \times n$ *tri-diagonal linear system*. The matrix A of this tri-diagonal linear system is strictly diagonally dominant which ensures the existence and the uniqueness of a solution.

$$A = \begin{pmatrix} a_1 & c_1 & 0 & \cdots & 0 & 0 \\ b_2 & a_2 & c_2 & 0 & & 0 \\ & b_3 & a_3 & c_3 & 0 & \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & & & b_{n-1} & a_{n-1} & c_{n-1} \\ 0 & 0 & \cdots & & b_n & a_n \end{pmatrix}$$

We consider here the effective calculation of a solution.

A natural way to proceed with such a tridiagonal system is to perform a LU decomposition of the matrix A.

We develop here the general case of an arbitrary tri-diagonal matrix A for which the LU decomposition simplifies in the Thomas algorithm, with a cost of $5n$ operations (multiplication or division).

We will then consider at the end of this chapter (in the form of a problem) the case associated with uniform splines, for which the application of the classical Gaussian elimination leads to a cost of $4n$ operations.

LU decomposition and Thomas algorithm (1)

Consider the LU factorization of the following tridiagonal matrix A which is assumed to be non singular.

$$A = \begin{pmatrix} a_1 & c_1 & 0 & & \cdots & 0 & 0 \\ b_2 & a_2 & c_2 & 0 & & & 0 \\ & b_3 & a_3 & c_3 & 0 & & \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \vdots \\ & & & \ddots & \ddots & \ddots & 0 \\ 0 & & & & b_{n-1} & a_{n-1} & c_{n-1} \\ 0 & 0 & & \cdots & 0 & b_n & a_n \end{pmatrix}$$

We thus have to determine two bidiagonal matrices on the form

$$L = \begin{pmatrix} 1 & 0 & & \cdots & 0 & 0 \\ v_2 & 1 & 0 & & & 0 \\ & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & & \\ 0 & & & v_{n-1} & 1 & 0 \\ 0 & 0 & \cdots & & v_n & 1 \end{pmatrix} \text{ and } U = \begin{pmatrix} u_1 & c_1 & & \cdots & 0 & 0 \\ 0 & u_2 & c_2 & & & 0 \\ & & \ddots & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & 0 \\ 0 & & & 0 & u_{n-1} & c_{n-1} \\ 0 & 0 & \cdots & & 0 & u_n \end{pmatrix},$$

with unknown coefficients v_i $(2 \leq i \leq n)$ and u_j $(1 \leq j \leq n)$.
Since the matrix A is non singular, coefficients u_j are non zero.

LU decomposition and Thomas algorithm (2)

A direct identification leads to

$$u_1 = a_1 \quad \text{and} \quad \begin{cases} v_i = \dfrac{b_i}{u_{i-1}} \\ u_i = a_i - v_i\, c_{i-1} \end{cases} \qquad i = 2, \ldots, n. \tag{11}$$

Then, the resolution of the tridiagonal linear system $A\,x = b$ is equivalent to the resolution of the two bidiagonal linear systems $L\,y = b$ and $U\,x = y$. Precisely :

$$\text{system} \quad L\,y = b : \quad y_1 = b_1 \quad \text{and} \quad y_i = b_i - v_i\, y_{i-1}\,, \quad i = 2, \ldots, n \tag{12}$$

$$\text{system} \quad U\,x = y : \quad x_n = \frac{y_n}{u_n} \quad \text{and} \quad x_i = \frac{y_i - c_i\, x_{i+1}}{u_i}\,, \quad i = n-1, n-2, \ldots, 1 \tag{13}$$

Finally, the cost is of $2\,n - 2$ operations (multiplication or division) for the factorization (11) and of $3\,n - 3$ operations for the resolution of the two systems (12) and (13), which leads to an approximate total of $5\,n$ operations.

Objective

The objective and the process are similar to those developed in the case of parametric polynomial interpolation.
Given a sequence of points $M_i = (x_i, y_i)$, $1 \leq i \leq N$, we look for a parametric cubic spline curve which interpolates these data points, that is which goes through each point (x_i, y_i) for some parameter t_i. An important question consists in the choice of the interpolation parameters t_i, which are also called the *interpolation nodes* (or knots) or simply the nodes. In the situation of parametric spline interpolation, *the interpolation nodes will coincide with the spline nodes*.

