


«Optimalité : choix, contraintes, hasard.» un thème où l'on retrouve la déraisonnable efficacité des mathématiques grâce à l'informatique.

 pixees.fr/optimalite-choix-contraintes-hasard-un-theme-ou-lon-retrouve-la-deraisonnable-efficacite-des-mathematiques-grace-a-linfo

20/04/2016

Trois mots clés qui se fédèrent en une théorie.



Choix. Pour faire un choix, pour prendre une décision, il faut clairement disposer d'informations et d'une représentation (ou "modèle") de la situation qui réclame ce choix. Que la proposition de décision soit confiée à l'humain-e ou à la machine, à moins de choisir au hasard, la situation va devoir être estimée, et un critère de choix devoir être arbitré. Une fois cette connaissance acquise on peut chercher parmi les solutions possibles, le meilleur ... choix. Le choix optimal. Tiens : voilà l'optimisation.

Contraintes. Pas si simple ? Souvent pas. Il y a des contraintes incontournables à respecter: lois physiques, souhaits de personnes. Il est parfois des problèmes qui ont tellement de contraintes qu'il n'y a pas de solution. Modéliser un problème concret, se ramène souvent à bien expliciter et formaliser les contraintes qui se posent dans les faits. Un fois cela fixé, on peut chercher la solution si elle existe, et si il y en a plusieurs, tant qu'à faire, choisir la meilleure. Revoilà l'optimisation.

Hasard. Tiens ? Serait-ce un hasard qu'on se hasarde à ajouter le hasard à ce tableau ? Certes pas. Mais attention : nous avons un double rendez-vous avec le [hasard](#).

Hasard: Phénomène imprédictible. Si je jette une pièce de monnaie qui retombe aléatoirement si pile ou face, pas d'erreur, ce n'est pas le fait du hasard. Le mouvement de ma main, les lois de la mécanique, la météo peut-être, tous ces éléments génèrent une trajectoire parfaitement déterministe : avec exactement les mêmes conditions initiales et les mêmes phénomènes, la pièce décrira la même trajectoire. Aucun troll ne se cache dans un monde parallèle pour perturber au hasard de son humeur le résultat. Et pourtant ... pourtant on arrive jamais à prédire numériquement le résultat, car la moindre toute petite erreur (et il y a toujours une dans le modèle ou au niveau des conditions initiales) va s'accumuler avec le temps de manière explosive (exponentielle, pour dire) et tout changer

très rapidement. Du coup le résultat est *imprédictible*. Ce n'est donc pas un hasard qu'on y voit du hasard.

Hasard: Mécanismes exploratoires. Me voilà dans le noir dans un labyrinthe sans aucun repère, à chercher à m'échapper à tâtons. À chaque embranchement dois-je plutôt tourner à droite ou à gauche ? Si je vais toujours à droite (ou toujours à gauche) il est clair que le moindre couloir circulaire va me faire tourner indéfiniment en rond. Et bien le hasard fait bien les choses : si je choisis au hasard souvent la gauche, mais de temps en temps la droite, un petit peu de mathématiques montre qu'avec une probabilité égale à un, je vais sortir de n'importe quel labyrinthe. Je confie au hasard ce que je ne sais choisir au mieux. En terme simple : je fais de l'exploration. Avec un ordinateur, un générateur pseudo-aléatoire (qui génère des choix imprédictibles) peut réaliser ce "tirage au sort". Et si je fais plusieurs tirages pour prendre le meilleur résultat, me voilà en train de faire de l'optimisation.

Comment en savoir plus sur ces éléments ?

Trois articles permettent d'illustrer et de rentrer dans ce sujet.





- **Et l'optimisation ?** *Mais si ! Le mieux est parfois l'ami du bien* est un document compagnon de celui-ci qui présente ce qu'est l'optimisation et en quoi c'est une idée majeure.

On redit la même chose avec des équations ?

Voici comment on peut faire un “modèle mathématique” de ces questions.

- Les choix possibles sont représentés comme le choix d'un élément, disons x , dans un ensemble de choix possibles, disons X . Il peut s'agir un nombre (réel), ou de plusieurs, disons n , nombres. Dans ce cas X est un ensemble de vecteurs de dimension n : $X \subset \mathbb{R}^n$.
- L'efficacité d'un choix se mesure par une fonction réelle (pour que le mot *meilleur* ait un sens simple : le plus grand ou le plus petit possible) $f(x)$, appelée *fonction objectif*.
- Mais il y a des contraintes, liant les coordonnées de x entre elles, ou interdisant certains (combinaisons de) valeurs. Elles peuvent être décrites comme des fonctions $g_i(x)$ qui doivent, suivant les cas
 - prendre une valeur donnée (on parle de contraintes égalité),
 - rester inférieures à une valeur donnée (on parle de contraintes inégalité).

