Introduction aux problèmes d'optimisation et à la modélisation

Résolution de Problèmes d'Optimisation
Master 1 informatique I2L / WeDSci

Sébastien Verel
verel@univ-littoral.fr
http://www-lisic.univ-littoral.fr/~verel

Université du Littoral Côte d'Opale
Laboratoire LISIC
Equipe OSMOSE

Plan

Exemple de problèmes d'optimisation

Problèmes prototypes :

- max-SAT (vecteurs binaires)
- Paysages NK (vecteurs binaires)
- Coloriage de graphe (vecteurs d'entiers)
- Ordonnancement de taches (permutations)
- Voyageur de commerce (permutations)
- Problème d'affectation quadratique (permutations)
- Commande de pilotage du cœur d'une centrale nucléaire (expensive)

Problème de Satisfiabilité (SAT)

Premier problème démontré NP-difficile [Cook, 1971] :
Pas d'algorithme de complexité polynomiale pouvant résoudre ce problème (à moins que N=NP)

Définition du problème de Satisfiabilité (SAT)

- n variables booléennes : $(x_1, x_2, ..., x_n)$ avec $x_i \in \{0, 1\}$
- m clauses avec k_j variables (ou négation) par clause :

$$C_j = x_{j_1} \vee \neg x_{j_2} \vee \ldots \vee x_{j_{k_j}}$$

- Formule logique en forme normale conjonctive :

$$\bigwedge_{j=1}^{m} C_j$$

Existe-t-il une assignation des variables telle que la formule soit vraie ?

Exemple de formule 3-SAT :
$(x_1 \vee \neg x_2 \vee x_3) \wedge (x_4 \vee x_2 \vee \neg x_3) \wedge (\neg x_4 \vee \neg x_1 \vee \neg x_3)$

De SAT au problème d'optimisation Max-SAT

Problème d'optimisation Max-SAT

Problème de maximisation définit à partir d'un problème SAT $(n, m, (C_j)_{j \in \{1,\dots,m\}})$:

- Espace de recherche : $\mathcal{X} = \{0,1\}^n$
  l'ensemble des chaines binaires de longueur n
- Fonction objectif (à maximiser) :

$$f(x) = \sum_{j=1}^{m} \delta_j$$

avec

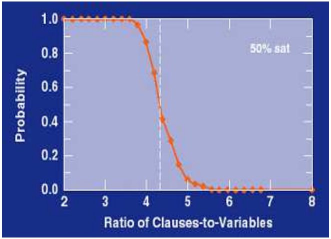
$$\delta_j = \begin{cases} 1 & \text{si } C_j \text{ est vraie} \\ 0 & \text{sinon} \end{cases}$$

L'assignement x satisfait la formule du problème SAT
si et seulement si $f(x) = m$.

Propriétés physiques (!) de SAT

Gent, Ian P., and Toby Walsh. "The SAT phase transition." ECAI. Vol. 94.1994.

Probabilité qu'une instance aléatoire de 3-SAT soit satisfiable :



Transition de phase suivant :

$$\alpha = \frac{m}{N} = \frac{\text{nombre de clauses}}{\text{nombre de variables}}$$

- Si $\alpha < \alpha_c$ alors forte probabilité qu'une instance aléatoire ait une solution
- Si $\alpha_c < \alpha$ alors faible probabilité qu'une instance aléatoire ait une solution

... analogie avec la transition de phase entre eau et glace autour de 0°C.

α_c temp. critique

Applications et solvers SAT

Applications

Beaucoup de problèmes peuvent se modéliser sous forme d'une formule logique :

- Planification :
  "Faire traverser le loup, la chèvre et les choux sur un bateau avec son fermier..."
- Vérification de système, *model checking* :
  Voir feuille td
- Traduction d'un autre problème d'optimisation (coloriage, graphe, ordonancement, etc.)
  "binarisation" des états et des propriétés attendues.

Solvers

Il existe de très nombreux solvers très efficaces openSource (miniSAT, etc.) ou non (ilog d'IBM, etc.) : cf. SAT competition.