Precisely, we look for a cubic parametric spline curve

$$s \; : \; \begin{array}{ccc} [a,b] \in \mathbb{R} & \longrightarrow & \mathbb{R}^2 \\ t & \longmapsto & s(t) = \begin{pmatrix} s_x(t) \\ s_y(t) \end{pmatrix} \end{array}$$

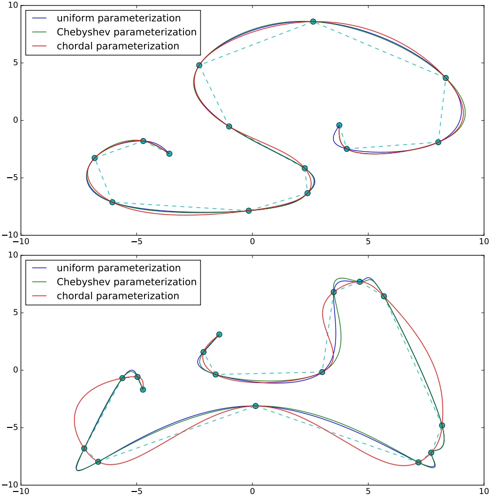
such that

$$s(t_i) \; = \; M_i \quad \Leftrightarrow \quad \left\{ \begin{array}{l} s_x(t_i) = x_i \\ s_y(t_i) = y_i \end{array} \right. , \quad 1 \leq i \leq N$$

with a sequence of nodes $a = t_1 < t_2 < \cdots < t_N = b$, and where $s_x(t)$ and $s_y(t)$ are C^2 cubic interpolating splines.
As a result, we are reduced to solve two separate non-uniform spline interpolation problems.

Examples

As in the case of parametric polynomial interpolation we consider the uniform, the Chebyshev and the chordal parameterizations.



Exercise 5.1 — *Energy evaluation –*
On note Ω_2 l'ensemble des fonctions réelles définies sur $[a, b]$, de classe C^2, et interpolant les données (x_i, y_i) pour $i = 1, 2, 3$ définies par

$$\begin{aligned} a = x_1 = -1, \qquad & x_2 = 0, \qquad & x_3 = 1 = b, \\ y_1 = 0, \qquad & y_2 = 1, \qquad & y_3 = -2. \end{aligned}$$

Déterminer

$$e_2 = \min_{h \in \Omega_2} \left(\int_a^b h''(x)^2 \, \mathrm{d}x \right)$$

Exercise 5.2 —
Soit $[a, b]$ un intervalle $(a < b)$, E un sous espace vectoriel de $C^0([a, b])$ ainsi qu'une subdivision (x_i) strictement croissante de l'intervalle $[a, b]$:

$$a = x_0 < x_1 < x_2 < \cdots < x_n < x_{n+1} = b$$

On considère par ailleurs n nombres réels quelconques :

$$y_1, y_2, \ldots, y_n$$

Est-il vrai que si $\dim E \geq n$, il est toujours possible de trouver une fonction $f \in E$ vérifiant

$$f(x_i) = y_i, \quad i = 1, 2, \ldots, n$$

Exercise 5.3 — *Towards B-splines* – (1/2)

On considère la subdivision suivante de l'intervalle $[-3, 3]$:

$$x_0 = -3,\ x_1 = -2,\ x_2 = -1,\ x_3 = 0,\ x_4 = 1,\ x_5 = 2,\ x_6 = 3,$$

et on étudie certaines splines d'interpolation aux points x_1, x_2, x_3, x_4, x_5. Les splines considérées seront toutes cubiques C^2 et on désignera par $f_i(x)$ la restriction de la spline à l'intervalle $[i, i+1]$.

Première partie.

On souhaite montrer qu'il n'existe pas de spline naturelle d'interpolation (cubique C^2) σ, non identiquement nulle, telle que

$$\sigma(x) = 0 \ \text{ pour } \ x \in [-2, -1] \cup [1, 2].$$

1. Montrer que pour une telle spline les polynômes cubiques $f_{-1}(x)$ et $f_0(x)$ admettent chacun une racine triple, et en déduire l'expression générale de ces deux polynômes (dépendant chacun d'un paramètre à déterminer).
2. Ecrire les conditions de raccord C^2 en x_3.
3. Conclure.

Exercise (suite) — *Towards B-splines* – (2/2)

Deuxième partie.

