Sorbonne Universités - Paris 6

UE 3m101 - « Projet », Parcours PIMA

Rapport de projet « Optimisation »

Calle Viera Andersson, Jeetoo Samir, Quiquemelle Xavier, Sanchez Jacobo

Encadré par
Marie Postel

Semestre 2 L3 - 2018

Table des matières

Introduction

Ce rapport a été rédigé dans le cadre de l'unité d'enseignement 3M101 - « Projet ». L'objectif principal pour cette UE consiste, pour chacun d'entre nous, à mener à bien un projet mathématique incluant une partie significative de programmation informatique encadré par des enseignants/chercheurs.

Ainsi, notre choix s'est porté sur l' « Optimisation du profil d'une route ». Dans ce projet, on optimisera le tracé d'une route de manière à respecter des contraintes d'utilisation tout en minimisant le coût de construction. Le profil du terrain étant donné sous la forme d'un relevé d'altitude en des points le long du tracé de la future route, il s'agit de modéliser le coût des déblais et des remblais à effectuer pour arriver à un profil de route donné, puis de minimiser ce coût tout en respectant des contraintes de pente maximum.

Partie 1 – Excursion en dimension infinie : l'optimisation du profil d'une route.

Dans cette première partie, on va imaginer que notre échantillon de relevé d'altitude est infini. On pose L la longueur de notre route.
Nos relevés d'altitudes sont définis par la fonction $g(x)$ pour $x \in [0, L]$ Notre but est de déterminer la fonction $u(x)$ pour $x \in [0, L]$ representant le profil de notre route. Ce profil doit bien évidemment respecter la contrainte de pente α tout en minisant le coût des aménagements nécessaires à sa construction.
On va donc supposer que ses coûts sont déterminés par la fonction suivante :

$$J(u) = \int_0^L \phi(u(x) - g(x))dx$$

avec $\phi(z)$ une fonction convexe et coercive. On pourra envisager deux cas :

- ➔ $\phi(z) = |z|$
- ➔ $\phi(z) = \frac{z^2}{2}$

Notre situation se résume donc au problème d'optimisation suivant :

$$\begin{cases} D_a = \{u \in \mathcal{C}([0, L]), \forall x, y \in [0, L], |u(x) - u(y)| \leq \alpha|x - y|\}, \\ \forall u \in D_a, J(u^\star) \leq J(u). \end{cases} \tag{1}$$

Convexité de D_a

On va tout d'abord montrer que l'ensemble D_a est convexe.
Rappelons la définition d'un ensemble convexe :
On dit qu'un ensemble C est convexe si $\forall x, y \in C$ et $\forall \lambda \in [0, 1]$

$$\lambda x + (1 - \lambda)y \in C$$

Prenons donc $u, v \in D_a$ et $\lambda \in [0, 1]$
Montrons que $\lambda u + (1 - \lambda)v \in D_a$

$$\begin{aligned} |\lambda u(x) + (1-\lambda)v(x) + (-\lambda u(y) - (1-\lambda)v(y))| &= |\lambda u(x) - \lambda u(y) + (1-\lambda)v(x) - (1-\lambda)v(y)| \\ &= |\lambda(u(x) - u(y)) + (1-\lambda)(v(x) - v(y))| \\ &\leq |\lambda(u(x) - u(y))| + |(1-\lambda)(v(x) - v(y))| \\ &\leq \lambda|(u(x) - u(y))| + (1-\lambda)|(v(x) - v(y))| \\ &\leq \lambda|x - y| + (1-\lambda)|x - y| \\ &\leq |x - y| \end{aligned}$$

Pour tout $\lambda \in [0, 1]$, pour tout $u, v \in D_a$: $\lambda u + (1 - \lambda)v \in D_a$
L'ensemble D_a est donc convexe.
Dans toute la suite, on supposera aussi que D_a est fermé.

Théorème de projection sur un convexe fermé

Sachant maintenant que D_a est convexe et fermé, on peut appliquer le théorème de projection sur un convexe fermé.
Soit $C \subset E$ un ensemble convexe fermé non vide.
Soit $x \in E$.
Alors il existe une unique application P_C de E dans C appelée projection sur un convexe ($x \mapsto P_C(x)$) telle que :

$$\begin{array}{rcl} \forall y \in E \quad ||x - P_C(x)|| & \leq & ||x - y|| \\ & \Leftrightarrow & \\ ||x - P_C(x)|| & = & \underset{y \in E}{min} ||x - y|| \end{array}$$

On rappelle que la norme euclidienne sur $L^2([0, L])$:

$$||u|| = \left(\int_0^L u(x)^2 dx \right)^{\frac{1}{2}}$$

On applique donc ce théorème à D_a.On note $P_{D_a}(u)$ la projection de u sur D_a On a donc :

$$||P_{D_a}(u) - u|| \quad = \quad \underset{v \in D_a}{min} ||u - v|| \tag{2}$$

On pose ici $\phi(x) = \frac{x^2}{2}$
D'où :

$$\begin{array}{rcl} J(u) & = & \displaystyle\int_0^L \frac{(u(x) - g(x))^2}{2} dx \\ J(u) & = & \displaystyle\frac{1}{2} \int_0^L (u(x) - g(x))^2 dx \\ J(u) & = & \displaystyle\frac{1}{2} ||u - g||^2 \end{array}$$

Soit $u^\star$ solution du système (1) On a donc $\forall u \in D_a$:

$$\begin{array}{rcl} J(u^\star) & \leq & J(u) \\ ||u^\star - g||^2 & \leq & ||u - g||^2 \end{array}$$

Ce qui nous ramène bien à l'équation (2) ci-dessus.
Lorsque $\phi(x) = \frac{x^2}{2}$, la projection de g sur D_a au sens de la norme $L^2([0, L])$ est solution(1).

