UNIVERSITE DE GENEVE — Section de mathématiques

Faculté des sciences — Mercredi 8 novembre 2006

À rendre pour le début de la séance du Mercredi 22 novembre 2006

Optimisation – Série 2 – Corrigé

Exercice 1 a) *Soit $A \in \mathbb{C}^{n\times n}$ une matrice hermitienne, i.e $A^* = A$. Démontrer que toutes les valeurs propres λ de A sont réelles.* Soit $\boldsymbol{x}$ vecteur non nul dans $\mathbb{C}^n$ et λ dans $\mathbb{C}$ tel que $A\boldsymbol{x} = \lambda\boldsymbol{x}$ alors

$$\begin{aligned}(A\boldsymbol{x},\boldsymbol{x}) &= (\lambda\boldsymbol{x},\boldsymbol{x})\\ &= \bar{\lambda}\|\boldsymbol{x}\|^2\end{aligned} \qquad \begin{aligned}(A\boldsymbol{x},\boldsymbol{x}) &= (\boldsymbol{x},A\boldsymbol{x})\\ &= (\boldsymbol{x},\lambda\boldsymbol{x})\\ &= \lambda\|\boldsymbol{x}\|^2\end{aligned}$$

donc $\lambda = \bar{\lambda}$.

b) *Soit Q une matrice unitaire dans $\mathbb{C}^{n\times n}$, i.e. $Q^*Q = I$. Démontrer que*

$$\|Q\boldsymbol{x}\|_2 = \|\boldsymbol{x}\|_2, \qquad \|Q^*BQ\|_2 = \|B\|_2.$$

pour tout vecteur $\boldsymbol{x}$ dans $\mathbb{C}^n$ et toute matrice B dans $\mathbb{C}^{n\times n}$.

$$\begin{aligned}\|Q\boldsymbol{x}\|^2 &= (Q\boldsymbol{x},Q\boldsymbol{x}),\\ &= (Q^*Q\boldsymbol{x},\boldsymbol{x}),\\ &= (\boldsymbol{x},\boldsymbol{x}),\\ &= \|\boldsymbol{x}\|^2.\end{aligned}$$

$$\begin{aligned}\|Q^*BQ\|_2 &= \sup_{\|\boldsymbol{x}\|=1}\|Q^*BQ\boldsymbol{x}\|_2\\ &= \sup_{\|\boldsymbol{x}\|=1}\|BQ\boldsymbol{x}\|_2\\ &= \sup_{\|Q\boldsymbol{x}\|=1}\|BQ\boldsymbol{x}\|_2\\ &= \sup_{\|\boldsymbol{y}\|=1}\|B\boldsymbol{y}\|_2\\ &= \|B\|_2.\end{aligned}$$

Exercice 2 *Un vigneron cultive deux sortes de vignes pour deux vins différents : un blanc de très bonne qualité et un rouge de qualité moindre. Le blanc se vend à 15CHF le litre et le rouge à 11CHF le litre. Il dispose de 400 hectares et peut produire 0.5 hectolitre de vin blanc par hectare et 0.75 hectolitre de vin rouge par hectare. La demande pour le vin rouge est limité à 250 hectolitres. Avec tous les saisonniers, ce vigneron dispose de 4200 heures de travail. Un hectolitre de vin rouge coûte 12 heures de travail et un de blanc 28 heures. Le vigneron souhaite évidemment maximiser ses profits. En faisant un dessin, résoudre le problème d'optimisation linéaire associé.*

Le problème à résoudre est

$$\begin{aligned} 1500B + 1100R &\to \max \\ 2B + \frac{4}{3}R &\leq 400 \\ 28B + 12R &\leq 4200 \\ R &\leq 250 \\ B &\geq 0 \\ R &\geq 0 \end{aligned}$$

Cela donne le dessin de la figure 1.

Il faut donc produire 250 hectolitres de vin rouge et 33.333 hectolitres de vin blanc.