On souhaite maintenant déterminer l'ensemble $\mathcal{S}$ des splines naturelles ***paires*** σ, telles que

$$\sigma(x) = 0 \text{ pour } x \in [-3,-2] \cup [2,3].$$

4. Vérifier que $\mathcal{S}$ est un espace vectoriel.
5. Pour des paramètres réels α et β donnés, peut-on dire que l'unique spline naturelle d'interpolation associée aux données d'interpolation

$$(-2,0),\ (-1,\alpha),\ (0,\beta),\ (1,\alpha),\ (2,0)$$

appartient à $\mathcal{S}$? Justifier votre réponse.

Soit maintenant un élément σ de $\mathcal{S}$ prenant la valeur α en -1 et 1 (parité oblige) et la valeur β en 0, et $f_0(x)$ et $f_1(x)$ ses restrictions à $[0,1]$ et $[1,2]$.

6. En s'inspirant de la méthode de la question 1 de la première partie, déterminer le polynôme cubique $f_1(x)$ en fonction du paramètre α.
7. Déduire des hypothèses la valeur de $f_0'(0)$.
8. Déterminer le polynôme cubique $f_0(x)$ en fonction des paramètres α (et β). *Il sera plus simple de chercher f_0 sous la forme* $f_0(x) = a + b(x-1) + c(x-1)^2 + d(x-1)^3$.
9. En déduire une relation entre α et β et expliciter la construction de σ.
10. Caractériser l'ensemble $\mathcal{S}$ à l'aide des splines naturelles d'interpolation.
11. Determine the unique element $\hat{\sigma}$ in $\mathcal{S}$ such that $\int_{-3}^{3} \hat{\sigma}(x)\,dx = 1$.

Exercise 5.4 — *Periodic C2 cubic splines* – (1/2)
Etant donnée une suite de n points x_i $(n \geq 2)$ tels que

$$a = x_1 < x_2 < \cdots < x_n = b,$$

on considère l'ensemble E des fonctions de $\mathcal{C}^2(\mathbb{R})$, périodiques, de période $T = b - a$, dont la restriction à chaque intervalle $[x_i, x_{i+1}]$ est un polynôme de degré 3.
On considère par ailleurs l'espace $\Pi_2^3 = \Pi_2^3(x_1, ..., x_n)$ des splines cubiques C^2 associées à cette famille de noeuds x_i. On rappelle que Π_2^3 est l'ensemble des fonctions de classe C^2 sur $[a, b]$ dont la restriction à chaque intervalle $[x_i, x_{i+1}]$, $i = 1, 2, ..., n-1$ est un polynôme de degré 3.

1. Vérifier que E est un espace vectoriel.
2. On considère le cas $n = 2$ avec $a = 0$ et $b = 1$. Expliciter l'espace E. Donner sa dimension et une base.
3. On se place désormais dans le cas général avec $n > 2$. Pour un élément $f \in E$, on note $\hat{f}$ sa restriction à l'intervalle $[a, b]$ et on considère l'ensemble $\hat{E} = \{\hat{f}, f \in E\}$. Vérifier que $\hat{E}$ est un sous espace vectoriel de Π_2^3.
4. Soit $s \in \Pi_2^3$. Ecrire les conditions nécessaires et suffisantes pour que $s \in \hat{E}$.

Exercise (suite) — *Periodic C2 cubic splines* – (2/2)
On rappelle que tout élément s de Π_2^3 s'écrit de manière unique sous la forme

$$s(x) = \sum_{i=0}^{3} \alpha_i x^i + \sum_{i=2}^{n-1} \beta_i (x - x_i)_+^3, \quad x \in [a, b].$$

5. Montrer que $\hat{E}$ est le noyau d'une application linéaire Φ de Π_2^3 dans $\mathbb{R}^3$.
6. En identifiant Π_2^3 avec $\mathbb{R}^{n+2}$, écrire la matrice de $\Phi : \mathbb{R}^{n+2} \to \mathbb{R}^3$, déterminer son rang et en déduire la dimension de $\hat{E}$.