Convexité de J

On souhaite maintenant déterminer que J est strictement convexe. Celà nous assurera l'unicité de la solution (si elle existe) du problème (1).
Encore une fois, rappelons la définition de stricte convexité.
$\forall \lambda \in]0, 1[$ et $\forall u, v | u \neq v$, J est strictement convexe si et seulement si

$$J(\lambda u + (1 - \lambda) v) < \lambda J(u) + (1 - \lambda) J(v) \tag{3}$$

Supposons que ϕ est strictement convexe. soit $\lambda \in]0,1[$ et $u, v \in D_a$ Par la stricte convexité de ϕ on a :

$$\begin{aligned}
J(\lambda u + (1-\lambda)v) &= \int_0^L \phi((\lambda u + (1-\lambda)v)(x) - g(x))dx \\
&= \int_0^L \phi(\lambda u(x) + (1-\lambda)v(x) - g(x))dx \\
&= \int_0^L \phi(\lambda(u(x) - g(x)) + (1-\lambda)(v(x) - g(x))dx \\
&< \int_0^L \lambda\phi(u(x) - g(x)) + (1-\lambda)\phi(v(x) - g(x))dx \\
&< \lambda \int_0^L \phi(u(x) - g(x))dx + (1-\lambda) \int_0^L \phi(v(x) - g(x))dx \\
&< \lambda J(u) + (1-\lambda)J(v)
\end{aligned}$$

Ce qui est équivalent à l'inéquation (3). J est donc bien strictement convexe si ϕ l'est aussi. Nos deux candidats pour ϕ le sont bien évidemment.

Différentiabilité de J

On va ici déterminer que la fonction J est différentiable ainsi que sa différentielle afin de déterminer si notre problème a une solution. Rappelons tout d'abord ce qu'est une fonction différentielle. Soit f une fonction de $U \subset \mathbb{R}^n$ dans $\mathbb{R}^p$.
Soit $a \in U$ on dit que f est différentiable en a s'il existe une application linéaire L de $\mathbb{R}^n$ dans $\mathbb{R}^p$ telle que :

$$f(a+h) = f(a) + L(h) + o(||h||)$$

Et on dit que L est la différentielle de f.

Ici on suppose que ϕ est différentiable. ϕ est une fonction de $\mathbb{R}$ dans $\mathbb{R}$ On a donc :

$$\begin{aligned}
\phi(u(x) - g(x) + h(x)) &= \phi(u(x) - g(x)) + d\phi(u(x) - g(x)).h(x) + \epsilon(h(x))||h(x)|| \\
\phi(u(x) - g(x) + h(x)) &= \phi(u(x) - g(x)) + \phi'(u(x) - g(x)).h(x) + \epsilon(h(x))||h(x)||
\end{aligned}$$

Avec $d\phi$ la différentielle de ϕ. Or ϕ est une fonction de $\mathbb{R}$ dans $\mathbb{R}$ sa différentielle est donc sa dérivée. On va pouvoir utiliser les formules précédentes dans J :

$$\begin{aligned}
J(u+h) &= \int_0^L \phi(u(x) - g(x) + h(x))dx \\
J(u+h) &= \int_0^L \phi(u(x) - g(x)) + \phi'(u(x) - g(x)).h(x) + \epsilon(h(x))||h(x)||dx
\end{aligned}$$

Par linéarité de l'intégrale on a :

$$\begin{aligned}
J(u+h) &= \int_0^L \phi(u(x) - g(x))dx + \int_0^L \phi'(u(x) - g(x)).h(x)dx + \int_0^L \epsilon(h(x))||h(x)||dx \\
J(u+h) &= J(u) + \int_0^L \phi'(u(x) - g(x)).h(x)dx + \int_0^L \epsilon(h(x))||h(x)||dx
\end{aligned}$$

J est donc bien différentiable sur D_a. Sa différentielle est :

$$dJ(u).h = \int_0^L \phi'(u(x) - g(x)).h(x)dx$$

On va montrer que cette différentielle est continue.
Mais tout d'abord montrons que ϕ' est continue.

On a ϕ convexe et dérivable. ϕ' est donc croissante.
D'après le théorème de la limite monotone, on a $\forall x \in \mathbb{R}$:

$$\sup_{x>y\in\mathbb{R}} \phi'(y) = \lim_{x^-} \phi' \leq \phi'(x) \leq \lim_{x^-} \phi' = \inf_{x<y\in\mathbb{R}} \phi'(y) \tag{4}$$

De plus, on a le Théorème de Darboux qui nous dit :
Soit $f : [a,b] \to \mathbb{R}$ une fonction dérivable. Alors f' vérifie la propriété des valeurs intermédiaires qui est :

$$\forall \lambda \text{ tel que } f'(a) \leq \lambda \leq f'(b), \exists c \in [a,b] \text{ tel quel } \lambda = f'(c)$$

ϕ vérifie cette propriété. Ce qui transforme les inégalités de (4) en égalités :

$$\sup_{x>y\in\mathbb{R}} \phi'(y) = \lim_{x^-} \phi' = \phi'(x) = \lim_{x^-} \phi' = \inf_{x<y\in\mathbb{R}} \phi'(y)$$

ϕ' est donc continue en x et donc continue en tout point de $\mathbb{R}$.

