Ensimag 2A – automne 2015
TDs optimisation
Alejandro Blumentals, Franck Iutzeler, Jérôme Malick

TP 5 – algorithmes d'optimisation convexe II

Objectif : Cette séance a pour but de programmer une version simplifiée d'un algorithme d'optimisation convexe non différentiable efficace : la méthode de faisceaux.

Rappel de cours sur la méthode des faisceaux : Soit $\theta\colon \mathbb{R}^n \to \mathbb{R}, \lambda \mapsto \theta(\lambda)$ une fonction convexe (non différentiable a priori). Rappelons rapidement le principe de la méthode des faisceaux pour minimiser θ (voir les détails en cours). À l'itération k, on dispose du *faisceau* d'information

$$\{(\theta(\lambda^j), g^j) : j = 1, \ldots, k\}.$$

(Dans tout ce qui suit, les indices supérieurs noteront un numéro d'itération.) On définit $\hat{\lambda}^k$ comme le meilleur des λ^j, i.e. $\theta(\hat{\lambda}^k) \leqslant \theta(\lambda^j)$, $j = 1, \ldots, k$. L'itéré suivant est alors obtenu comme solution de

$$\left\{ \begin{array}{lll} \min & r + \dfrac{1}{2}\|\lambda - \hat{\lambda}^k\|^2\,, & (\lambda, r) \in \mathbb{R}^n \times \mathbb{R}\,, \\ & r \geqslant \theta(\lambda^j) + (\lambda - \lambda^j)^\top g^j, & j = 1, \ldots, k\,. \end{array} \right. \tag{1}$$

Pour minimiser θ, la méthode de faisceau rudimentaire s'écrit alors

0. on connaît le faisceau initial $(\theta(\lambda^1), g^1)$ en λ^1 – poser $\hat{\lambda}^1 = \lambda^1$ et $k = 1$;
1. obtenir λ^{k+1} en résolvant le programme quadratique (1)
2. appeler le simulateur pour calculer $\theta(\lambda^{k+1})$ et g^{k+1} ;
3. si θ a diminué, faire $\hat{\lambda}^{k+1} = \lambda^{k+1}$ (sinon ne pas le toucher) ;
4. ajouter la contrainte $k + 1$ au problème quadratique (et modifier l'objectif si besoin)
5. faire $k = k + 1$ et boucler en 1.

Noter que, pour simplifier, on ne se préoccupe pas du test d'arrêt. En pratique, mettre simplement un nombre maximal d'itérations, et prévoir d'afficher les valeurs successives de $\theta(\lambda^k)$ et $\theta(\hat{\lambda}^k)$.

Exercice 1 – Écriture sous forme standard. Pour résoudre le problème quadratique (1), on va utiliser la fonction Scilab `qld`. Faire `help qld`, et mettre (1) sous une forme adaptée (rem : on l'a déjà fait au TD2). Remarquer au passage que le problème quadratique que l'on doit résoudre à l'itération $k + 1$ change peu par rapport à celui de l'itération k : il y a une ligne en plus dans les contraintes et le terme linéaire de la fonction-objectif est différent.

Exercice 2 – Méthode de faisceaux. Programmer une fonction Scilab implémentant la méthode de faisceau ci-dessus. Comme d'habitude, le nom du simuateur sera passé en paramètre.

Exercice 3 – Tests. Faire des expériences avec le simulateur pour le problème d'affectation du TP 4.

Exercice 4 – Tension de ressort. En géneral, l'impact du terme quadratique stabilisateur est modulable par un paramètre t. Concrètement, le problème (1) est donc remplacé par

$$\left\{ \begin{array}{lll} \min & r + \dfrac{1}{2t}\|\lambda - \hat{\lambda}^k\|^2\,, & (\lambda, r) \in \mathbb{R}^n \times \mathbb{R}\,, \\ & r \geqslant \theta(\lambda^j) + (\lambda - \lambda^j)^\top g^j, & j = 1, \ldots, k\,. \end{array} \right. \tag{2}$$

Faire des expériences pour plusieurs valeurs de t. Y a-t-il un bon choix en général ?