Soit donc le problème suivant. Soit $f(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction objectif à minimiser. Soit $g(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ m fonctions exprimant des contraintes. Toutes ces fonctions sont supposées continuellement dérivables. Soit $a \in \mathbb{R}^m$ donné.

Problème : minimiser $f(x)$ sous la contrainte $g(x) = a$.

On propose l'approche suivante : on pose

$$X(a) = \{x \in \mathbb{R}^n \mid g(x) = a\},$$

supposé non vide pour tout a (qu'on va faire varier) puis on suppose que le minimum existe et on pose

$$V(a) = \min_{x \in X(a)} f(x).$$

On a une inégalité absolument triviale ... mais dont il prend parfois un peu de temps de se convaincre (on l'argumentera précisément) :

$$\forall x \in \mathbb{R}^n, \quad \mathcal{E}(x) := f(x) - V(g(x)) \geq 0.$$

En outre cette fonction $\mathcal{E}(x)$ est nulle en tout x qui résout le problème d'optimisation sous contrainte pour un certain a . Elle est donc minimum en ces points. **En supposant que la fonction $V(\cdot)$ est dérivable**, en déduire une condition nécessaire d'optimalité. (On pourra poser $\partial V / \partial a_i = -\lambda_i$)

On pourra essayer d'étendre cette approche aux contraintes inégalité, d'abord pour une unique contrainte $g(x) \leq a$ scalaire. (Étendre ensuite au cas de plusieurs contraintes est facile.) Il faut alors distinguer suivant que le minimum est atteint en un point x où $g(x) < a$ ou $g(x) = a$ (on dit alors que la contrainte est saturée). On s'intéressera au signe de la dérivée $V'(a)$.

La condition de dérivabilité de V est malheureuse, parce qu'on sait mal dire quand elle est satisfaite. La théorie de la convexité permet de s'en passer pour le cas des contraintes inégalité (voire des contraintes égalité si elles sont affines).

- **Programmation par contrainte ?** [La programmation par contrainte expliquée à ma garagiste ou à mon fleuriste](#) permet de comprendre ce qu'est une contrainte et explique comment formaliser des problèmes par ce moyen.
- **Estimation aléatoire ?** [Pourquoi ne pas confier au hasard ce qui est trop compliqué à estimer ?](#) propose une introduction à l'utilisation du hasard pour estimer des valeurs ... sûres.

Bien entendu, cette présentation ne peut-être exhaustive. Ainsi, on parle aussi de contraintes dans d'autres domaines des sciences de l'ingénieur : par exemple en résistance des matériaux, en robotique ([contraintes](#)

holonomes ou non-holonomes), ou au niveau des bases de données informatiques (contrainte d'intégrité).

Et cela ne se limite pas aux applications en ingénierie. L'optimisation est beaucoup utilisée en économie aussi, et dans d'autres domaines des sciences humaines et sociales, par exemple l'aide à la décision juridique grâce aux sciences du numérique.

N'hésitons pas à compléter ces idées en commentant cet article en bas de page !

Mais alors : quelles idées de sujets dans un univers aussi vaste ?

Le choix ne manque pas. Voici une sélection pour vous aider à ... choisir parmi des sujets encore plus originaux :).

Des algorithmes d'optimisation

Il faut bien comprendre que dans la plupart des problèmes d'ingénierie (ou d'économie), étant donné un choix de décision(s) (il peut y avoir plusieurs variables à choisir) proposé, on sait calculer la valeur du critère d'optimisation et des fonctions exprimant les contraintes à ne pas violer. Mais dans cette phrase, "on sait calculer" veut dire que si on donne des valeurs numériques caractérisant ce choix, un programme plus ou moins compliqué donne la valeur des fonctions évoquées. Il ne s'agit pas de formules plus ou moins explicites. De même pour le calcul des dérivées, à l'aide d'un autre programme plus compliqué (parfois déduit automatiquement du premier). Aucun espoir d'en déduire d'autres formules qui résoudraient des conditions d'optimalité, comme une dérivée nulle par exemple. On a donc recours à des *algorithmes d'optimisation* dont la nature dépend fortement du type de problème à résoudre. Pour des problèmes en variables continues (par opposition à des problèmes en nombres entiers, ou combinatoires, ...) on pourra par exemple s'intéresser à l'algorithme de gradient à pas fixe ou à pas optimal, l'algorithme de recherche unidimensionnelle par la "section dorée", l'algorithme d'Uzawa, (et décomposition d'un grand problème en plusieurs petits problèmes), l'algorithme de "recuit simulé" (intervention du hasard dans un algorithme), les algorithmes génétiques, etc. Cette liste à la Prévert donne des pistes ; il sera intéressant de programmer un algorithme et d'en examiner le comportement sur des problèmes types.