Revenons donc à la continuité de notre différentielle. $\forall h \in \mathcal{C}[0,L]$

$$\begin{aligned}
|dJ(u).h| &= \left| \int_0^L \phi'(u(x) - g(x)).h(x)dx \right| \\
&\leq \int_0^L \left| \phi'(u(x) - g(x)).h(x) \right| dx \\
&\leq \int_0^L M\,|h(x)|\,dx \\
&\leq \int_0^L M||h||_\infty dx \\
&\leq ML||h||_\infty < \infty
\end{aligned}$$

Avec $||h||_\infty = \sup_{x\in[0,L]} |h(x)|$
Et $M = \max_{x\in[0,L]} \phi'(u(x) - g(x))$
L'existence de M est validée par la continuité des fonctions ϕ', u et g.
$dJ(u)$ est donc une application linéaire continue.

Existence d'une solution

Soit f strictement convexe et différentiable sur $\mathbb{R}$, alors une condition nécéssaire et suffisante pour que $x^\star$ soit un minimum global de f est $\nabla f(x^\star) = 0$

Soit $u^\star \in D_a$ solution de (1).

$$J(u^\star) \quad \leq \quad J(u)\ \forall u \in D_a$$

De toute évidence, $u^\star$ minimise J. Comme J est strictement convexe et différentiable, on en déduit que $\nabla J(u^\star) = 0$.

$$\begin{aligned}
dJ(u).h &= \int_0^L \phi'(u^\star(x) - g(x)).h(x)dx = 0 \\
\text{Or } dJ(u).h &= < \nabla J(u^\star), h >
\end{aligned}$$

On en déduit donc que :

$$\int_0^L \phi'(u^\star(x) - g(x))dx = 0$$

D_a étant convexe et fermé, J strictement convexe, différentiable et de différentielle continue, notre problème admet donc bien une unique solution globale.

Partie 2 – Retour à la dimension finie : l'approximation du problème.

On va maintenant s'intéresser à l'approximation discrétisée du problème précédent. Pour cela nous allons approcher le profil recherché $u(x)$ par une fonction continue affine par morceaux. Soit :

— $h = \frac{L}{n-1}$ avec $n \in \mathbb{N}$

— V_h l'ensemble des fonctions continues affines par morceaux sur $[0, L]$ pour un partage en intervalles de longueur h.

— On pose :

$$\begin{aligned} x_i &= ih, \quad i = 0, \ldots, n-1, \\ u_i &= u(x_i), \quad U = (u_0, \ldots, u_{n-1})^T, \\ g_i &= g(x_i), \quad G = (g_0, \ldots, g_{n-1})^T. \end{aligned}$$

Une fonction $u(x)$ de V_h est obtenue par interpolation linéaire entre des valeurs $u_i = u(x_i)$ aux points x_i, $i = 0, \ldots, n-1$. Elle est parfaitement définie par ces valeurs. *i.e.* l'application qui à $u \in V_h$ associe $U \in \mathbb{R}^n$ est un isomorphisme d'espace vectoriel : on peut donc identifier V_h à $\mathbb{R}^n$.
Soit C_h le polyèdre de $\mathbb{R}^n$ défini par

$$C_h = \{U \in ⋗∂≈≂\mathbb{R}^n, \forall i, 1 \leq i \leq n-1, |u_i - u_{i-1}| \leq \alpha h\}$$

Commençons par montrer que

$$u \in V_h \cap D_a \Leftrightarrow U \in C_h.$$

Montrons que $u \in V_h \cap D_a \Rightarrow U \in C_h$.
On suppose que $u \in V_h \cap D_a$
Alors

— u est continue affine par morceaux sur $[0, L]$ pour un partage en intervalles de longueur h.

— $u \in \mathcal{C}([0, L]), \forall x, y \in [0, L], |u(x) - u(y)| \leq \alpha |x - y|$

On veut montrer que $U \in C_h$, i.e si $U = (u_0, \ldots, u_{n-1})^T$, alors $\forall i, 1 \leq i \leq n-1, |u_i - u_{i-1}| \leq \alpha h$
Or, $\forall i, 0 \leq i \leq n-1, u_i = u(x_i) = u(ih)$
On a donc

$$\begin{aligned} |u_i - u_{i-1}| = |u(ih) - u((i-1)h)| \leq \alpha |ih - (i-1)h| \quad &= \alpha |(i - i + 1)h| \\ &= \alpha h \end{aligned}$$