Exercice 3 a) *Implémenter un algorithme de recherche de la meilleure permutation par la méthode de force brute.*

```
function s=PartitionBruteForce(G)
% PARTITIONBRUTEFORCE find best two frequency partition
%   s=PartitionBruteForce(G); finds for a given link gain matrix G
%   the best partition into four submatrices, such that the maximum
%   of the largest eigenvalues of the two diagonal blocks is minimized.
%   The result is given by the binary index s for one of the submatrices.

n=size(G,1);
for k=1:2^(n-1)-1
  s=logical(double(dec2bin(k,n))-'0');
  r(k)=max([max(eig(G(s,s))) max(eig(G(~s,~s)))]);
end
[rmin,k]=min(r);
s=logical(double(dec2bin(k,n))-'0');
```

b) *Implémenter l'algorithme de recherche expliqué en cours basé sur la norme de Frobenius.*

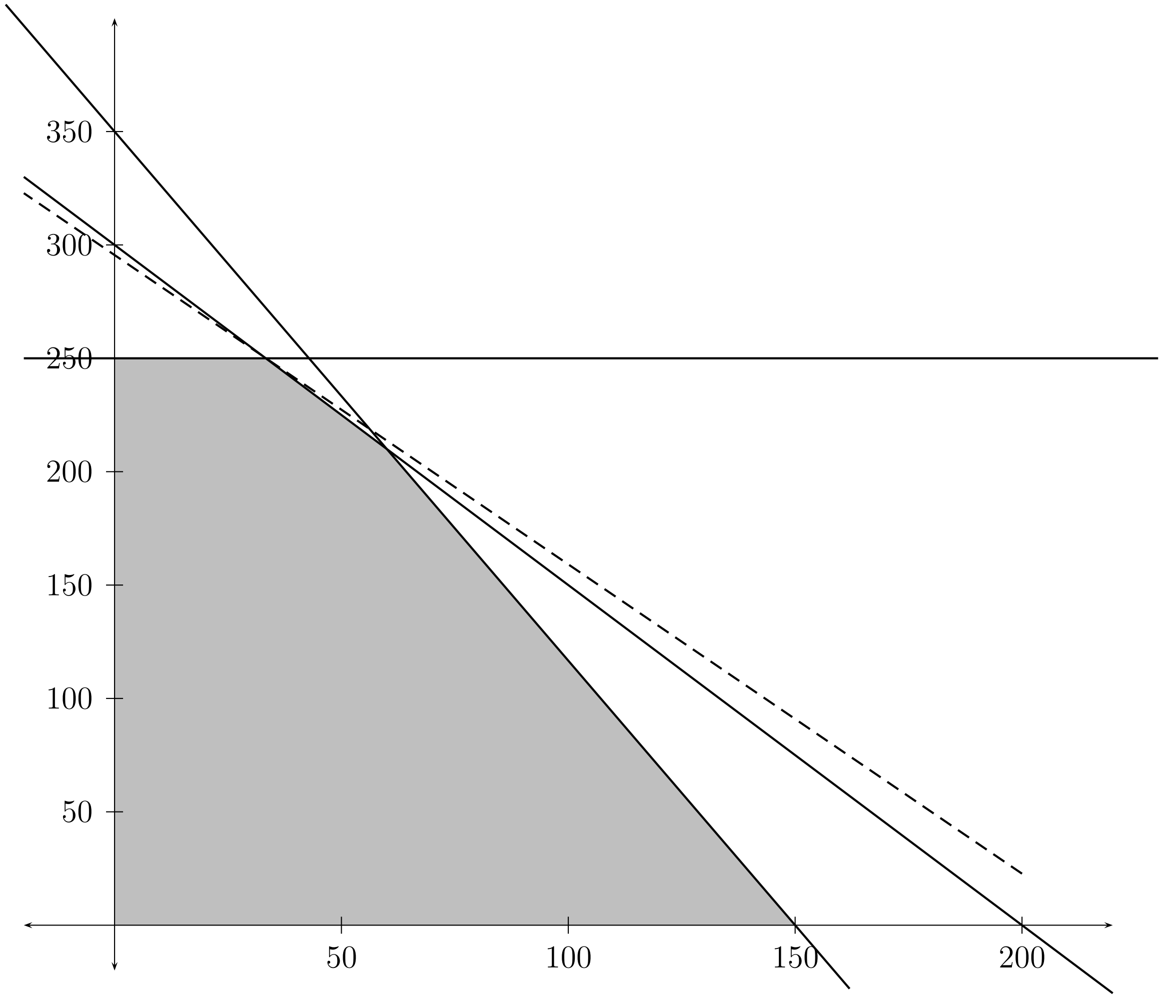

Fig. 1 – Optimisation linéaire

```
function s=PartitionSpectral(G)
% PARTITIONSPECTRAL find approximate two frequency partition
%   s=PartitionSpectral(G); finds for a given link gain matrix
%   G an approximation to the best partition into four submatrices,
%   such that the maximum of the largest eigenvalues of the two
%   diagonal blocks is minimized.  The result is given by the binary
%    index s for one of the submatrices.

n=size(G,1);
G2=G.^2;
[V,E]=eig(G2);
```

```
[dummy,k]=min(diag(E));
[v,id]=sort(V(:,k));
for p=1:n-1
  r(p)=max([max(eig(G(id(1:p),id(1:p))))
