Mini-projet optimisation

Delbouve Paul, Mondon Arnaud

Avril 2020

Chapitre 1

Modélisation

1.1 Généralités pour un véhicule

Considérons dans un premier temps un seul véhicule. L'utilisateur dépose le véhicule à une heure t_0 avec un état de charge c_0 et vient le récupérer à une heure t_f avec une charge c_f.

Le véhicule est relié au réseau électrique, celui-ci peut fournir une intensité I_{max}. On fera l'hypothèse que les chargements se feront à tension constante. On peut donc éliminer cette variable du problème.

1.1.1 Fonction coût

Il existe deux tarifs horaires, on considère alors les deux ensembles :

— T_+ qui est l'ensembles des heures pleines, le prix de l'$A.h$ est alors de p_+.

— T_- qui est l'ensemble des heures creuses, le prix de l'$A.h$ est alors de p_-.

Vraisemblablement ces ensembles sont des intervalles ou unions d'intervalles (heures creuses la nuit par exemple).

L'objectif est ici de minimiser le coût de l'énergie nécessaire à la recharge du véhicule. On a une fonction coût représentée par :

$$\int_{t_0}^{t_f} I(t)p(t)dt$$

On peut séparer cette intégrale en deux parties (selon les heures creuse ou pleines).

$$\begin{aligned} & \int_{[t_0,t_f]\cap T_+} I(t)p(t)dt + \int_{[t_0,t_f]\cap T_-} I(t)p(t)dt \\ = \quad & p_+ \int_{[t_0,t_f]\cap T_+} I(t)dt + p_- \int_{[t_0,t_f]\cap T_-} I(t)dt \end{aligned}$$

On cherche une fonction I qui miniserait ce réel.

1.1.2 Contraintes

Un véhicule a une batterie de capacité Q_{max} (en $A.h$). Afin que l'utilisateur soit satisfait il faut que :

$$c_f - c_0 = \frac{1}{Q_{max}} \int_{t_0}^{t_f} I(t)dt$$

De plus on a la contrainte :

$$\forall t, 0 \leq I(t) < I_{max}$$

1.2 Problème

1.2.1 Avant-propos

La difficulté vient ici du fait que l'on cherche non pas un vecteur $x \in R^n$ pour lequel la fonction coût serait minimale mais on cherche une fonction I qui minimiserait la fonction coût. Il s'agit là d'optimiation en dimension infinie. Pour résoudre ce problème avec nos connaissances il faudra faire des simplifications.

Intuitivement on peut se dire que I ne prendra au plus que deux valeurs, I_{max} et 0 (fonctionnement tout ou rien). De plus, manuellement on voit que pour minimiser le coût de l'électricité, il suffit de charger le plus possible durant les heures creuses puis compléter en chargeant pendant les heures pleines si nécessaire.

1.2.2 Données

Le graphique ci-dessous représente la charge et l'intensité fournie par la batterie d'un véhicule au cours du temps. On remarque que l'intensité est le plus souvent négative ce qui est cohérent avec le fait que le véhicule est en chargement.

A l'aide de ces données vérifions l'équation de chargement donnée précedemment :

$$c_f - c_0 = \frac{1}{Q_{max}} \int_{t_0}^{t_f} I(t)dt \tag{1.1}$$

Calcul de C_{max} :

$$Q_{max} = \frac{1}{c_f - c_0} \int_{t_0}^{t_f} I(t)dt$$

Numériquement on obtient une valeur de -187A.h pour l'intégrale et donc $Q_{max} = 505$ A.h.