À chacun-e sa chacun-e

Le "problème des mariages" est un exemple type de problème combinatoire, dont la forme plaisante donnée ici recouvre des problèmes bien réels en informatique, en gestion de la production, etc.

Le problème consiste à former, dans un pool d'hommes et de femmes, des couples homme-femme correspondant au mieux aux ordres de préférence de chacun(e) parmi les partenaires possibles. On pourra examiner l'algorithme de Gale Shapley qui résout ce problème de façon élégante, puis à ses généralisations.

Que faire manger aux vaches ? Mathématiquement ? C'est un jeu !

Un fabricant d'aliments pour bétails (que l'un d'entre nous a vraiment visité) avait au catalogue divers aliments pour lesquels il garantissait la teneur en divers produits : protéines, différents types de lipides, de glucides, fibres. . . , une dizaine de constituants, entre deux bornes. Toutes les semaines, il allait sur le port du Havre et relevait les prix à la tonne des matières premières dont il se servait pour composer ses aliments : soja, tourteau de ricin, maïs . . . , une vingtaine de produits. Il expédiait par telex (c'était avant le fax (c'était avant Internet)) ces prix au siège de sa société (aux Pays Bas), qui par retour lui disait quel mélange faire pour chacun de ses aliments. Et cette prescription était réputée satisfaire aux garanties en minimisant le coût des matières premières utilisées. Comment faisaient-ils ?

C'est typiquement un problème de "[programmation linéaire](#)", d'abord étudié pendant la deuxième guerre mondiale (on dit que c'était le même problème, mais pour la nourriture des GI) par George Dantzig, auteur du premier algorithme dédié à ce problème : [l'algorithme du simplex](#).

Depuis, des algorithmes plus performants ont été inventés. C'est sûrement la forme la plus répandue d'optimisation numérique utilisée dans l'industrie. On sait résoudre des problèmes avec des milliers, voire des dizaines de milliers, de variables et de contraintes.

Derrière ces algorithmes figure une très belle théorie mathématique de la dualité, et, de façon inattendue, un lien étroit avec la théorie des jeux à deux joueurs et somme nulle.

Votre défi sera de trouver d'autres problèmes auxquels cette famille d'algorithme s'applique.

Des applications rigolotes de la programmation par contraintes.

La programmation par contrainte s'est appliquée à beaucoup de problèmes industriels, mais il a aussi des applications rigolotes :

- à l'oenologie, pour faire des combinaisons de cépages avec de bonnes propriétés ([travail de l'université de Montpellier](#)) ; il y a le même genre de problèmes pour les mélanges de pétroles, d'ailleurs...
- à la musique, comme outil d'assistance à la composition au niveau symbolique de la musique (aka la partition, sans le son) : il y a pas mal de travaux, que ce soit en musique classique ([problème d'harmonisation](#)) ou en musique contemporaine ([problèmes de rythme](#))
- à l'urbanisme : pour le développement de villes durables, comme dans le [projet Sustain](#). Là on est vraiment dans l'aide à la décision, c'est l'humain qui commande ([voir le livre de Belin-Christie-Truchet](#))
- à la gestion des catastrophes naturelles, pour [planifier les secours](#) ou router les secours dans un contexte hasardeux.

Au niveau des algorithmes, la communauté scientifique a produit des algorithmes qui sont tous référencés dans un [gros catalogue accessible en ligne](#) ce qui en soi est remarquable, car cela permet une réutilisation maximale de ces codes libres et ouverts.

Un détour par le monde des automates. Les [automates](#) se confondent avec la notion de machine à traiter de l'information. Il en existe de différentes natures et ceux que l'on se propose de regarder ici sont des automates probabilistes (on parle aussi de [modèle de Markov caché](#)). On les utilise dans le traitement de la langue naturelle, en bio-informatique, en neuroscience computationnelle, etc.