Montrons que $u \in V_h \cap D_a \Leftarrow U \in C_h$.
On suppose que $U \in C_h$, i.e si $U = (u_0, \ldots, u_{n-1})^T$, alors $\forall i, 1 \leq i \leq n-1, |u_i - u_{i-1}| \leq \alpha h$
Soit $x \in [x_i, x_{i+1}]$ et $y \in [x_j, x_{j+1}]$
On a :

$$\begin{aligned} \frac{u(x) - u(y)}{x - y} &= u(x) - u(x_{i+1}) + u(x_{i+1}) - u(x_i) + \ldots + u(x_{k+1}) - u(x_k) \\ &\quad + \ldots + u(x_{j+1}) - u(x_j) + u(x_j) - u(y) \\ &= (x - x_{i+1}) \frac{u(x) - u(x_{i+1})}{x - x_{i+1}} + (x_{i+1} - x_i) \frac{u(x_{i+1}) - u(x_i)}{x_{i+1} - x_i} + \ldots + (x_{k+1} - x_k) \frac{u(x_{k+1}) - u(x_k)}{x_{k+1} - x_k} \\ &\quad + \ldots + (x_{j+1} - x_j) \frac{u(x_{j+1}) - u(x_j)}{x_{j+1} - x_j} + (x_{j+1} - x_j) \frac{u(x_j) - u(y)}{x_{j+1} - x_j} \end{aligned}$$

Or, $\frac{u(x_{k+1}) - u(x_k)}{x_{k+1} - x_k} \leq \alpha, \quad \forall k$

On a donc :

$$
\begin{aligned}
\frac{|u(x)-u(y)|}{|x-y|} &\leq \frac{(x-x_{i+1})\alpha+(x_{i+1}-x_i)\alpha+\ldots+(x_{k+1}-x_k)\alpha+\ldots+(x_{j+1}-x_j)\alpha+(x_j-y)\alpha|}{|x-y|} \\
&= \frac{\alpha|(x-x_{i+1})+(x_{i+1}-x_i)+\ldots+(x_{k+1}-x_k)+\ldots+(x_{j+1}-x_j)+(x_j-y)|}{|x-y|} \\
&= \frac{\alpha|x-y|}{|x-y|} \\
&= \alpha
\end{aligned}
$$

Par suite, on a bien : $\forall x, y \in [0, L], |u(x)-u(y)| \leq \alpha|x-y|$

On approche l'intégrale $J(u)$ en remplaçant les intégrales exactes par leur approximation par la formule des trapèzes

$$\int_{x_i}^{x_{i+1}} \phi(u(x)-g(x))dx \approx \frac{h}{2}(\phi(u(x_i)-g(x_i))+\phi(u(x_{i+1})-g(x_{i+1}))),$$

ce qui donne

$$J(u) \approx I(U) = h\left(\frac{1}{2}\phi(u_0-g_0)+\phi(u_1-g_1)+\ldots+\phi(u_{n-2}-g_{n-2})+\frac{1}{2}\phi(u_{n-1}-g_{n-1})\right).$$

Cas Linéaire

On se place dans le cas d'un problème d'optimisation linéaire, qui demande de minimiser une fonction linéaire sur un polyèdre convexe sous des contraintes linéaires.

On pose $\phi(x) = |x|$ et on montre que l'on peut mettre le problème sous la forme

$$
\begin{cases}
\min_{u_1, z_i} \frac{1}{2}z_0 + z_1 + \ldots + z_j + \ldots + \frac{1}{2}z_{n-1} \\
\text{sous les contraintes :} \\
\forall j, 1 \leq j \leq n-1, |u_j - u_{j-1}| \leq \alpha h \\
\forall j, 0 \leq j \leq n-1, z_j \geq h|u_i - g_i|
\end{cases}
$$

On souhaite minimiser $I(U)$, soit :

$$min_{u_j} \quad \frac{1}{2}\phi(u_0-g_0)+\phi(u_1-g_1)+\ldots+\phi(u_j-g_j)+\ldots+\phi(u_{n-2}-g_{n-2})+\frac{1}{2}\phi(u_{n-1}-g_{n-1})$$

Ici, $\phi(x) = |x|$
Le problème devient donc :

$$min_{u_j} \quad \frac{1}{2}|u_0-g_0|+|u_1-g_1|+\ldots+|u_j-g_j|+\ldots+|u_{n-2}-g_{n-2}|+\frac{1}{2}|u_{n-1}-g_{n-1}| \qquad \text{(PO1)}$$

Au niveau des contraintes, on a :

— $U \in C_h \Rightarrow \forall j \in [\![1; n-1]\!], \quad |u_j - u_{j-1}| \leq \alpha h$

— Soit $z_j \geq |u_j - g_j|, \quad \forall j \in [\![1; n-1]\!]$
On a alors :
$\frac{1}{2}z_0 + z_1 + \ldots + z_j + \ldots + z_{n-2} + \frac{1}{2}z_{n-1} \geq h(\frac{1}{2}|u_0-g_0|+|u_1-g_1|+\ldots+|u_j-g_j|+\ldots+|u_{n-2}-g_{n-2}|+\frac{1}{2}|u_{n-1}-g_{n-1}|)$
Ainsi :
$min_{z_j} \quad \frac{1}{2}z_0 + z_1 + \ldots + z_j + \ldots + z_{n-2} + \frac{1}{2}z_{n-1} \geq min_{u_j} \quad h(\frac{1}{2}|u_0-g_0|+|u_1-g_1|+\ldots+|u_j-g_j|+\ldots+|u_{n-2}-g_{n-2}|+\frac{1}{2}|u_{n-1}-g_{n-1}|)$

En posant $z_j = h|u_j - g_j|$, on a bien :

$$min_{u_j,z_j} \quad \frac{1}{2}z_0 + z_1 + \ldots + z_j + \ldots + z_{n-2} + \frac{1}{2}z_{n-1}$$

On a déjà prouvé l'existence de $u^\star = (u_0^\star, \ldots, u_j^\star, \ldots, u_{n-1}^\star)$ qui minimise (PO1). Donc $z^\star = (z_0^\star, \ldots, z_j^\star, \ldots, z_{n-1}^\star)$ avec $z_j^\star = h|u_j^\star - g_j|, \quad \forall j \in [\![1; n-1]\!]$ existe aussi.