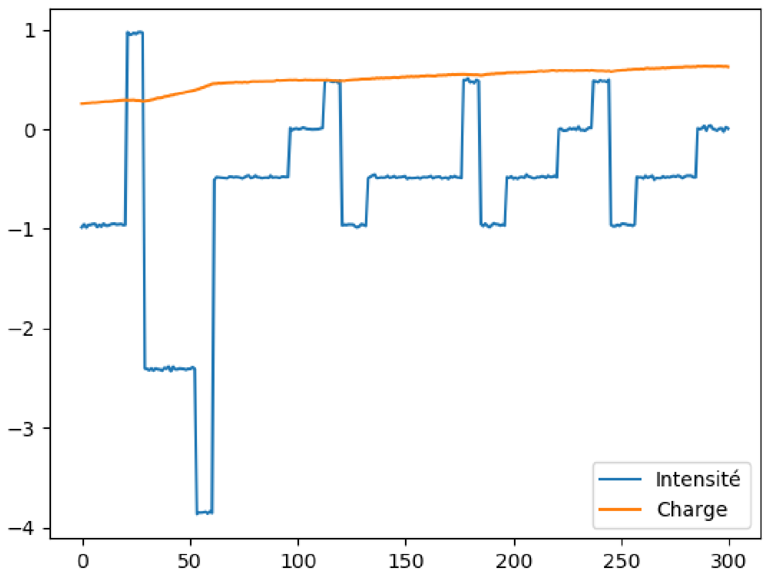


FIGURE 1.1 – Intensité et taux de charge de la batterie

On trace la courbe de chargement avec le modèle (1).

On voit bien que les courbes de la charge se superposent parfaitement, le modèle est validé.

1.2.3 Discrétisation

Afin de simplifier le problème on décide de discrétiser le temps, on approche donc la fonction I par une fonction en escalier. Ainsi quand un véhicule se présente à la station de chargement on a t_0 et t_f qui sont fixés. On découpe alors notre intervalle en une subdivision $\{x_i, \forall i \in \{0, ..., n\}\}$. On définit alors I par :

$$\forall t \in [t_0, t_f], I(t) = y_i, t \in [x_i, x_{i+1}]$$

La fonction coût s'exprime alors comme une fonction des y_i. On note $J^+ = \{i, [x_i, x_{i+1}] \subset T^+\}$ et $J^- = \{i, [x_i, x_{i+1}] \subset T^-\}$

$$f(y_1, ..., y_n) = p_+ \sum_{i \in J^+} y_i(x_{i+1} - x_i) + p_- \sum_{i \in J^-} y_i(x_{i+1} - x_i)$$

De même on écrit l'équation de chargement dans le cas discrétisé :

$$c_f = c_0 + \frac{1}{Q_{max}} \sum_i y_i(x_{i+1} - x_i)$$

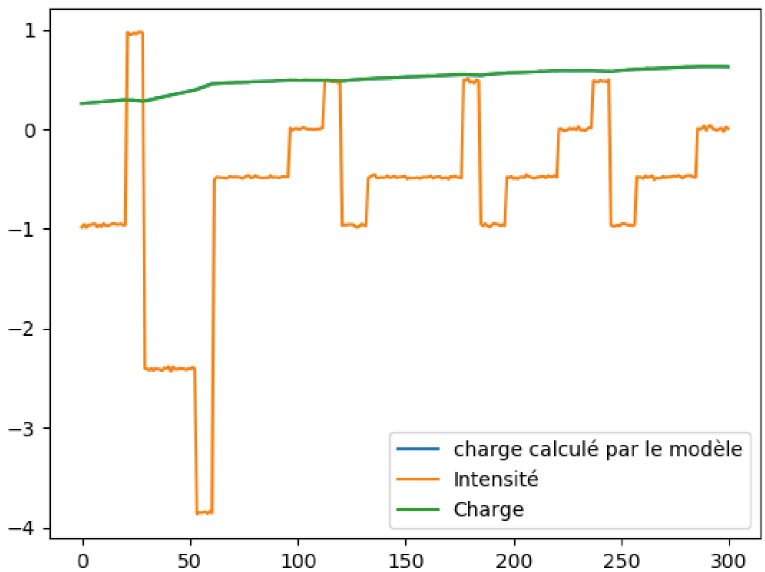


FIGURE 1.2 – Intensité, taux de charge de la batterie et taux de charge calculé par le modèle

Ainsi que les contraintes sur les y_i :

$$\forall i, 0 \leq y_i < I_{max}$$

1.3 Gestion de plusieurs véhicules

On s'interesse maintenant au cas où plusieurs véhicules arrivent à la station.