Ils sont très faciles à décrire. Un système (par exemple l'[alphabet thaï](#)) a un ensemble d'état (chacun des signes de ce système d'écriture). Si on me dicte un texte thaï que je tape sur mon clavier, peut-être qu'après le signe « [๒](#) » (de l'éléphant), je vais entrer un autre signe (peut-être « [๓](#) » (la souris)) ou pas. Je passe d'un état à l'autre par une transition, et comme je ne comprends strictement rien au thaï, tout ce que je vois c'est que d'un état (un signe) le "hasard" me dit quelle transition faire (et donc quel autre signe écrire) ou m'arrêter. Me voilà sur un graphe dont les noeuds sont les signes et des poids sur les transitions me disent qu'elle probabilité j'ai d'écrire un signe après un autre. En me promenant sur ce graphe, je peux calculer la distribution de probabilités sur l'ensemble de tous les mots possibles pour un alphabet donné.

Ce qui est remarquable, c'est que même sans comprendre quoi que ce soit ces informations peuvent donner des informations remarquables sur le texte en deça de son sens. On peut ainsi [reconnaitre un écrivain à sa ponctuation](#) sans regarder ...aucune lettre du texte. Des mécanismes de [traduction automatique](#) fonctionnent sur des mécanismes similaires (mais pas Google translate).

Voici quelques problématiques à aborder sur un tel sujet.

Prenons nos distances. On peut avoir envie de mesurer une distance entre deux graphes ou automates, comme

étant la distance entre les deux distributions de probabilité. L'idée peut être par exemple de réussir à faire une compression, c'est-à-dire remplacer un automate à 100000 états par un automate à 100 états, proche.

Formaliser cette notion de distance, la calculer de manière efficace en effectuant des tirages aléatoires (car le calcul exhaustif est impraticable) sont de jolis sujets.

Qui est le chef ? Quand j'ai un tel automate, je peux avoir envie d'avoir un représentant de la distribution. C'est peut être le mot qui a le chemin le plus probable, le mot le plus probable, le mot qui correspond à la médiane de la distribution.

Fermez les routes vous rentrerez plus vite à la maison.

Dans une ville encombrée, pour fluidifier la circulation, on peut penser ouvrir de nouvelles voies, mais il est arrivé (paradoxe de Braess) que cela fasse tellement empirer la situation de tout le monde qu'il faille parfois "bloquer des voies" pour ... limiter les embouteillages. C'est une expérience très facile à mettre en place et qui permet de « visualiser » le résultat théorique : avec des jetons et un dessin sur une grande feuille. Ces situations sont intéressantes parce qu'elles sont faciles à analyser et aussi à visualiser et qu'elles montrent la difficulté de diviser un problème d'optimisation (la minimisation globale d'un temps de trajet, c'est-à-dire la minimisation des bouchons) en plusieurs problèmes d'optimisation interdépendants (la minimisation du temps de trajet par chacun des conducteurs). Cela montre aussi comment l'optimisation "chacun pour soi" diffère de "l'optimisation collective". C'est donc un problème de théorie des jeux non coopératifs, où chacun "joue" pour soi.

Ce qui est intéressant, c'est qu'en ce qui concerne les points d'équilibre, les maths derrière sont bien accessibles à des élèves de prépas (les multiplicateurs de multiplicateurs de Lagrange ne sont pas au programme les outils nécessaires pour les comprendre sont là). On peut aller plus loin avec des bornes sur les dégradations possibles en fonction de la concavité de la courbe de latence des liens.

En ce qui concerne les dynamiques d'apprentissage des meilleures routes par les utilisateurs (qui convergent vers les points d'équilibres), ça peut être intéressant à implanter sur un ordinateur.

Les applications de ce type de problème sont nombreuses. Si le problème original de Braess concernait le trafic autoroutier, aujourd'hui c'est un problème qui concerne beaucoup les télécommunications : on a des paquets qui circulent dans un réseau et des problèmes de latence peuvent être critiques, par exemple dans les applications temps-réels comme la vidéo ou un paquet en retard est un paquet inutile.

Introduire des décisions aléatoires.

Regardons ce problème avec l'angle du hasard. Sans hasard chaque utilisateur va donc se « figer » sur une action telle que s'il change, sa situation ne sera pas meilleure (i.e. ici il ne mettra pas moins de temps sur son parcours). Cependant il y a pleins d'autres situations d'interaction qui existent et dans lesquelles la meilleure stratégie pour chaque individu n'est pas de prendre une action unique, mais au contraire de prendre un choix aléatoire parmi un ensemble. Intuitivement, c'est difficile à avaler que notre meilleur choix c'est de tirer une action et pourtant on trouve cela dans plein de situations. C'est par exemple évidemment le cas dans les systèmes biologiques (il y a pleins d'exemples que l'on peut donner au niveau des populations de bactéries notamment), mais aussi dans le cas des réseaux de télécommunications (par exemple pour éviter les collisions entre paquets on utilise des algorithmes qui envoient les paquets sur le système à des temps aléatoires).