max(eig(G(id(p+1:n),id(p+1:n))))]);
end
[rmin,p]=min(r);
s=logical(dec2bin(sum(2.^(id(1:p)-1)),n)-'0');
```

c) *Tester les deux algorithmes sur les matrices* `G1` *et* `G2` *obtenues par ...*

```
> oldseed=rand('seed');
> rand('seed',900);
> [G1,x1,y1]=GenerateProblem(4,4,0.1);
> print -depsc G1graph
> [G2,x2,y2]=GenerateProblem(4,4,0.4);
> print -deps G2graph

> s1b=PartitionBruteForce(G1);
> find(s1b)
        ans = 2 4 5 7 10 12 13 15
> plot(x1(s1b),y1(s1b),'b+',x1(~s1b),y1(~s1b),'bo')
> print -depsc SolPb1brute

> s1s=PartitionSpectral(G1);
> find(s1s)
        ans = 1 3 6 8 9 11 14 16
> plot(x1(s1s),y1(s1s),'b+',x1(~s1s),y1(~s1s),'bo')
> print -depsc SolPb1spectral


> s2b=PartitionBruteForce(G2);
> find(s2b)
        ans = 2 4 5 7 10 12 13 15
> plot(x2(s2b),y2(s2b),'b+',x2(~s2b),y2(~s2b),'bo')
> print -depsc SolPb2brute

> s2s=PartitionSpectral(G2);
> find(s2s)
        ans = 4 5 7 10 12 14 15
> plot(x2(s2s),y2(s2s),'b+',x2(~s2s),y2(~s2s),'bo')
> print -depsc SolPb2spectral
```

```
> rand('seed', oldseed);
```

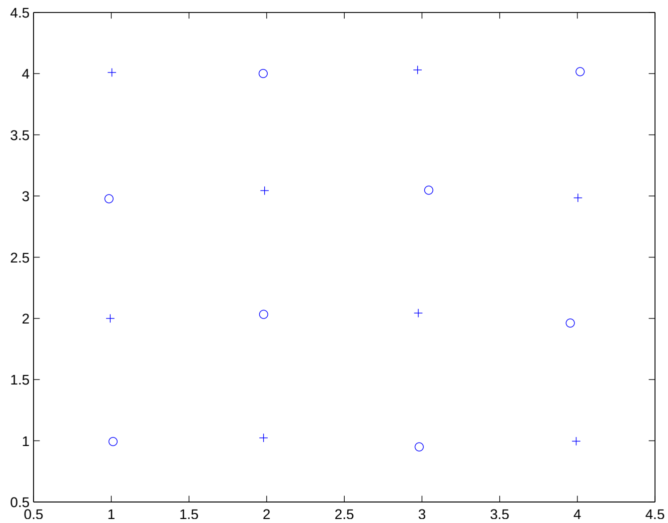

FIG. 2 – Problème 1 : Résolution par force brute

Evaluation du cours d'optimisation

- Les exercices : Les séries d'exercices rendues en retard seront comptées comme non rendues (*i.e.*, note 1 sur 6) dans le calcul de la note finale.
- Un examen oral durant la session d'examens sur le cours.

