Recherche Opérationnelle

Chapitre 1 : Introduction

Julian Tugaut

Vendredi 07 *Octobre* 2016

Sommaire

Plan

La Recherche Opérationnelle dans l'antiquité

L'appellation « Recherche Opérationnelle » date des années quarante. Néanmoins, les problèmes qui lui sont relatifs sont bien plus anciens. Ainsi, les Carthaginois ont bâti leur ville en arc de cercle autour de sa citadelle. Cela implique que les bâtisseurs connaissaient déjà, au 9$^{\text{ème}}$ siècle avant Jésus Christ, la figure plane qui, à périmètre donné, présente la surface maximale.

Quelques noms

- Léonard Euler, qui a résolu le problème des ponts de Königsberg (à l'origine de la théorie des graphes).
- Gaspard Monge, qui a étudié le problème de transports (pan entier des probabilités et d'analyse des Équations aux Dérivées Partielles).
- Augustin Cournot, qui a étudié l'économétrie.
- André Ampère, qui a étudié la théorie des jeux.
- Emile Borel, qui a étudié la théorie des jeux.

Naissance de la Recherche Opérationnelle - 1

La véritable naissance de la Recherche Opérationnelle a eu lieu pendant la seconde guerre mondiale. Elle portait l'appellation « Operationnal research » (en Anglais) et « Operation research » (en Américain).

Le mot « operation » est à comprendre dans le sens d'activité militaire. Son application est donc avant tout militaire.

Lors de la naissance de cette science, l'objectif affiché était la préparation scientifique de décisions militaires : problèmes d'organisation, de planifications, de transport, de ravitaillements, implantations de radars, protection de convois maritimes contre les sous-marins...

Naissance de la Recherche Opérationnelle - 2

Patrick Maynard Stuart Blackett est le fondateur de la recherche opérationnelle. En effet, sur la demande du gouvernement britannique, Blackett a réuni autour de lui une équipe hétérogène comprenant des mathématiciens (dont des statisticiens), des physiciens, des biologistes, des économistes...

Le but était d'avoir les points de vue les plus différents sur les questions étudiées. Parallèlement, l'exigence était faite d'avoir accès à toutes les sources d'information pour pouvoir produire des rapports aussi objectifs que possibles. Enfin, Blackett insista sur le fait que les documents ne constituaient jamais qu'une opinion raisonnable sur les décisions à prendre, le dernier mot revenant toujours aux responsables politiques.

Puis, les Américains ont emboîté le pas.

L'après-guerre - 1

Après la guerre, la démarche et les méthodes mises au point furent de plus en plus utilisées dans le cadre industriel : coordination du travail pour en accroître la productivité et la rentabilité (gestion, politique d'investissements, répartition de tâches...).

Ces applications furent bien sûr grandement facilitées par la commercialisation des premiers ordinateurs dans les années cinquante.

L'après-guerre - 2

- Dans quel ordre doit-on effectuer les opérations de montage d'une voiture pour en minimiser le coût de production ?
- Comment découper des pièces géométriques dans une tôle métallique pour en limiter les pertes ?
- Déterminer les contenances minimales des entrepôts pour éviter les ruptures de stocks ?
- Comment calculer les meilleurs itinéraires (temps, coût, distance, etc) de livraison de marchandises ?

Il s'agit ainsi de problèmes d'ordonnancement de tâches, d'optimisation de production, de gestion des stocks, de détermination d'itinéraires, de répartitions de tâches, de production optimale (bénéfice maximal avec des ressources limitées, moindres coûts tout en satisfaisant la demande...).

Plan

Une définition ? - 1

Commençons par donner la définition qui apparaît sur le dictionnaire de l'informatique du Larousse en 1996 : « Science de la préparation des décisions qui procède de la mathématisation des facteurs essentiels entrant en jeu dans les problèmes d'organisation militaire, économique, industrielle, afin de clarifier les données motivant une décision ».

Une définition ? - 2

Donnons maintenant la définition de Wikipedia (du 28 septembre 2015) : « La recherche opérationnelle peut être définie comme l'ensemble des méthodes et techniques rationnelles orientées vers la recherche de la meilleure façon d'opérer des choix en vue d'aboutir au résultat visé ou au meilleur résultat possible. Elle fait partie des aides à la décision dans la mesure où elle propose des modèles conceptuels en vue d'analyser et de maîtriser des situations complexes pour permettre aux décideurs de comprendre et d'évaluer les enjeux et d'arbitrer ou de faire les choix les plus efficaces. Ce domaine fait largement appel au raisonnement mathématique (logique, probabilités, analyse des données) et à la modélisation des processus. Il est fortement lié à l'ingénierie des systèmes, ainsi qu'au management du système d'information. ».