Sachant que les contraintes se réécrivent :

$$\begin{array}{rrcl}
\forall j \in [\![1; n-1]\!], |u_j - u_{j-1}| \leq \alpha h \Leftrightarrow \forall j \in [\![1; n-1]\!], & u_j - u_{j-1} & \leq & \alpha h \\
 & u_{j-1} - u_j & \leq & \alpha h \\
\forall j \in [\![0; n-1]\!], z_j \geq \alpha h|u_j - u_{j-1}| \Leftrightarrow \forall j \in [\![0; n-1]\!], & u_j - g_j & \leq & \frac{z_j}{h} \\
 & g_j - u_j & \leq & \frac{z_j}{h} \\
 & \Leftrightarrow & & \\
 & u_j - \frac{z_j}{h} & \leq & g_j \\
 & -u_j - \frac{z_j}{h} & \leq & -g_j
\end{array}$$

On montre alors que ce problème peut se résoudre avec l'algorithme du simplexe en le mettant sous la forme suivante :

$$\left\{ \begin{array}{l} min_{V \in \mathrm{R^n}} \quad a^t v \\ s.c \quad Cv - d \preceq 0 \end{array} \right.$$

avec

$$a = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ \frac{1}{2} \\ 0 \\ \vdots \\ 0 \\ \frac{1}{2} \end{pmatrix} C = [C_u \quad C_z], \quad v = \begin{pmatrix} u_0 \\ \vdots \\ u_{n-1} \\ z_0 \\ \vdots \\ z_{n-1} \end{pmatrix} \quad \text{et} \quad d = \begin{pmatrix} \alpha h \\ \vdots \\ \alpha h \\ g_0 \\ \vdots \\ g_{n-1} \\ -g_0 \\ \vdots \\ -g_{n-1} \end{pmatrix}$$

où $a \in \mathbb{R}^{2n}, \quad C \in \mathcal{M}_{2(2n-1) \times 2n}, \quad v \in \mathbb{R}^{2n} \quad \text{et} \quad d \in \mathbb{R}^{2(2n-1)}$ et

$$C_u = \begin{pmatrix} -1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & -1 & 1 \\ 1 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -1 \\ 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 1 \\ -1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & -1 \end{pmatrix} \quad \text{et} \quad C_z = \begin{pmatrix} 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ -\frac{1}{h} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & -\frac{1}{h} \\ -\frac{1}{h} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & -\frac{1}{h} \end{pmatrix}$$

$$C = \begin{pmatrix} -1 & 1 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & -1 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -1 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 & -\frac{1}{h} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 1 & 0 & 0 & \dots & 0 & -\frac{1}{h} \\ -1 & 0 & \dots & 0 & 0 & -\frac{1}{h} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & -1 & 0 & 0 & \dots & 0 & -\frac{1}{h} \end{pmatrix},$$

Ainsi, la fonction à minimiser et les contraintes sont bien linéaires.
Par suite, le problème peut se résoudre avec l'algorithme du simplexe.

Visualisation

Le programme est codé en python 3 et se lance avec la commande *python projet.py.* On a alors le choix entre le cas linéaire (0) ou le cas quadratique (1). Ici, il faut entrer 0. Un jupyter notebook du programme, *projet3m101NB.ipynb*, est également disponible.

Après avoir calculer les vecteurs a, d et la matrice C, le programme résout le problème en utilisant la fonction linprog de la bibliothèque scipy (0) ou un programme maison du simplexe (1). Ce dernier prend un certains temps à s'exécuter en entier.
Les tracés sont lissés en s'appuyant sur l'interpolation de Spline (interpolate.splrep et interpolate.splev).