Problèmes des paysages NK

Modélisation des interactions des acides animés d'une protéine pour montrer que plus les interactions sont importantes, plus le nombre de variantes stables est important.

Paramètres

- n : longueur de la chaine binaire,
- $k \in [0, n-1]$ (entier) : degré d'interaction
- Graphe d'interaction : chaque variable i est reliée à $i_1, i_2, \ldots, i_k$
- Energie des interactions : pour chaque variable i, une fonction donne la valeur de l'énergie $c_i : \{0,1\}^{k+1} \to [0,1]$

Définition du problème d'optimisation

- Espace de recherche : $\mathcal{X} = \{0,1\}^n$
- Fonction objectif : $f(x) = \frac{1}{n} \sum_{i=1}^{n} c_i(x_i, x_{i_1}, \ldots, x_{i_k})$

exemple $n = 4$ $k = 2$

$$x = 0110$$

$x_1 x_2 x_4$	c_1
000	0.9
001	0.6
010	0.1
011	0.2
...	...

$x_1 x_2 x_3$	c_2
000	0.4
001	0.8
010	0.3
011	0.2
...	...

$x_2 x_3 x_4$	c_3
000	0.2
...	...
101	0.9
110	0.1
111	0.5

$x_1 x_2 x_4$	c_4
000	0.1
001	0.2
010	0.8
011	0.0
...	...

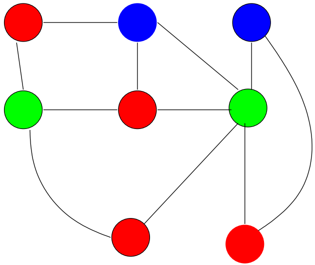
$$\begin{aligned} f(x) &= \frac{1}{4}(c_1(010) + c_2(011) + c_3(110) + c_4(010)) \\ &= \frac{1}{4}(0.1 + 0.2 + 0.1 + 0.8) \\ &= 0.3 \end{aligned}$$

Résultats : plus k est grand, plus le paysage est rugueux et difficile à optimiser...
Application : apprentissage de la dernière couche d'un réseau de neurones artificiels

Coloration de graphe

Données

- Graphe non orienté $G = (S, A)$ avec S ens. de n sommets et $A \subset S^2$ ens. des arêtes.
- Ens. de k couleurs : $\{1, \ldots, k\}$



Coloration : assignation d'une couleur à chaque noeud

But : Trouver une coloration telle que tous les nœuds voisins n'aient pas la même couleur.

Définition du problème d'optimisation

- Espace de recherche :
  $$\mathcal{X} = \{1, \ldots, k\}^n$$
- Fonction objectif (à minimiser) :
  $$f(x) = \sum_{i=1}^{n} \sum_{j \in \mathcal{V}(i)} \delta_{x_i = x_j}$$
  où $\mathcal{V}(i)$ est l'ensemble des nœuds voisins du nœud i,
  $\delta_{x_i = x_j} = 1$ ssi $x_i = x_j$ et 0 sinon.

Coloration de graphe

Affectation de fréquences

Exercice

Des émetteurs radio sont disposés dans l'espace géographique. Pour éviter les interférences, ils ne peuvent diffuser sur la même fréquence si leur distance est inférieure à une constante D.

Traduire ce problème en problème de coloration de graphe.

Sudoku

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Questions :

- Calculer le nombre de grilles (solutions) possibles au sudoku.
- Calculer le temps de traitement par l'ensemble des ordinateurs sur Terre de ces solutions à l'aide d'un algorithme "itératif".

sudoku : coloration de graphe ?

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Exercice

Traduire le problème de sudoku en un problème de coloration de graphe.

Problème d'ordonnancement : Job Scheduling Problem

Ordonnancement de taches sur une chaine de production composée de machines. Cas le plus simple : Flow Shop Scheduling (NP-difficile).