Interpolation. – On considère maintenant une suite de n réels $y_1, y_2, \ldots, y_n$ et on souhaite interpoler les données (x_i, y_i), $i = 1, \ldots, n$ par une fonction de $\hat{E}$. Autrement dit, on cherche $\hat{f} \in \hat{E}$ tel que

$$\hat{f}(x_i) = y_i, \quad i = 1, 2, \ldots, n.$$

7. Les données y_i peuvent-elles toutes être choisies indépendamment ? On suppose

   désormais que les données y_i sont cohérentes au sens de la question précédente et que les noeuds x_i sont équirépartis dans $[a, b]$, autrement dit que $x_{i+1} - x_i = h$, $i = 1, \ldots, n-1$ avec $h = (b-a)/(n-1)$. On propose de construire l'interpolant $\hat{f}$ selon une méthode semblable à celle des splines naturelles.
8. Quelles sont les inconnues à déterminer ?
9. Ecrire le système linéaire permettant de déterminer ces inconnues.

Exercise 5.5 — *Clamped splines* –
On considère ici le problème d'interpolation spline cubique C^2 où la condition de nullité de la dérivée seconde aux extrémités (condition des splines naturelles) est remplacée par une contrainte sur la dérivée première aux extrémités. On se place dans le cas d'une suite de nœuds équidistants

$$a = x_1 < x_2 < \cdots < x_{n-1} < x_n = b,$$

avec $n \geq 2$, $x_{i+1} - x_i = h,\ i = 1, \ldots, n-1$, et on considère l'ensemble $E \subset \Pi_2^3(x_1, ..., x_n)$ des fonctions de classe C^2 sur l'intervalle $[a, b]$ dont la restriction à chaque intervalle $[x_i, x_{i+1}]$ est un polynôme de degré inférieur ou égal à 3, et dont la dérivée est nulle aux deux extrémités a et b.
On rappelle que $1, x, x^2, x^3, \{(x - x_i)_+^3\}_{i=2,\ldots,n-1}$ est une base de $\Pi_2^3 = \Pi_2^3(x_1, ..., x_n)$.

1. Vérifier que E est un espace vectoriel.
2. Déterminer précisément la dimension de E. On introduira pour cela une application linéaire Φ de Π_2^3 à valeurs dans $\mathbb{R}^2$.

On donne maintenant n valeurs réelles $y_1, y_2, ..., y_n$ ainsi que deux autres réels y_a' et y_b'.

3. Modifier la méthode développée dans le cours sur l'interpolation spline (cas des splines naturelles) afin de déterminer (si cela est possible) un élément s de Π_2^3 vérifiant

$$s'(a) = y_a', \qquad s(x_i) = y_i,\ i = 1, ..., n, \qquad s'(b) = y_b'.$$

   On précisera les inconnues et on écrira le système linéaire permettant de déterminer ces inconnues. Discuter de l'existence et l'unicité des solutions.

Exercise 5.6 — *C2 cubic spline with condition "not a knot"* – (1/2)
On considère ici le problème d'interpolation spline cubique C^2 où la condition de nullité de la dérivée seconde aux extrémités (condition des splines naturelles) est remplacée par la condition *"not a knot"* qui consiste à imposer un raccord de classe C^3 sur le deuxième nœud et sur l'avant dernier. On se place dans le cas uniforme.
Soit la famille suivante de nœuds équidistants

$$a = x_1 < x_2 < \cdots < x_{n-1} < x_n = b,$$

avec $n \geq 4,\ x_{i+1} - x_i = h,\ i = 1, \ldots, n-1$, et on considère l'ensemble $E \subset \Pi_2^3(x_1, ..., x_n)$ des fonctions de Π_2^3 qui sont de classe C^3 aux nœuds x_2 et x_{n-1}.

1. Vérifier que E est un espace vectoriel.
2. Soit $s(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \sum_{i=2}^{n-1} b_i (x - x_i)_+^3$ un élément Π_2^3. On note s_i la restriction de s à l'intervalle $[x_i, x_{i+1}]$.
   Donner l'expression de chacun des polynômes $s_1(x), s_2(x), s_{n-2}(x), s_{n-1}(x)$, ainsi que l'expression des dérivées troisième $s_1'''(x), s_2'''(x), s_{n-2}'''(x), s_{n-1}'''(x)$.
3. En déduire les conditions sur les coefficients a_k et b_i pour que s appartiennent à l'ensemble E.
4. Déterminer la dimension de E. On introduira pour cela une application linéaire φ de $\mathbb{R}^{n+2}$ à valeurs dans $\mathbb{R}^2$.