1.3.1 Hypothèses

Les hypothèses de fonctionnement de notre système sont les suivantes.

— Chaque véhicule peut recevoir de l'énergie de la station ou bien en fournir grâce à sa batterie. On suppose que $-I_{max} < I(t) < I_{max}$ pour chaque véhicule. Cela pourrait correspondre à une limitation technique des bornes. Une batterie peut au plus se charger à Qmax et se décharger à 0.

— De même on suppose que le réseau peut apporter un énergie limitée à la station. Ainsi on suppose que $I_{rseau} < I_{max}$. Ainsi si deux véhicules

entièrement dechargés se connectent à la station ils pourront au plus se recharger à $I_{max}/2$.
— On suppose que le nombre de véhicules présents est inférieur au nombre de bornes disponibles.

1.3.2 Fonction de coût et contraintes

La fonction coût dans se cas se resume à l'énergie prélevée sur le réseau. On note y_i^k l'intensité sortant du k_{ime} véhicule au pas de discretisation i.Ainsi à un temps de discrétisation i l'intensité venant du réseau est :

$$I_{rseau} + \sum_k y_i^k = 0$$

Cette relation découle simplement d'une loi des noeuds (on rappelle que les y_i^k sont algébriques). Les y_i^k sont négatifs si le véhicule reçoit de l'énergie et positifs dans le cas contraire. Ainsi si les véhicules se chargent plus qu'il ne fournissent au réseau on aura $I_{rseau} > 0$ (le réseau fournit de l'énergie). On fait l'hypothèse que la station ne peut revendre d'énergie au réseau et donc $0 \leq I_{rseau}$

On peut alors écrire la fonction de coût :

$$f((y_i^k)_{i,k}) = \sum_i (\sum_k y_i^k)(x_{i+1} - x_i)p(x_i)$$

La contrainte sur le réseau donne alors :

$$\forall i, -I_{max} \leq \sum_k y_i^k \leq 0$$

On écrit les deux contraintes pour chacun des véhicules :

$$\forall i, \forall k - I_{max} < y_i^k < I_{max}$$

De même pour la contrainte de chargement pour chacun des véhicules :

$$\forall k, c_f^k = c_0^k + \frac{1}{C_{max}^k} \sum_i y_i^k (x_{i+1} - x_i)$$

On note $[x_0, x_n]$ l'intervalle sur lequel on étudie la station de chargement. Un véhicule ne pouvant être chargé que lorsqu'il est connecté il apparait les deux contraintes supplémentaires par véhicule :

$$\forall k, t \in [x_0, t_0^k] \cup [t_f^k, x_n] \rightarrow I^k(t) = 0$$

Cela donne dans le cas discrétisé :

$$\forall k, \forall i, [x_i, x_{i+1}] \subset [x_0, t_0^k] \cup [t_f^k, x_n] \rightarrow y_i^k = 0$$

Cette contrainte permet de définir l'intensité de chargement pour un véhicule pendant tout le temps d'étude et non pas seulement pendant le temps de chargement.

Il convient également de rajouter que le taux de charge de chacun des véhicule ne peut exceder 100% ni être en dessous de 0%. Ceci s'écrit formellement

$$\forall k, \forall p \in [1, n],\ 0 \leq c^k(x_p) \leq 1$$

Soit

$$\forall k, \forall p \in [1, n],\ 0 \leq c_0 + \frac{1}{C^k_{max}} \sum_{i=1}^{p-1} y_i^k (x_{i+1} - x_i) \leq 1$$