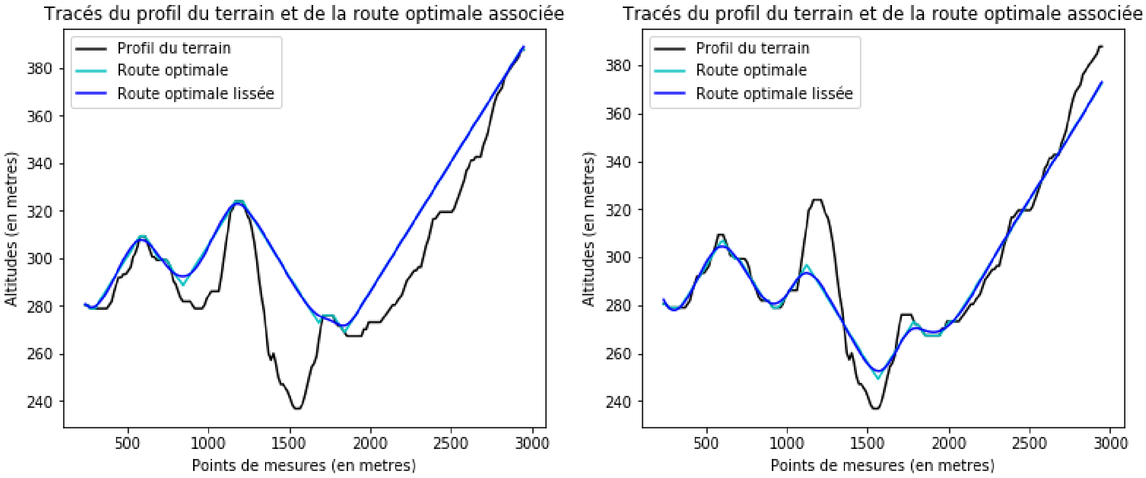

FIGURE 1 – Tracés des routes optimales obtenues par scipy.linprog (à gauche) et par le simplexe maison (à droite)

Le simplexe maison fournit aussi l'évolution de la valeur de la fonction objective au cours des itérations.

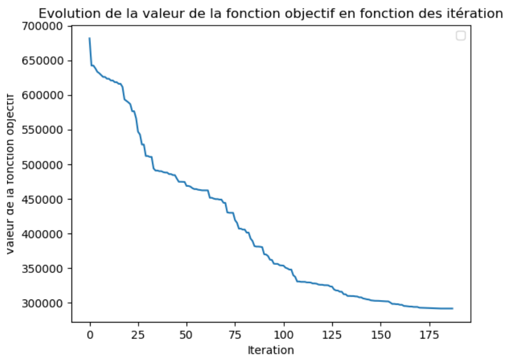

FIGURE 2 – Pour $\alpha = 10\%$, évolution de la valeur de la fonction objective au cours des itérations par le simplexe maison

On avait fixé $\alpha = 10\%$ mais on peut faire varier la valeur de la pente maximale autorisée.

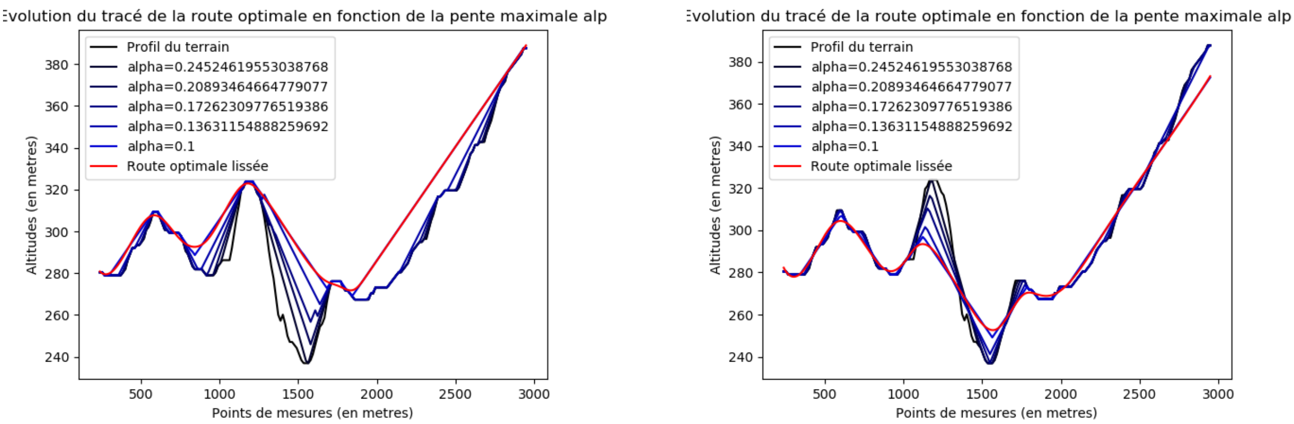

FIGURE 3 – Tracés des routes optimales obtenues par scipy.linprog (à gauche) et par le simplexe maison (à droite) en fonction de la pente maximale autorisée α

Cas quadratique

On se place dans le cas d'un problème d'optimisation quadratique dans lequel on minimise une fonction quadratique sur un polyèdre convexe sous des contraintes linéaires..

On pose $\phi(z) = z^2/2$.
Tout d'abord, on explicite une matrice A telle que

$$I(U) = \langle A(U-G), U-G\rangle.$$

En posant :

$$U = \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-2} \\ u_{n-1} \end{pmatrix}, \quad G = \begin{pmatrix} g_0 \\ g_1 \\ \vdots \\ g_{n-2} \\ g_{n-1} \end{pmatrix}, \quad A = \begin{pmatrix} \frac{h}{4} & 0 & \dots & 0 & 0 \\ 0 & \frac{h}{2} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \frac{h}{2} & 0 \\ 0 & 0 & \dots & 0 & \frac{h}{4} \end{pmatrix} \quad \text{avec } U, G \in \mathbb{R}^n \text{ et } A \in \mathcal{M}_n$$

On a :

$$U - G = \begin{pmatrix} u_0 - g_0 \\ u_1 - g_1 \\ \vdots \\ u_{n-2} - g_{n-2} \\ u_{n-1} - g_{n-1} \end{pmatrix}$$