Exercise (suite) — *C2 cubic spline with condition "not a knot"* – (2/2)

Résultat intermédiaire. Soit p le polynôme d'interpolation cubique de Hermite associé aux données $\left(\alpha, y_\alpha, y'_\alpha\right)$ et $\left(\beta, y_\beta, y'_\beta\right)$ et posons $h = \beta - \alpha$. *On notera que la dérivée troisième de p est un polynôme constant.*

5. Déterminer cette dérivée troisième de p uniquement en fonction des données d'interpolation $\alpha, y_\alpha, y'_\alpha$ et β, y_β, y'_β.

   Indication : on écrira (comme dans le cours) le polynôme d'interpolation de Hermite p sous la forme d'un dévelopement de Taylor en α, ainsi que sa dérivée :

$$p(x) = y_\alpha + (x-\alpha)\, y'_\alpha + \frac{(x-\alpha)^2}{2}\, p''(\alpha) + \frac{(x-\alpha)^3}{6}\, p'''(\alpha),$$

$$p'(x) = y'_\alpha + (x-\alpha)\, p''(\alpha) + \frac{(x-\alpha)^2}{2}\, p'''(\alpha),$$

   et on évaluera ces deux polynômes en $x = \beta$.

Construction. On donne maintenant n valeurs réelles $y_1, y_2, \ldots, y_n$.

6. Modifier la méthode développée dans le cours pour l'interpolation spline cubique C^2 dans le cas des splines naturelles, afin de déterminer un élément s de E vérifiant

$$s(x_i) = y_i, \quad i = 1, \ldots, n\,.$$

   On remplacera donc la condition de nullité de la dérivée seconde aux extrémités par la condition de raccord C^3 en x_2 et x_{n-1}. On précisera les inconnues et on écrira le système linéaire permettant de déterminer ces inconnues.

Exercise 5.7 — *Tri-diagonal uniform spline linear system & Gauss elimination* – Considering the tri-diagonal linear system

$$\begin{pmatrix} 2 & 1 & 0 & \cdots & \cdots & 0 & 0 \\ 1 & 4 & 1 & & & & 0 \\ 0 & 1 & 4 & 1 & & & 0 \\ \vdots & & \ddots & \ddots & \ddots & & \vdots \\ 0 & & & 1 & 4 & 1 & 0 \\ 0 & & & & 1 & 4 & 1 \\ 0 & 0 & \cdots & \cdots & 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} y'_1 \\ y'_2 \\ y'_3 \\ \vdots \\ y'_{n-2} \\ y'_{n-1} \\ y'_n \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{n-2} \\ v_{n-1} \\ v_n \end{pmatrix}$$

we apply a standard gaussian elimination so as to get the following bidiagonal triangular system, where coefficients a_k and b_k are (unknown) integers.

$$\begin{pmatrix} a_1 & b_1 & 0 & & \cdots & 0 & 0 \\ 0 & a_2 & b_2 & 0 & & & 0 \\ & 0 & a_3 & b_3 & 0 & & \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \vdots \\ & & & \ddots & \ddots & \ddots & 0 \\ 0 & & & & 0 & a_{n-1} & b_{n-1} \\ 0 & 0 & & \cdots & & 0 & a_n \end{pmatrix} \begin{pmatrix} y'_1 \\ y'_2 \\ y'_3 \\ \vdots \\ y'_{n-2} \\ y'_{n-1} \\ y'_n \end{pmatrix} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_{n-2} \\ w_{n-1} \\ w_n \end{pmatrix}.$$

Exercise (suite) — *Tri-diagonal uniform spline linear system & Gauss elimination* –

1. Prove that coefficients a_k, b_k, w_k are determined as follows

$$\begin{cases} a_1 = 2,\ a_2 = 7, & a_{k+1} = 4\,a_k - a_{k-1}, \quad \text{for } k = 2, \dots, n-2, \\ & a_n = 2\,a_{n-1} - a_{n-2}, \\ b_1 = 1, & b_{k+1} = a_k, \quad \text{for } k = 1, \dots, n-2, \\ w_1 = v_1 & w_{k+1} = a_k\, v_{k+1} - w_k, \quad \text{for } k = 1, \dots, n-1. \end{cases}$$

2. Determine the total cost of this resolution approach.