1.3.3 Bilan

contrainte	expression	nombre
réseau	$\forall i, -I_{max} \leq \sum_k y_i^k \leq 0$	2
borne	$\forall i, \forall k - I_{max} < y_i^k < I_{max}$	$2k$
chargement	$\forall k, c_f^k = c_0^k + \frac{1}{C^k_{max}} \sum_i y_i^k (x_{i+1} - x_i)$	k
présence du véhicule	$\forall k, \forall i, [x_i, x_{i+1}] \subset [x_0, t_0^k] \cup [t_f^k, x_n] \rightarrow y_i^k = 0$	$2k$
capacité de la batterie	$\forall k, \forall p \in [1, n],\ 0 \leq c_0 + \frac{1}{C^k_{max}} \sum_{i=1}^{p-1} y_i^k (x_{i+1} - x_i) \leq 1$	$2k$

On obtient $7k + 2$ contraintes.

Chapitre 2

Etude et résolution numérique pour un véhicule

2.1 Ordres de grandeurs

Il existe trois grandes familles de chargeurs de véhicules électriques :
- la recharge sur une prise domestique (8A) qui prend plusieurs heures pour un chargement complet.
- La recharge sur une *wallbox* (16 A) qui permet de réduire le temps de chargement.
- les bornes de recharge en station (120 A) qui permettent de réduire considérablement le temps de recharge (moins d'une heure).

Notre modélisation correspond au dernier cas, on prendra une intensité maximale $I_{max} = 100$A.

On suppose que les véhicules ont des batteries de capacités 100 A.h, donc en chargeant à intensité maximale il est possible de charger le véhicule en 1h ce qui semble être un bon ordre de grandeur.

Dans notre modélisation le temps est en minutes, les capacités des batteries valent donc : $6000 A.min$. Le prix de l'A.min est choisit de telle sorte à avoir un prix de recharge complet inférieur à 50 euros.

Afin de normaliser les données on choisit de diviser les intensités et les capacités par 100.

2.2 Etude du problème

Les fonctions f et c sont linéaires donc convexes. On écrit les fonctions sous les formes suivantes :

$$f(y_1, ..., y_n) = F^T Y$$

$$c(y_1, ..., y_n) = AY - b$$

Avec F un vecteur de taille n, A une matrice de taille (m, n) et b une vecteur de taille m, où m est le nombre de contraintes.

2.2.1 Ensemble des solutions réalisables

L'ensemble des solutions réalisables est non vide sous certaines conditions. Pour que le véhicule puisse se charger comme le souhaite l'utilisateur, il faut que l'énergie que peut fournir la borne de chargement soit supérieure à l'énergie demandée par l'utilisateur sur la plage de temps de chargement. Si ce n'est pas le cas, les contraintes sur le caractère borné de I et sur le chargement à obtenir sont contradictoires et aucune solution n'est réalisable. Explicitement on doit donc avoir :

$$(c_f - c_0)Q_{max} \leq (t_f - t_0)I_{max}$$

On peut constuire explicitement une solution réalisable (bien que probablement pas optimale) en prennant une intensité constante valant :

$$\forall i, y_i = \frac{(c_f - c_0)Q_{max}}{(t_f - t_0)}$$

On supposera maintenant que la condition est vérifiée et que l'ensemble des

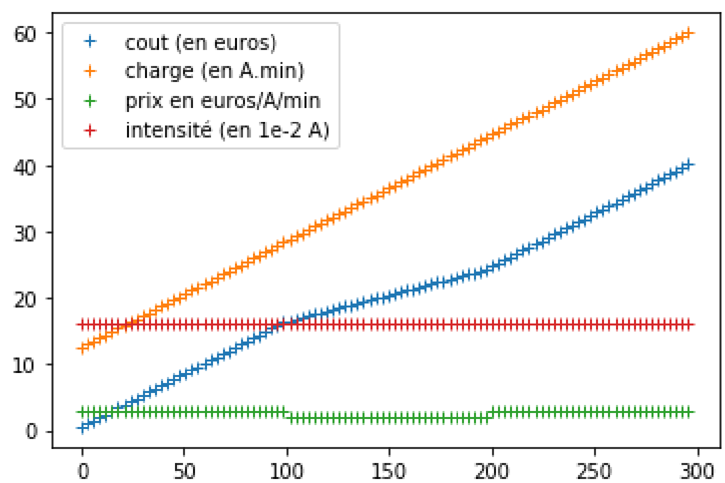


FIGURE 2.1 – Construction d'une solution optimale

solutions réalisables est non vide.