Données

- Ens. de n taches (job) : $J = \{j_1, j_2, \ldots, j_n\}$
- Ens. de m machines : $M = \{M_1, \ldots, M_m\}$
- Temps d'exécution de la tache i sur la machine j : $\delta_j(i)$

But : Minimiser la date de fin de l'exécution de toutes les taches

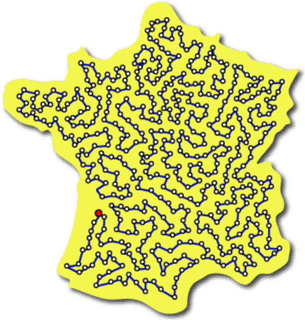
Définition du problème d'optimisation

- Espace de recherche :
  $\mathcal{X} = \mathcal{S}_n$ (ens. des permutations de taille n)
- Fonction objectif (à minimiser) : makespan
  $f(x) = \max\{C_1, C_2, \ldots, C_n\}$
  où C_i est la date de fin du job i sur la machine m.

D'autres critères sont possibles (Tardiness, avg. completion times, etc.)

Voyageur de commerce - Traveling Salesperson Problem (TSP)

But : Trouver le parcours le plus court passant par toutes les villes et revenant au point de départ.



Données

- n : nombre de villes
- (d_{ij}) matrice de dimension $n \times n$, des distances entre les villes i et j.

Définition du problème d'optimisation

- Espace de recherche :
  $\mathcal{X} = \mathcal{S}_n$ (ens. des permutations)
- Fonction objectif (à minimiser) :

$$f(\sigma) = \sum_{i=1}^{n-1} d_{\sigma_i, \sigma_{i+1}} + d_{\sigma_n, \sigma_1}$$

Problème NP-difficile. Solvers : Concorde TSP Solver, Lin-Kernighan heuristics, etc. (cf. TSPLib)

Problème d'affection quadratique - Quadratic Assignment Problem

QAP : Problème d'affectation de *n* objets à *n* emplacements.

Source image : `http://www.collectorchic.com`

Flux connu entre toutes les paires d'objets.

Distance connues entre toutes les paires d'emplacement

But : Minimiser la somme du produit du flux par la distance correspondante

Applications : répartition de bâtiments ou de services, affectation des portes d'aéroport, placement de modules logiques, claviers...
Problème NP-difficile. Définition : voir le TD01.

Optimisation de la commande de pilotage d'un Réacteur à Eau Pressurisée

Exemple d'un travail inter-disciplinaire de modélisation et d'optimisation
afin de proposer de nouvelles façons de piloter le cœur d'une centrale nucléaire.

Optimisation du pilotage d'un Réacteur à Eau Pressurisée dans le cadre de la transition énergétique à l'aide d'algorithmes évolutionnaires

Thèse de Mathieu Muniglia (CEA) - dont ces slides sont issus
avec J.-C. Le Pallec, J.-M. Do, H. Grard, S. David.

Contexte

- Forte augmentation de la part des énergies intermittentes :
  éolien, solaire - 5% en 2013, 30% selon l'ademe en 2030.
- Les variations peuvent être compensées par :
  une gestion du réseau, le stockage ou les autres sources,
  i.e. le **nucléaire** dans le contexte français (50% en 2030).

Objectif

Adapter les centrales nucléaires à cette nouvelle configuration, afin qu'elles puissent compenser au mieux les variations

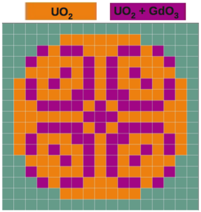
Méthodologie : Modéliser / Simuler / Optimiser (MSO)

Méthodologie

- **Modéliser** un réacteur nucléaire :
  modélisation multiphysique (neutronique,
  thermodynamique, thermomécanique)
  multi-échelle (échelle du coeur au système complet)
- **Simuler** le modèle par un calcul :
  Définir le schéma de calcul,
  développer les outils informatiques
- **Optimiser** certains paramètres de contrôle
  afin de permettre la manoeuvrabilité du réacteur :
  Algorithme évolutionnaire

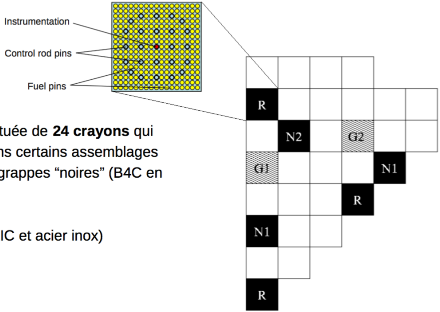
Modèle du réacteur (REP 1300MV)