Une définition ? - 3

En fait, il n'y a pas de définition réellement satisfaisante et exhaustive de la Recherche Opérationnelle car il n'y a pas vraiment de domaine spécifique ni de méthode unique de résolution mais de nombreux champs d'application très divers.

L'appellation d'origine est militaire mais elle reste l'appellation standard. On en trouve parfois d'autres :

- Optimisation discrète. (par opposition au cours d'optimisation continue).
- Optimisation combinatoire.
- Mathématiques de la décision.
- Programmation mathématique.
- Programmation discrète.

Une définition ? - 4

Ces appellations ne sont pas dénuées de sens. Néanmoins, elles ne rendent pas entièrement compte de la diversité des domaines étudiés ni de celles des méthodes employées.

Remarque

Plutôt que d'en rechercher une définition précise, ce qui s'avèrerait par ailleurs fort inutile, donnons quelques outils mathématiques classiques de modélisation à voir comme des cadres dans lesquels peuvent se formuler les problèmes et qui permettent d'en proposer des éclairages, quelques domaines usuels d'étude, ainsi que des méthodes classiques de résolution.

Problèmes et Modélisation

- Programmation linéaire classique ou programmation linéaire en nombres entiers.
- Graphes (plus courts chemins, flots, arbres, colorations, recouvrements...).
- Hypergraphes (généralisation naturelle des graphes).
- Ensembles ordonnés (ordre total, ordre partiel, extension linéaire...).
- Modèles probabilistes (chaînes de Markov, phénomènes aléatoires, files d'attente...).
- Théorie des jeux.

Méthodes de résolution

- Géométrie ou algorithme du simplexe en programmation linéaire.
- Algorithmes sur les graphes.
- Programmation dynamique.
- Méthodes arborescentes (BAB[1] ou SEP[2]).
- Méthodes approchées (heuristiques, métaheuristiques).

1. Branch And Bound
2. Séparation et Évaluation Progressive

Remarque

Un même problème peut parfois être modélisé de plusieurs façons différentes et donc être résolu par plusieurs méthodes. Dans ce cas de figure, il conviendra de trouver la modélisation en rapport avec les attendus du problème et conduisant à la résolution la plus efficace et la plus adaptée.

Plan

But

Notre propos est ici de présenter (et non de résoudre) quelques problèmes très classiques et qu'il faut connaître de Recherche Opérationnelle. Le but est tout simplement d'essayer d'en faire comprendre la diversité, les types de questions abordées, les aspects très concrets des problèmes, les mathématiques sous-jacentes...

Dilemme du brasseur - 1

Un brasseur de bière dispose de 240 kg de maïs, de 5 kg de houblon et de 595 kg de malt.

Il peut produire aussi bien de la bière blonde que de la bière brune.

Un tonneau de bière blonde nécessite 2.5 kg de maïs, 125g de houblon et 17.5 kg de malt. Chaque tonneau produit un bénéfice net de 65 euros.

Un tonneau de bière brune nécessite 7.5 kg de maïs, 125g de houblon et 10 kg de malt. Chaque tonneau produit un bénéfice net de 115 euros.

Dilemme du brasseur - 2

On suppose que les autres ingrédients (comme l'eau) sont disponibles en quantité illimitée : on ne tiendra donc pas compte de leurs coûts.

Quel est, pour le brasseur, le meilleur plan de production ? En d'autres termes, quel est le plan de production lui fournissant le bénéfice maximum avec les stocks disponibles ?

Dilemme du brasseur - 3

Mathématiquement, il s'agit de maximiser la fonction

$$B(x, y) := 65x + 115y$$

sous les contraintes (inégalités) :

$$\begin{aligned}
&2.5x + 7.5y \leq 240\\
&0.125x + 0.125y \leq 5\\
&17.5x + 10y \leq 595\\
&0 \leq x\\
&0 \leq y\,.
\end{aligned}$$

Dilemme du brasseur - 4

C'est un problème classique de Programmation Linéaire car il s'agit d'optimiser une fonction linéaire sous des contraintes linéaires.

Notons que ce problème peut se résoudre de deux manières différentes :

1. En utilisant une interprétation géométrique (ce qui est possible car on a deux variables).
2. En utilisant l'algorithme du simplexe (recommandé dès lors que l'on a trois variables ou plus).

