Optimisation Combinatoire 2A
Premier cours

Zoltán Szigeti

Ensimag
INP Grenoble

Enseignant-chercheur

Enseignant

1. SZIGETI, Zoltán
2. Professeur à l'Ensimag.
3. Adresse e-mail : zoltan.szigeti@imag.fr
   - Je déteste les mails !
   - Écrivez si c'est une question de vie ou de mort !

Chercheur:

1. Recherche au Laboratoire G-SCOP.
   - Laboratoire des Sciences pour la Conception, l'Optimisation et la Production de Grenoble.
2. Sujets de recherche :
   - Optimisation Combinatoire,
   - Théorie des graphes,
   - Connexité (fiabilité d'un réseau),
   - Couplages (affectation des ressources).

Cours : Optimisation Combinatoire

Cours

1. optionnel en MMIS (65 étudiants en 2020 (58 en 2019, 48 en 2018)).

Optimisation Combinatoire

1. Partie optimisation discrète de la RO, consiste à "Trouver une meilleure solution dans un très grand ensemble de possibilités."
   - Exemples vus en 1A :
     - Plus courts chemins,
     - Arbre couvrant de coût minimum.
2. Résultats structurels sur les graphes,
   - Exemples vus en 1A :
     - un sous-chemin d'un plus court chemin est un plus court chemin.
3. Algorithmes efficaces sur les graphes,
   - Exemples vus en 1A :
     - Bellmann, Dijkstra, Floyd-Warshall pour plus courts chemins,
     - Kruskal (glouton) pour arbre couvrant de coût minimum.

Planning

Sujets à traiter :

1. Chemins de coût minimum dans un graphe orienté,
2. Flot de valeur maximum dans un réseau,
3. Flot de coût minimum dans un réseau,
4. Couplage de coût maximum dans un graphe biparti,
5. Couplage de cardinal maximum dans un graphe quelconque.

Citation au sujet du flot :

"*But anyone who has experienced flow knows that the deep enjoyment it provides requires an equal degree of disciplined concentration.*"

Mihály Csikszentmihályi

1. Problèmes de plus courts chemins

Plus courts chemins :

- Graphes sans circuits (algorithme de Bellmann),
- Coûts positifs (algorithme de Dijkstra),
- Cas général : algorithme de Floyd-Warshall,
- Détection de circuits absorbants (algo. de Ford-Fulkerson-Dantzig).

et ses Applications :

1. Sauvegarder les données d'un GPS : Minimiser le coût du stockage et l'erreur de précision.
2. Allocation des contrôles sur une chaîne de production : minimiser le coût pour avoir les objets sans défaut.
3. Problème du sac à dos,
4. Transport maritime,
5. Système d'inéquations formées de différences.

Application 3.