2.2.2 Construction d'une solution optimale

Il est possible de consruire manuellement une solution optimale au problème. Pour cela on maximise la part du temps de chargement pendant la période où le tarif est bas. Il y a deux cas :

- les heures creuses suffisent à fournir le chargement nécessaire et on ne charge donc pas pendant les heures pleines. On charge uniformément pendant les heures creuses.
- les heures creuses ne sont pas suffisantes, on charge à I_{max} et on complète en chargeant uniformément pendant les heures pleines.

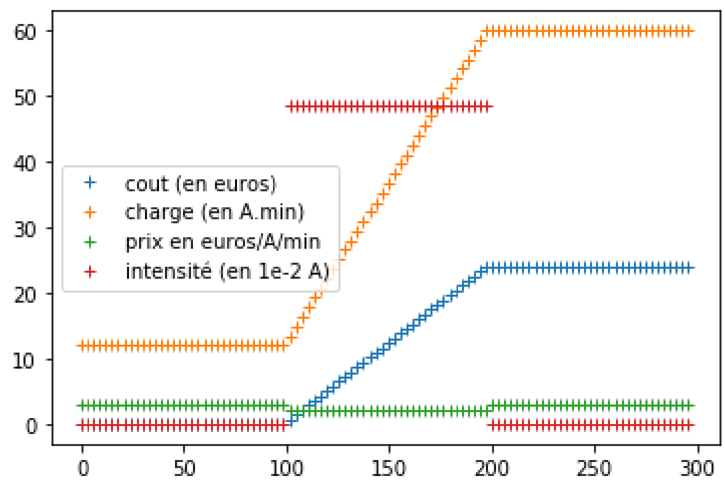


FIGURE 2.2 – Construction d'une solution optimale

2.2.3 Optimalité de la solution construite

Remarquons que seules les proportions de charges fournies en heures creuses et en heures pleines influencent le prix. Il est donc clair qu'il n'y a pas d'unicité de la solution au problème. Si on charge moins pendant les heures creuses alors il faudra compenser sur les heures pleins ce qui augmentera le prix. Si on charge plus sur les heures creuses alors la charge de la voiture sera trop importante et la contrainte de charge ne sera plus respectée. On conclue que la proportion choisie dans la construction minimise le prix. Elle n'est cependant pas l'unique solution.

On remarque que le gain pour l'utilisateur est de l'ordre de 15 euros pour un montant initial de 40 euros, ce montant n'est pas négligeable.

2.3 Résolution

2.3.1 Résolution avec *Scipy*

La librairie Python *Scipy* et son module *optimize* fournissent une fonction permettant de résoudre des problèmes d'optimisation linéaire sous contraintes.

Son exécution est très rapide et donne un résultat très semblable à la méthode constructiviste.

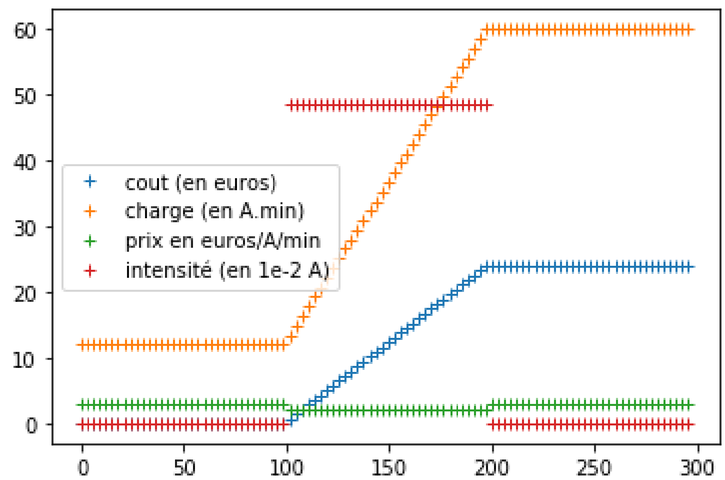


FIGURE 2.3 – Résolution avec Scipy

2.3.2 Recherche d'un point selle par l'algorithme d'Uzawa

L'algorithme d'Uzawa permet de rechercher un point selle du Lagrangien, ce point selle donne l'intensité solution du problème d'optimisation sous contrainte.