Donc :

$$A(U-G) = \begin{pmatrix} \frac{h}{4} & 0 & \dots & 0 & 0 \\ 0 & \frac{h}{2} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \frac{h}{2} & 0 \\ 0 & 0 & \dots & 0 & \frac{h}{4} \end{pmatrix} \begin{pmatrix} u_0 - g_0 \\ u_1 - g_1 \\ \vdots \\ u_{n-2} - g_{n-2} \\ u_{n-1} - g_{n-1} \end{pmatrix} = \begin{pmatrix} \frac{h}{4} \times (u_0 - g_0) \\ \frac{h}{2} \times (u_1 - g_1) \\ \vdots \\ \frac{h}{2} \times (u_{n-2} - g_{n-2}) \\ \frac{h}{4} \times (u_{n-1} - g_{n-1}) \end{pmatrix}$$

Ainsi :

$$\begin{aligned}
\langle A(U-G), U-G\rangle &= \begin{pmatrix} \frac{h}{4} \times (u_0 - g_0) \\ \frac{h}{2} \times (u_1 - g_1) \\ \vdots \\ \frac{h}{2} \times (u_{n-2} - g_{n-2}) \\ \frac{h}{4} \times (u_{n-1} - g_{n-1}) \end{pmatrix}^T \begin{pmatrix} u_0 - g_0 \\ u_1 - g_1 \\ \vdots \\ u_{n-2} - g_{n-2} \\ u_{n-1} - g_{n-1} \end{pmatrix} \\
&= \begin{pmatrix} \frac{h}{4}(u_0 - g_0) & \frac{h}{2}(u_1 - g_1) & \dots & \frac{h}{2}(u_{n-2} - g_{n-2}) & \frac{h}{4}(u_{n-1} - g_{n-1}) \end{pmatrix} \begin{pmatrix} u_0 - g_0 \\ u_1 - g_1 \\ \vdots \\ u_{n-2} - g_{n-2} \\ u_{n-1} - g_{n-1} \end{pmatrix} \\
&= \frac{h}{4}(u_0 - g_0)^2 + \frac{h}{2}(u_1 - g_1)^2 + \dots + \frac{h}{2}(u_{n-2} - g_{n-2})^2 + \frac{h}{4}(u_{n-1} - g_{n-1})^2 \\
&= \frac{h}{2}\frac{(u_0 - g_0)^2}{2} + h\frac{(u_1 - g_1)^2}{2} + \dots + h\frac{(u_{n-2} - g_{n-2})^2}{2} + \frac{h}{2}\frac{(u_{n-1} - g_{n-1})^2}{2} \\
&= h(\frac{1}{2}\frac{(u_0 - g_0)^2}{2} + \frac{(u_1 - g_1)^2}{2} + \dots + \frac{(u_{n-2} - g_{n-2})^2}{2} + \frac{1}{2}\frac{(u_{n-1} - g_{n-1})^2}{2} \\
&= h(\frac{1}{2}\phi(u_0 - g_0) + \phi(u_1 - g_1) + \dots + \phi(u_{n-2} - g_{n-2}) + \frac{1}{2}\phi(u_{n-1} - g_{n-1}) \\
&= I(U)
\end{aligned}$$

Le problème revient à trouver $U^\star = (U_0^\star, \dots, U_{n-1}^\star)^T \in C_h,$ tel que $\forall U \in C_h \quad I(U^\star) \leq I(U)$.
ou mis sous la forme canonique d'un problème d'optimisation sous contraintes d'inégalités

$$\inf_{CU-b\preceq 0} I(U)$$

avec $C \in \mathcal{M}_{2(n-1)\times n}(\mathrm{R})$ et $b \in \mathrm{R}^{2(n-1)}$. On explicite alors la matrice C et le vecteur b.

Sachant que la contrainte $U \in C_h$ se réécrit : $\forall i \in [\![1; n-1]\!], |u_i - u_{i-1}| \leq \alpha h$
ou bien :

$$\forall i \in [\![1; n-1]\!], \quad \begin{array}{rcl} u_i - u_{i-1} & \leq & \alpha h \\ u_{i-1} - u_i & \leq & \alpha h \end{array}$$

On a le problème d'optimisation suivant :

$$\begin{cases} inf_{U\in C_h} = \langle A(U-G), U-G\rangle \\ s.c \quad CU - b \preceq 0 \end{cases} \tag{PO2}$$

avec

$$C = \begin{pmatrix} -1 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -1 & 1 \\ 1 & -1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 & -1 \end{pmatrix} \quad \text{et} \quad b = \begin{pmatrix} \alpha h \\ \vdots \\ \alpha h \\ \vdots \\ \alpha h \end{pmatrix}$$

avec $C \in \mathcal{M}_{2(n-1)\times n}$ et $b \in \mathbb{R}^{2(n-1)}$

Au passage, ce problème admet bien une unique solution : I est strictement convexe (vrai si $\phi(z)$ l'est) et C_h est convexe (vrai car c'est un polyedre), alors le problème d'optimisation a au plus une solution.