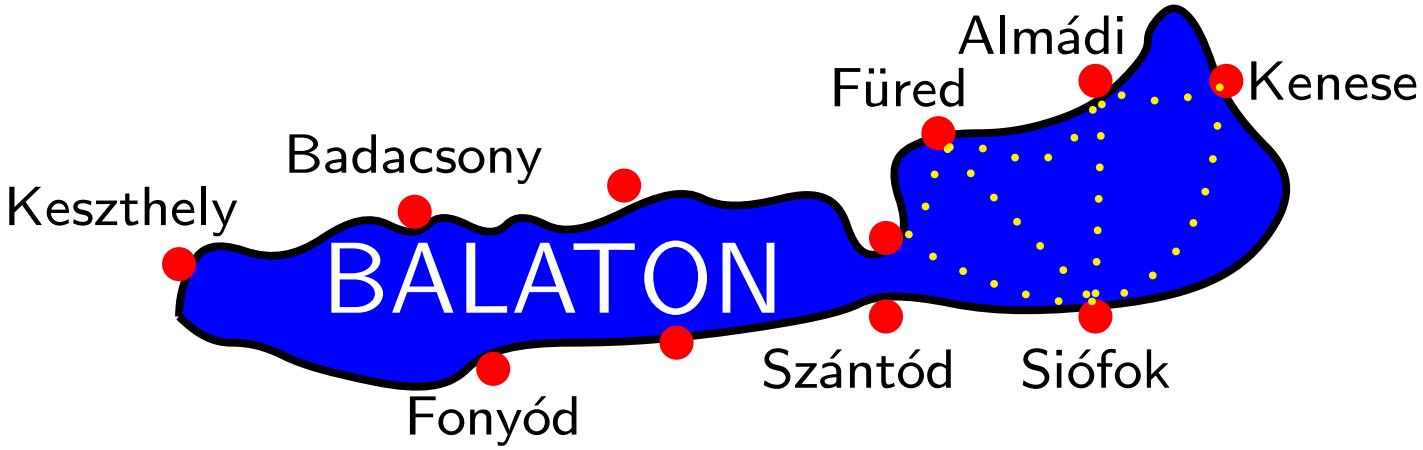
Problème du sac à dos

1. Etant donnés
   1. un sac à dos et
   2. des objets,
2. il faut choisir quelques uns des objets de telle sorte que
   1. on puisse les mettre dans le sac à dos et
   2. la somme de leurs utilités soit maximum.

Application 4.

Transport maritime

Une compagnie maritime offre des transports réguliers entre les différentes villes autour du Lac Balaton en Hongrie. On sait que le voyage de la ville i à la ville j dure t_{ij} minutes et rapporte p_{ij} Forints. Le directeur de la compagnie doit décider s'il existe une tournée C (un circuit) dont le rapport moyen $\frac{\sum_{ij \in C} p_{ij}}{\sum_{ij \in C} t_{ij}}$ est supérieur à une valeur fixée.



Application 5.

Système d'inéquations formé de différences

Étant donné un système d'inéquations dont toutes les inégalités sont de la forme $x_{j_k} - x_{i_k} \leq b_k$, décider s'il existe une solution.

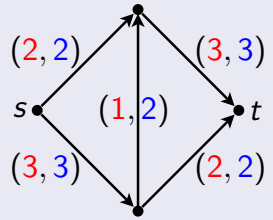
1. Si oui, trouver une telle solution et
2. si non, trouver les contraintes contradictoires.

$$\begin{aligned}
x_3 - x_4 &\leq 5 \\
x_4 - x_1 &\leq -10 \\
x_1 - x_3 &\leq 8 \\
x_2 - x_1 &\leq -11 \\
x_3 - x_2 &\leq 2
\end{aligned}$$

2. et 3. Problème du flot dans un réseau :

Exemples :

- Problème des chemins arc-disjoints,
- Problème des camions,
- Problème des tuyaux,
- Problème de transport.



Flots :

- Flot de valeur maximum
  - Théorèmes min-max de Ford-Fulkerson, Hoffmann,
  - Algorithme d'Edmonds-Karp et ses diverses variantes (circulation),
- Flot de coût minimum
  - Algorithme Suppression des circuits absorbants.

4. et 5. Problème du couplage dans un graphe

Couplages :

- Couplage de cardinal maximum dans un graphe biparti :
  - Théorèmes de Frobénius, Hall, Kőnig,
  - Exemple : Problème de couples de danseurs.
- Couplage de coût maximum dans un graphe biparti :
  - Algorithme : Méthode Hongroise,
  - Exemple : Affectation du personnel de profit maximum.
- Couplage de cardinal maximum dans un graphe quelconque :
  - Théorèmes de Tutte,
  - Algorithme d'Edmonds,
  - Exemple : Affectation des pilotes.

Applications

Applications

1. Segmentation des images : Diviser l'image en régions cohérentes,
2. Ordonnancement d'avions : Minimiser le nombre d'avions en effectuant tous les vols d'une compagnie aérienne,
3. Calcul distribué sur un ordinateur à deux processeurs,
4. Exploitation minière à ciel ouvert,
5. Arrondi de matrice,
6. Reconstruire le ventricule gauche du coeur à partir de radiographies,
7. Détections des objets.