Source : Mathieu Muniglia



Representation of a reactor core

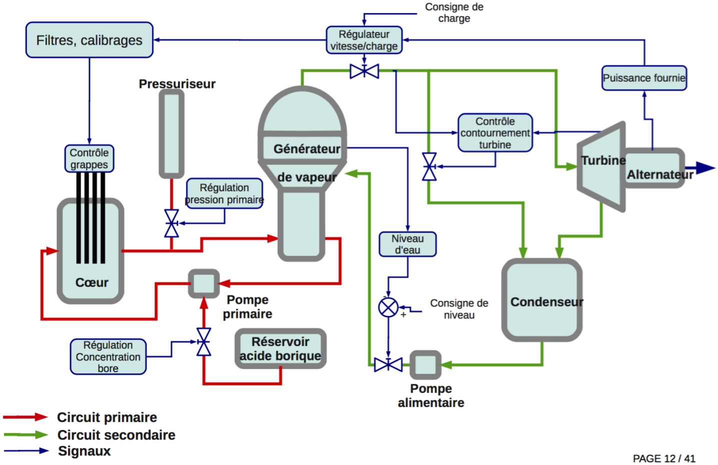
- **PWR 1300** (Electrical power : 1300MW, Thermal power : 3800MW)
- **193** assemblies (120 UO2, 73 UO2+GdO3)
- Dimensions : diameter = 3m, height = 4,5m



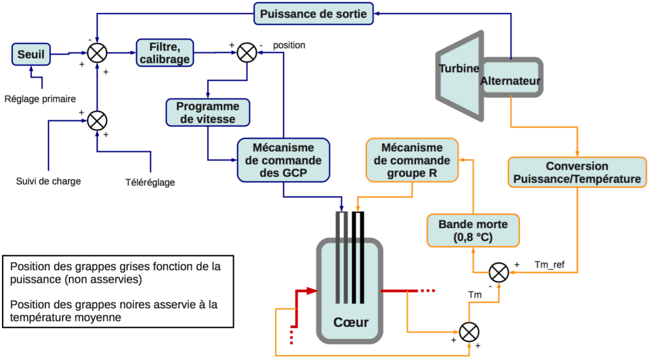
- Chaque grappe de contrôle et constituée de **24 crayons** qui s'insèrent dans des tubes guides dans certains assemblages
- **Régulation de température** (R) : 9 grappes "noires" (B4C en haut, AIC en bas)
- **Compensation de puissance** :
  - 4 grappes "grises" G1 (AIC et acier inox)
  - 8 grappes "grises" G2
  - 8 grappes "noires" N1
  - 8 grappes "noires" N2

Schéma simplifié d'une centrale

Source : Mathieu Muniglia



Chaine de régulation des barres de commande
Source : Mathieu Muniglia



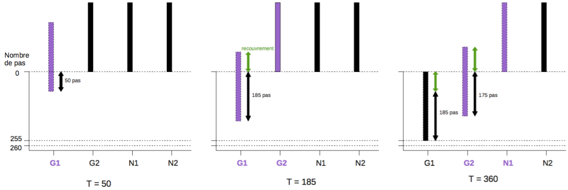
Modèles et couplage : "best estimate" puis dégradé pour réduire temps de calcul

Cœur		
Neutronique	Thermohydraulique	Combustible
Géométrie : 3D, 2D, 1D ; quart/huitième, nb de ss-assemblages Solveur 3D : **SN**, SPN, Diffusion Nombre de groupes neutronique : 2, **8**	Géométrie : **3D**, multi1D Type : stationnaire, **transitoire** Code de calcul : Apollo, Cathare Conditions limites : Te=f(P) (connue)	Calcul : **thermo-mécanique**, thermique Type : stationnaire, **transitoire**