En effet, en raisonnant par l'absurde.
Soient $U_1^\star, U_2^\star \in C_h$ avec $U_1^\star \neq U_2^\star$ deux minumums de I sur C_h. On a alors :

$$I(U_1^\star) = I(U_2^\star) \leq I(U) \forall U \in C_h$$

On pose $U = \frac{1}{2}U_1^\star + \frac{1}{2}U_2^\star$. Comme I est strictement convexe, on a :

$$I(\frac{1}{2}U_1^\star + \frac{1}{2}U_2^\star) < \frac{1}{2}I(U_1^\star) + \frac{1}{2}I(U_2^\star) = I(U_1^\star)$$

(car $I(U_1^\star) = I(U_2^\star)$)
On a exhiber un U tel que $I(U) < I(U_1^\star)$. Contradiction !

On rappelle les conditions KKT du premier ordre relatif à ce problème.

Soit $U^\star$ la solution de (PO2). On sait que I est différentiable en $U^\star$ et on a $rg(C) = n$.
Alors, les conditions KKT du premier ordre impliquent qu'il existe $\lambda^\star \in (R)^n$ tel que l'on ait :

$$\begin{cases} \nabla_{U^\star} L(U^\star, \lambda^\star) = 0 \quad \text{avec } L(U^\star, \lambda^\star) = I(U^\star) + \langle \lambda^\star, CU^\star - b \rangle \\ CU^\star - b \preceq 0 \\ \lambda^\star \geq 0 \\ (\lambda^\star)^T (CU^\star - b) = 0 \end{cases}$$

On peut obtenir une expression de $\lambda^\star$.
Comme $L(U^\star, \lambda^\star) = I(U^\star) + \langle \lambda^\star, CU^\star - b \rangle$, alors $\nabla_{U^\star} L(U^\star, \lambda^\star) = \nabla_{U^\star} I(U^\star) + \nabla_{U^\star} \langle \lambda^\star, CU^\star - b \rangle$.

On cherche $\nabla_{U^\star} I(U^\star)$:

$$\begin{aligned} I(U^\star + h) &= \frac{1}{2}\langle M(U^\star + h), (U^\star + h)\rangle + \langle V, (U^\star + h)\rangle \\ &= \frac{1}{2}\langle MU^\star + Mh, (U^\star + h)\rangle + \langle V, U^\star\rangle + \langle V, h\rangle \\ &= \frac{1}{2}[\langle M_1 U^\star, U^\star\rangle + \langle MU^\star, h\rangle + \langle Mh, U^\star\rangle + \langle Mh, h\rangle] + \langle V, U^\star\rangle + \langle V, h\rangle \end{aligned}$$

Or,

$$\langle Mh, U^\star\rangle = (Mh)^T u = h^T A^T u = \langle h, M^T U^\star\rangle = \langle h, MU^\star\rangle = \langle MU^\star, h\rangle$$

Donc,

$$\begin{aligned} I(U^\star + h) &= \frac{1}{2}\langle M_1 U^\star, U^\star\rangle + \langle V, U^\star\rangle + \frac{1}{2}2\langle MU^\star, h\rangle + \frac{1}{2}\langle Mh, h\rangle + \langle V, h\rangle \\ &= I(U^\star) + \langle MU^\star, h\rangle + \langle V, h\rangle + \frac{1}{2}\langle Mh, h\rangle \\ &= I(U^\star) + \langle MU^\star + V, h\rangle + \frac{1}{2}\langle Mh, h\rangle \end{aligned}$$

Ainsi, $\nabla_{U^\star} I(U^\star) = MU^\star + V$.

On cherche $\nabla_{U^\star} \langle \lambda^\star, CU^\star - b\rangle$:

$$\begin{aligned} \langle \lambda^\star, CU^\star - b\rangle &= \langle \lambda^\star, CU^\star\rangle - \langle \lambda^\star, b\rangle \\ \nabla_{U^\star}\langle \lambda^\star, CU^\star - b\rangle &= \nabla_{U^\star}\langle \lambda^\star, CU^\star\rangle - \nabla_{U^\star}\langle \lambda^\star, b\rangle \\ \nabla_{U^\star}\langle \lambda^\star, CU^\star - b\rangle &= \nabla_{U^\star}\langle \lambda^\star, CU^\star\rangle \\ \nabla_{U^\star}\langle \lambda^\star, CU^\star - b\rangle &= \nabla_{U^\star}(CU^\star)^T \lambda^\star \\ \nabla_{U^\star}\langle \lambda^\star, CU^\star - b\rangle &= C^T \lambda^\star \end{aligned}$$

On a donc, $\nabla_{U^\star} L(U^\star, \lambda^\star) = MU^\star + V + C^T \lambda^\star$.

En toute généralité, ces conditions KKT sont des conditions nécessaires (sont toujours réalisées à l'optimum du problème). Mais elles ne sont pas forcément suffisantes. Néanmoins, elles le sont ici car I est convexe et les contraintes sont linéaires.

Conclusion

Notre but était d'optimiser le tracé d'une route de manière à respecter des contraintes d'utilisation (pente maximum) tout en minimisant le coût de construction.
Au cours de ce projet, on a approché le problème sous deux points de vue : un problème d'optimisation linéaire avec l'algorithme du simplexe et un problème d'optimisation quadratique avec l'algorithme d'Uzawa.
Les tracés de la future route obtenus par les différents algorithmes sont ensuite lissés à l'aide d'une interpolation de Spline. Les profils des routes optimales trouvés ainsi que le coût des déblais et des remblais à effectuer sont par la suite affichés.