Application 3.

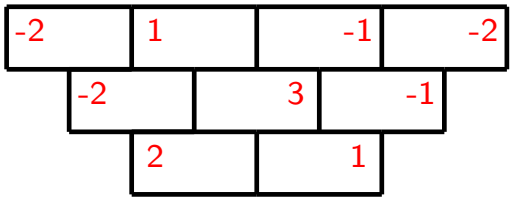
Calcul distribué sur un ordinateur à deux processeurs

- On voudrait distribuer les modules d'un programme sur un ordinateur à deux processeurs
  - en minimisant le coût total
    - de traitement et
    - de communication entre processeurs.
- On connait par avance
  - d'une part pour chaque module le coût de son traitement sur chacun des deux processeurs et
  - d'autre part pour chaque paire de modules le coût de communication entre processeurs
    - si ces deux modules sont associés aux processeurs différents.
- On va modéliser ce problème par les flots.

Application 4.

Exploitation d'une mine à ciel ouvert

- Le directeur d'une entreprise voudrait exploiter une mine à ciel ouvert
  - en procédant à des extractions par blocs et
  - en maximisant le profit.
    - L'extraction d'un bloc n'est possible que si les blocs situés juste au-dessus de ce bloc sont déjà extraits.
- On connait par avance pour chaque bloc le profit de son extraction.
  - Cette valeur peut être positive ou négative, elle dépend du coût
    - de l'extraction du bloc et
    - de la richesse de son contenu.
- On va modéliser ce problème par les flots.



Application 5.

Arrondissement de matrice

Étant donné une matrice réelle A, il faut arrondir (vers le bas/vers le haut)

1. chaque élément a_{ij} de A,
2. la somme de chaque ligne a_i $(1 \leq i \leq m)$,
3. la somme de chaque colonne a^j $(1 \leq j \leq n)$,

de manière à ce que la somme des éléments arrondis de chaque ligne (et colonne) soit égale à l'arrondi de la somme de la ligne (de la colonne).

On va modéliser ce problème par une circulation.

0.33	0.44	1.24	2.01
1.53	0.41	2.32	4.26
1.86	0.85	3.56	

pas correcte

0	0	1	2
2	0	2	4
2	1	4	

correcte

1	0	1	2
1	1	2	4
2	1	3	

Application 6.

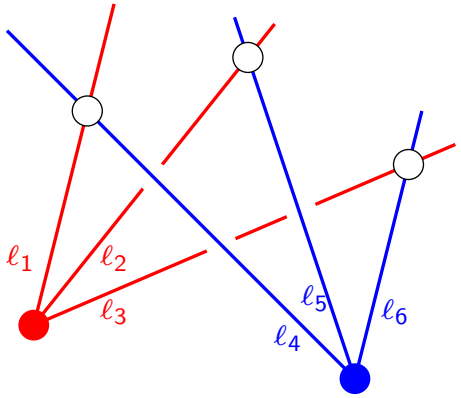
Reconstruire le ventricule gauche du coeur à partir de radiographies

- Pour diagnostiquer une possible maladie de coeur d'un patient, on doit voir la forme 3-dimensionnelle du ventricule gauche du coeur.
- Évidemment, le problème se réduit à plusieurs problèmes 2-dimension.
- En prenant les rayons X, on ne peut pas avoir directement l'image 2-dimensionnelle ; on aura plutôt deux projections 1-dimensionnelles de la densité de la région sur les deux axes.
- Le problème revient donc à construire une matrice binaire dont la somme de chaque ligne et de chaque colonne est connue.
- On suppose que pour chaque pixel de l'image, on connait une probabilité p_{ij} que le pixel appartient à la région cherchée.
- On va modéliser ce problème par un flot de coût minimum.

Application 7.