Système	
Thermohydraulique	Composants
Géométrie : 3D, 1D Fluide : monophasique, diphasique	Injection de bore : modélisation opérateur, calcul de bore critique Couplage cœur : couplage GV simplifié puis introduction des composants du secondaire : condenseur, GCT

(modèle **best-estimate**/adopté)

Paramètres de la position des barres de contrôle



Paramètres (11 variables discrètes)

- Recouvrement : nombre de pas entre 2 insertions de barre
- Programme de vitesse : vitesse nominale (pas/min), bande morte, bande de manœuvre groupe R

Critères à optimiser

- Hétérogénéité du coeur relatif à l'axial offset
- Volume de bore

Quelques considérations sur ce travail

Interaction multi-disciplinaire

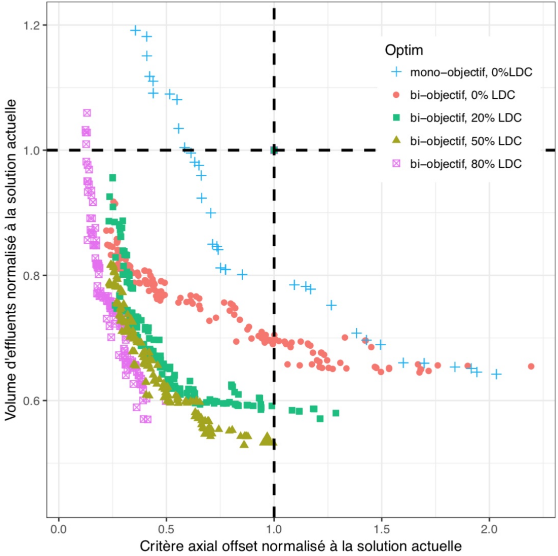
- Définir un modèle et une simulation est difficile :
  Connaissances expertes nécessaires
- Logiquement l'optimisation commence lorsque $(\mathcal{X}, f)$:
  Mais participation à la modélisation nécéssaire :
  - comprendre le problème,
  - conseil dans la modélisation (variable, critères, etc.),
  - expliquer ce que l'on attendre comme résultat,
  - regard critique sur les méthodes, etc.
- Résultats dans les deux domaines (physique nucléaire et optimisation) :
  Publications dans les deux domaines

Quelques considérations sur ce travail

Intérêt du point de vue optimisation stochastique

- Problème d'optimisation combinatoire black-box multiobjectif
- Temps de calcul d'une simulation long et hétérogène :
  20 min par simulation en moyenne
  de 15 min à 35 min
- Développer de nouveaux algorithmes distribués :
  Prise en considération des temps hétérogènes
- Aucune connaissance sur ce problème (black-box) :
  Technique pour régler les paramètres des algorithmes

Exemples de résultats



Problème d'optimisation

Définition : problème d'optimisation

Un **problème d'optimisation** est un couple $(\mathcal{X}, f)$ avec :

- Espace de recherche : ensemble des solutions possibles,

$$\mathcal{X}$$

- fonction objectif : critère de qualité (ou de non-qualité)

$$f : \mathcal{X} \to \mathbb{R}$$

Résoudre un problème d'optimisation

Trouver la (ou les) meilleure solution selon le critère de qualité

$$x^\star = \mathrm{argmax}_{\mathcal{X}}\ f$$

(dans le cas de maximisation)

Contexte

Optimisation boite noire (Black box)

Nous ne pouvons connaitre que $\{(x_0, f(x_0)), (x_1, f(x_1)), \ldots\}$ donnés par un "oracle"
Aucune information sur la définition de la fonction objectif f n'est soit disponible ou soit nécessaire

$x \longrightarrow$  $\longrightarrow f(x)$

- Fonction objectif donnée par un calcul ou une simulation
- Fonction objectif peut être irrégulière, non différentiable, non continue, etc.

Typologie des problèmes d'optimisation

Classification

- **Optimisation combinatoire** : Espace de recherche dont les variables sont discrètes (cas NP-difficile)
- **Optimisation numérique (continue)** : Espace de recherche dont les variables sont continues
- **N'entrant pas dans les deux autres catégories** : combinaison discret/continue, programme, morphologie, topologie, etc.