La note finale est de : $\frac{1}{5}$(exercices.) + $\frac{4}{5}$(note examen oral).

Assistant :	Kévin Santugini
Email :	`Kevin.Santugini@math.unige.ch`
Page web :	`http://www.unige.ch/~santugin/index.php?page=enseignement`

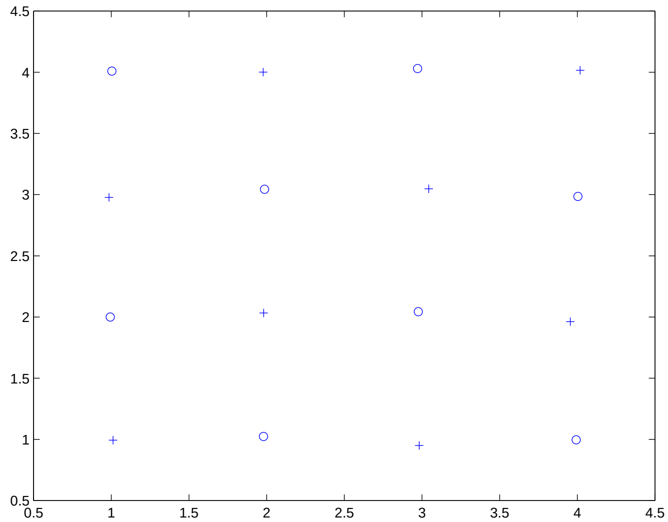

FIG. 3 – Problème 1 : Résolution par méthode spectrale

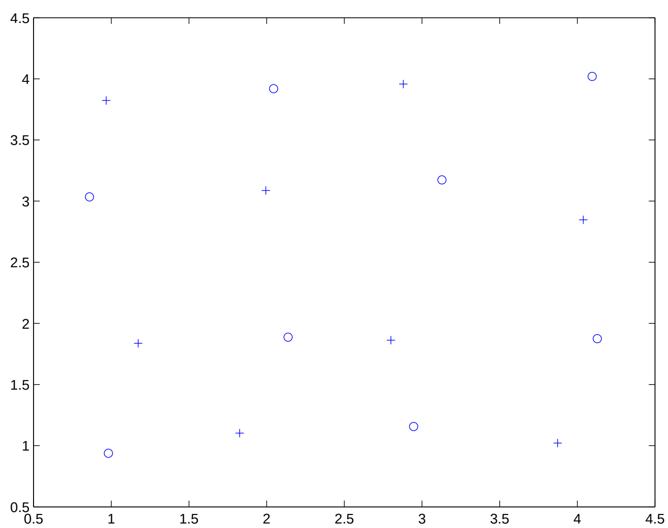

FIG. 4 – Problème 2 : Résolution par force brute

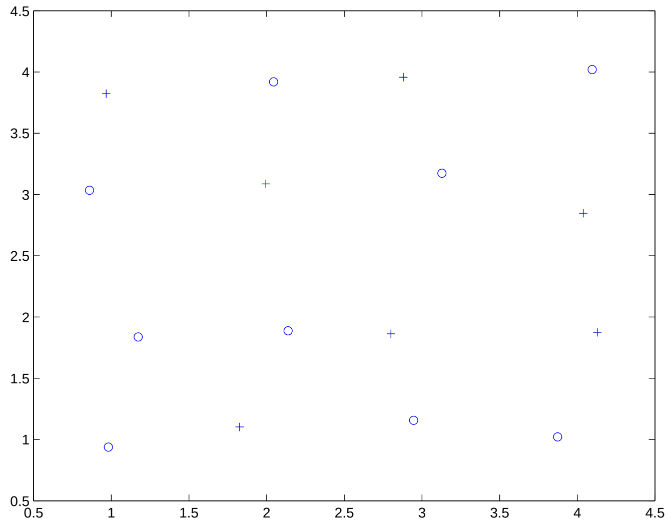

FIG. 5 – Problème 2 : Résolution par méthode spectrale