L'algorithme d'Uzawa requiet de minimiser $Y \to L(Y, \lambda^k)$ à chaque itération. Or cette fonction de Y est linéaire (de courbure infinie) et ne peut donc être résolue.

Utiliser un pas constant l en écrivant $Y^{k+1} = Y^k - l\nabla_Y L(Y^k, \lambda^k)$ ne donne pas des résultats probants. De même il n'est pas possible de faire une recherche de pas optimal car la fonction de l est aussi linéaire. La recherche d'un pas respectant les critères de Wolfe ne fonctionne pas non plus.

Ainsi nous avons décidé de nous orienter vers des méthodes plus spécifiques aux problèmes linéaires. Il est existe principalement deux classes de méthodes pour la résolution de ces problèmes :

- Les méthodes de points intérieurs qui partent d'un point dans l'intérieur de l'ensemble des solutions réalisables et naviguent dans cette ensemble jusqu'à approcher la solution. On implémentera en particulier la méthode d'*affine scaling*.
- Les methodes de simplexes qui utilisent le fait qu'une solution à un problème linéaire sous contraintes se trouve nécessairement sur un point

extrême de l'ensemble des solutions réalisables (sur un sommet du polyèdre formé par les contraintes). L'algorithme du simplexe permet de passer d'un point extrême à l'autre pour trouver la solution exacte au problème.

2.3.3 Algorithme des points intérieurs

On utilise la méthode *affine scaling*. Cette méthode permet de résoudre les problèmes de la forme :

$$min_x c^T x$$

$$Ax = b$$

$$x \geq 0$$

Ce n'est pas exactement la forme sous lequel est notre problème. En effet on a des contraintes inégalités et Y qui est négative. Ce dernier point ne pose pas de difficulté en considérante la nouvelle variable $X = -Y$.

En ce qui concerne les inégalités, on remarque que :

$$ax \leq b \Leftrightarrow \exists x_1 \geq 0, ax + x_1 = b$$

Ainsi par l'ajout de variables appelées *variables d'écart*, il est possible d'écrire notre problème sous la forme voulue.

Les contraintes que l'on considère sont la contrainte (égalité) de charge et les contraintes inégalités de borne. Ces dernières n'ont pas besoin d'être appliquées en tout temps, seulement lors des heures creuses (si elles devaient l'être en tout temps alors l'énergie demandée par le véhicule serait trop importante). Le nombre de contraintes égalités (avec la méthode présentée ci-dessus) passe alors à $1 + m$ pour $n + m$ variables, où m est le nombre de pas de discretisation pendant les heures creuses.

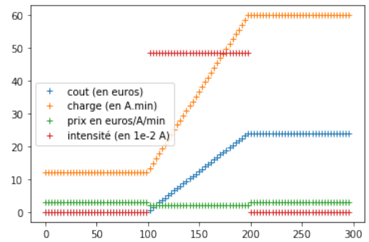
Pour que l'algorithme fonctionne il faut que :
— les lignes de A soient linéairement indépendantes
— c ne soit pas une combinaison linéaire des lignes de A
— avoir une solution réalisable initialement

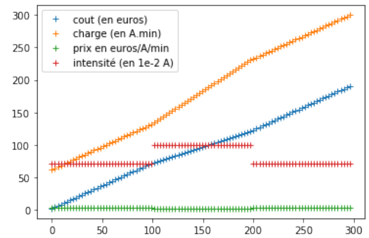
La première et la seconde hypothèse sont mmédiatement satisfaites au vu des matrices A et c.