Un exemple facile pour comprendre est celui de jeux de type pierre-feuille-ciseaux. Dans ce type de problème, la stratégie optimale est de jouer au hasard avec une probabilité $1/3$ $1/3$ $1/3$ chacune des stratégies. (Si on ne fait pas cela, alors l'adversaire peut s'adapter et jouer en conséquence des stratégies qui nous font perdre). D'ailleurs, en pratique, l'humain sait très mal appréhender le hasard et ce que l'on observe, c'est que les gens ont tendance à jouer selon des séquences assez prévisibles. Du coup, il y a des concours de jeux pierre-feuille-ciseaux

(si, si) et des sites web qui vous expliquent comment gagner au jeu. Il y a aussi des programmes informatiques qui sont fait pour jouer contre des humains et « apprendre » leur comportement pour nous battre. Or il se trouve que dans le cas de surentrainement sur un jeu particulier, les humains peuvent apprendre à réellement jouer optimalement. Il a été montré par exemple que les joueurs de football professionnels savent tirer les pénalités de façon optimale de sorte que l'on ne peut pas prévoir s'ils vont viser à gauche ou à droite. (C'est sympa parce que les statistiques de tous les matchs de la FIFA sont disponibles à tous — du coup on a bien pu faire les études statistiques dessus).

Il y a même un mécanisme encore plus rigolo à propos du choix et du hasard.

Il se trouve qu'il existe des situations dans lesquelles le choix optimal est fondamentalement différent si il n'y a pas de hasard, que si il y en a un, même si le hasard est arbitrairement faible. Un exemple typique est le problème des deux généraux ou problème de l'information commune. C'est un problème qui est connu à la fois [en économie](#) et [en informatique](#). C'est intéressant parce qu'il s'agit d'une situation dans laquelle si l'échange d'information entre les 2 personnes est parfaite le résultat est complètement différent que si il y a du hasard dans la transmission de l'information (ce qui se traduit par la capture du message ou la perte du paquet), et ce indépendamment de la probabilité de perte. C'est un problème qui a plein d'applications.

La théorie de l'Évolution, source d'optimalité par le hasard.

On admet que l'évolution des espèces biologiques a sélectionné les comportements les plus efficaces. Pourquoi, donc, observe-t-on des comportements différents au sein d'une même espèce ?

Un peu de dynamique des populations (comment évoluent les effectifs des populations) permet de répondre. L'exemple type est celui dit "des faucons et des colombes" par référence non pas à deux espèces animales (on parle bien de variabilité intra-spécifique : au sein d'une même espèce) mais à la terminologie désignant au congrès





des USA les députés belliqueux ou pacifistes. «Être agressif ou ne pas l'être» ? that is the question, here La coexistence de plusieurs comportements au sein d'une même espèce vivante est prédite C'est un phénomène de théorie des jeux. L'équilibre de l'évolution est atteint quand plusieurs comportements donnent le même résultat, et, c'est ce qui est original par rapport à une simple non-unicité de l'optimum, quand la proportion des individus qui adoptent chaque comportement est exactement celle qui permet l'équilibre. C'est exactement la même chose que la multiplicité des trajets utilisés par les automobilistes qui tous prennent le chemin le plus rapide. (Équilibre de Wardrop) C'est très voisin du phénomène des stratégies aléatoires (on dit "mixtes", mauvaise traduction de "mixed", mélangées) en théorie des jeux.

Plus généralement, la théorie de l'évolution introduit aussi une source d'optimalité par le hasard. La reproduction de notre ADN via les ribosomes et l'ARN messenger produit des erreurs de recopie. Ces erreurs donnent naissance à des individus "mutés". La plupart d'entre eux sont non ou peu viables. Mais si une mutation aléatoire donne naissance à un animal mieux adapté (disons avec une meilleure efficacité reproductive) il peut être à l'origine d'un nouveau groupe d'animaux (une nouvelle espèce) qui va petit à petit envahir toute la "niche écologique". La reproduction et ses mutations aléatoires se comporte donc comme un algorithme de recherche aléatoire de la meilleure efficacité reproductive.

Charlotte Truchet, Colin de la Higuera, Corinne Touati, avec l'aide de **Pierre Bernhard** et le travail d'édition de **Valérie François** et **Martine Courbin-Coulaud**.