Détections des objets

- On veut trouver les positions exactes de n objets dans l'espace 3-dimensionnel à l'aide de deux détecteurs infrarouges fixés.
- Chaque détecteur nous fournit n droites sur lesquelles les objets se trouvent.
- Ces $2n$ droites donnent théoriquement les positions exactes des n objets.
- À cause de problèmes techniques on n'a que des droites approximatives.
- On sait que deux droites correspondant au même objet ont une distance euclidienne très petite.
- On va modéliser ce problème avec les couplages.



Books

Livres pour approfondir

1. **Ahuja, Magnanti, Orlin,** *Network flows; Theory, Algorithms and Applications,*
2. **Cook, Cunningham, Pulleyblank, Schrijver,** *Combinatorial Optimization,*
3. **Frank,** *Connections in Combinatorial Optimization,*
4. **Korte, Vygen,** *Optimisation Combinatoire; Théorie et Algorithmes,*
5. **Lovász, Plummer,** *Matching Theory,*
6. **Schrijver,** *Combinatorial Optimization; Polyhedra and Efficiency,* 3 volumes.

Continuation en 2A :

pour tout le monde :

1. Projets filés, obligatoire, 2S
   1. Projet de spécialité, optionnel,
      - sur l'application de la théorie des jeux avec moi,
      - ? groupes en 2021, (1 en 2020, 4 en 2019)
   2. IRL (Introduction à la recherche en laboratoire), optionnel,
      - info sur ENSIWIKI,
      - pas de garantie : succès ou échec,
      - un ou deux étudiant(s) sur la théorie des graphes avec moi.

pour la filière MMIS :

1. Recherche opérationnelle avancé, optionnel, 2S
   - Problèmes de Localisation, Gestion des stocks, Job-scheduling, Problèmes de tournées de Véhicules, Pilotage de robots, Yield management, Network design

Continuation en 3A :

pour tout le monde :

1. PFE (Projet de fin d'étude), obligatoire, 2S

pour la filière MMIS :

1. M2 Operations Research, Combinatorics and Optimization (ORCO)
   1. pour carrière académique :
      - Enseignant-chercheur (comme moi),
      - Chercheur.
   2. pour carrière industrielle :
      - RO = Aide à la décision
      - Conseiller : Air France, EDF,
      - R&D : Air France, Amadeus, Bouygues Telecom, France Telecom.
2. Thèse (éventuellement, plus tard).

Stages en ORCO :

Stages 2018 :

1. Decision support system for carpooling to reduce car traffic in Grenoble area,
2. Modeling the activity of maintenance for decision aiding,
3. Optimization of parcels delivery,
4. Optimizing rolling stock planning on defined train services,
5. Stochastic optimization for air traffic conflict resolution,
6. Technical and economic optimization of a renewable energy production plant coupled with an energy storage system,
7. The robust kidney exchange problem.

Organisation :

Déroulement :

- 1 CM + 2 groupes de TD : G1 et G2.
- 11 semaines.

séances :

- CM : Mardi, 08h15-09h45 Amphi E,
- TD G1 : Mercredi, 11h15-12h45 D111,
- TD G2 : Mercredi, 09h45-11h15 D111.

Permanence :

- pas seulement soutien,
- pas pour réviser l'exam,
- ?

Remarques diverses :

A savoir :

1. il faut apprendre
   - toutes les définitions,
   - tous les théorèmes,
   - tous les algorithmes,
2. Il y aura une feuille d'exercices pour chaque sujet :
   - il faut connaître les exercices aussi.

Remarques diverses :

A faire :

- travailler : 6h de travail personnel,
- rédiger les exercices,
- expliquer les solutions : je vais demander des volontaires.

Transparents :

- qui seront disponibles sur ma page web plus tard,
- que vous pouvez utiliser pendant l'examen,
- mais qui ne remplacent pas du tout vos notes, les CM et les TD,
- où on apprendra comment résoudre les problèmes.

Note

- Il n'y pas de TP,
- La note = note de l'examen (de 3h).