On construit une intensité réalisable (qui vérifie le contrainte de chargement et est bornée) et on complète à l'aide des variables écarts pour vérifier toutes les égalités pour l'initialisation.

L'algorithme renvoie le résultat attendu rapidement et se comporte comme esperé.

Les deux graphiques suivants illustrent les deux cas, soit les heures creuses suffisent, soit il faut compléter avec des heures pleines.





2.3.4 Algorithme du simplexe

Une autre piste de résolution pour le problème d'un unique véhicule est l'algorithme du simplexe. Il consiste à résoudre des problème de la forme suivante

$$\min_{Ax=b, x\geq 0} c^T x$$

On se place donc dans ce contexte. En réecrivant notre problème qui, rappelons-le, est le suivant

$$\begin{cases} \sum_{k=0}^{n-1} x_i \Delta t_i & = & (c_f - c_0)Q_{max} \\ x_0 & \leq & I_{max} \\ \dots & & \\ x_{n-1} & \leq & I_{max} \end{cases}$$

Où x_i est l'intensité discrétisée au i-ème pas de discrétisation. On le réécrit donc

$$\begin{cases} \sum_{k=0}^{n-1} x_i \Delta t_i & = & (c_f - c_0)Q_{max} \\ x_0 + y_0 & = & I_{max} \\ \dots & & \\ x_{n-1} + y_{n-1} & = & I_{max} \end{cases}$$

Où $\forall i, y_i \geq 0$.

On est donc rammené au problème voulu où le vecteur x est composé des x_i et des y_i et est donc de dimension $2n$

Détaillons maintenant la méthode du simplexe. A est une matrice de taille $(n+1, 2n)$. On pose $m = n+1$ et $p = 2n$.

On note B une sous matrice inversible de la matrice A de taille (m, m).

On note iB les indices des colonnes de A composant B et iN les autres indice.

On pose $x_B = (x_i)_{i \in iB}$ et $x_N = (x_i)_{i \in iN}$.

On fait de même pour le vecteur c^T que l'on décompose en c_B^T et c_N^T.

Une solution quelconque de l'équation $Ax = b$ est x tel que

$$x_B = B^{-1}b$$

et

$$x_N = 0$$

On a ensuite

$$Ax = b$$

qui donne

$$Bx_B + Nx_N = b$$

Soit

$$x_B = B^{-1}b - B^{-1}Nx_N$$

Puis la fonction coût s'écrit

$$c^T x = c_B^T x_B + c_N^T x_N$$

soit en réinjectant l'expression de x_N

$$c^T x = c_B^T B^{-1}b + (c_N^T - c_B^T B^{-1}N)x_N$$

On notera dans toutes la suite le vecteur

$$\tilde{c_N} = c_N^T - c_B^T B^{-1}N$$

Dont les coefficients sont les

$$\tilde{c_{N_i}} = c_{N_i}^T - c_B^T B^{-1}A^i$$

pour $i \in iN$ et où A^i est la i-ème colonne de A.

L'algorithme du simplexe consiste à remplacer les indices de iB par ceux de iN pour arriver à un x minimisant $c^T x$ dont l'extremisation est déterminée par les conditions de KKT.

Pour cela on procède de la manière suivante :

1. On choisit $i \in iN$ tel que $\tilde{c_{N_i}} < 0$ qui va rentrer dans la base iB s'il n'y en a pas la solution est optimale. On prend donc le plus grand i.e

$$i = \arg\max_{k \in iN} \tilde{c_{N_k}}$$

2. On choisit $j \in iB$ qui va sortir. Pour cela on choisit j tel que $x_B = B^{-1}b - B^{-1}A^i t$ s'annule lorsque t augmente on a donc

$$j = \arg_{j \in iB} \min_{(B^{-1}A^i)_{j\in iB}>0} \frac{(B^{-1}b)_j}{(B^{-1}A^i)_j}$$

3. On met à jour iB, iN, B et N puis on recalcule x_N et x_B.
4. Et on recommence ! Jusqu'à ce que $\forall i \in iN, c_{N_i} \geq 0$

Chapitre 3

Gestion de plusieurs véhicules

3.1 Etude du problème

3.1.1 Dépendances des bornes de chargement

La difficulté lorsque plusieurs véhicules se connectent à la station est qu'il n'est pas possible de considérer le chargement des véhicules indépendament. En effet le réseau étant limité il ne peut délivrer à tous les véhicules une puissance maximale à un instant donné. Les bornes sont interdépendantes.

Il y a cepedant la majorité des contraintes qui sont propres aux bornes (contrainte de chargement, de l'intensité passant par la borne, de capacité de la batterie et de présence du véhicule).

3.1.2 Construction d'une solution réalisable

Dans le cas où plusieurs véhicules sont présents, il semble difficile de construire manuellement une intensité respectant les contraintes. En effet il n'est plus possible de se contenter d'utiliser des intensités constantes au risque de dépasser la limitation du réseau.

Dans un premier temps afin de simplifier cette construction et de pouvoir vérifier que l'algorithme fonctionne, on supposera que les véhicules sont présents suffisamment longtemps sur les bornes pour que l'intensité à fournir ne soit pas trop importante (figure 3.1).

3.1.3 Décomposition de la fonction coût et des contraintes

La fonction de coût est décomposable sous la forme :

$$f((x_i^k)_{i,k}) = \sum_k f_k((y_i^k)_i)$$

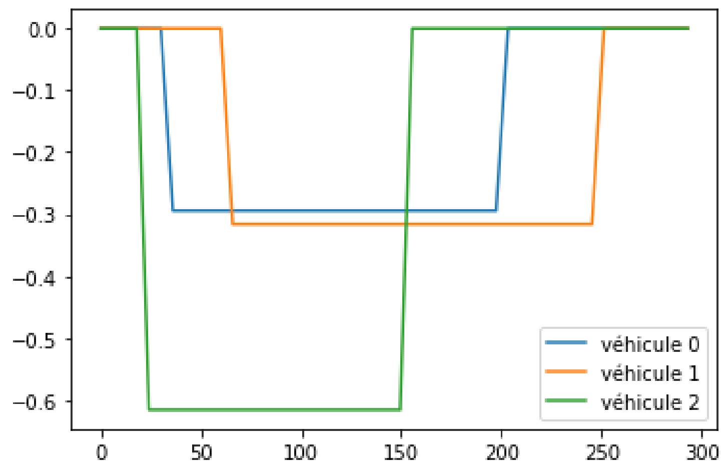


FIGURE 3.1 – Solution réalisable avec intensité constante

avec $f_k((y_i^k)_i) = \sum_i y_i^k(x_{i+1} - x_i)p(x_i)$.

Les variables des f_k sont donc indépendantes. On remarque que les f_k représentent le coût pour l'utilisateur de la voiture k.

De même la plupart des contraintes sont indépendantes entre les voitures (contraintes de chargement, contraintes des bornes). Cependant il reste une contrainte globale qui dépend du chargement de tous les véhicules à un instant i. C'est celle-ci qui rend impossible une résolution indépendante de chaque problème.

Cette contrainte représente la limite du réseau. Elle est représentée comme i inégalités (pour chaque temps) s'écrivant comme des sommes sur k, on peut donc la décomposer comme étant

$$c((y_i^k)_{i,k}) = \sum_k c_k > (-I_{max}, ..., -I_{max})^T$$

avec $c_k = (y_1^k, ..., y_n^k)^T$.

Chacune des variables dont dépendent les c_k sont indépendantes. Il est alors possible d'appliquer la décomposition/coordination par la méthode d'Uzawa pour résoudre le problème d'optimisation.

Il semblerait que le système ne minimisera pas le coût pour chaque utilisateur mais pour le système dans sa globalité, ainsi il est possible qu'un utilisateur paie plus cher lorsque d'autres utilisateurs sont présents.