Tous les problèmes de programmation linéaire ne sont bien sûr pas aussi simples car en général, ils contiennent de très nombreuses variables.

Dilemme du brasseur - 5

Ce sujet est un grand classique de la Recherche Opérationnelle et est étudié depuis les années quarante. De très nombreuses méthodes, ainsi que des logiciels de résolution plus ou moins efficaces suivant la nature des variables sont connues comme le fameux algorithme du simplexe, dû à Dantzig.

Ordonnancement de tâches - 1

Tout projet quelque peu conséquent se décompose en général en l'exécution d'un certain nombre de tâches élémentaires différentes $x_1, \cdots, x_n$ de durées respectives $d_1, \cdots, d_n$ et soumises à des contraintes de précédence du type « x_i doit être réalisée avant x_j », que l'on notera « $x_i < x_j$ ».

Ordonnancement de tâches - 2

Considérons par exemple huit tâches x_1, x_2, x_3, x_4, x_5, x_6, x_7 et x_8 soumises aux contraintes de précédence :

$$\begin{aligned}
x_1 &< x_6 \\
x_2 &< x_6 \\
x_3 &< x_5, x_6 \\
x_4 &< x_2 \\
x_5 &< x_1, x_2 \\
x_7 &< x_1, x_2, x_4 \\
x_8 &< x_4, x_5, x_7 .
\end{aligned}$$

On se donne les durées suivantes : $d_1 = 6$, $d_2 = 1$, $d_3 = 1$, $d_4 = 3$, $d_5 = 5$, $d_6 = 1$, $d_7 = 2$ et $d_8 = 3$.

Ordonnancement de tâches - 3

La question fondamentale est bien sûr : Comment réaliser le projet ? En d'autres termes, après avoir vérifié la cohérence des tâches et détecté d'éventuelles contraintes inutiles, il s'agit de déterminer un ordre d'exécution possible des tâches, les dates de réalisation des tâches et la durée minimale globale du projet.

La réponse à ces questions passe par une modélisation du problème (quelle structure mathématique peut représenter la situation ?) puis sa résolution (comment, à partir de cette structure, répondre aux questions posées ?).

Plusieurs modélisations sont ici possibles : Graphe de précédence des tâches, Graphe PERT, Diagramme de Gantt.

Ordonnancement de tâches - 4

Le domaine concerné par les deux premières modélisations est la théorie des graphes.

Définition

Un graphe est une relation binaire a priori anti-réflexive (pas de boucle) sur un ensemble a priori fini.

Définition

Si la relation est quelconque, on parle de graphe orienté. On le note alors $G = (X, U)$ où X est appelé ensemble des sommets et U ensemble des arcs.

Définition

Si la relation est symétrique, on parle de graphe non orienté. On le note alors $G = (X, E)$ où E est l'ensemble des arêtes.

Ordonnancement de tâches - 5

Dans notre cas, le graphe de prédécence, $G = (X, U, v)$ est un graphe simple orienté valué défini par $X = \{\text{tâches}\} \cup \{f\}$, $U = x_i x_j \ : \ x_i < x_j$ et v est une valuation, ou pondération, des arcs correspondant à leurs durées. Le sommet f est un sommet supplémentaire rendant compte de la fin du projet, il peut donc succéder à toutes les tâches.

Les situations pouvant se modéliser sous la forme d'un graphe sont innombrables : réseaux de transport, réseaux informatiques, réseaux électroniques, réseaux de connaissance, éléments finis, automates finis, problèmes d'emploi du temps et d'ordonnancements, Internet...

Problème du Voyageur de Commerce - 1

Ce problème s'énonce traditionnellement de la façon suivante : un Voyageur de commerce partant de la ville de son domicile doit visiter, une fois et une seule, chacun de ses clients et revenir chez lui. Quel est alors le meilleur itinéraire possible, c'est-à-dire un itinéraire minimisant le coût de son parcours ?

On peut aussi le présenter en considérant n points du plan, chacun connu par ses coordonnées, la question est alors de déterminer une « tournée » de longueur minimale passant une fois et une seule par chacun de ces points et revenant au point de départ. Le coût est bien sûr ici la distance euclidienne.

Problème du Voyageur de Commerce - 2

Les principales méthodes connues pour résoudre exactement ce type de problème sont des méthodes arborescentes, du type Séparation et Évaluation Progressive (SEP) ou Séparation et Évaluation Séquentielle (SES, BAB [Branch and Bound] en Anglais). Elles consistent à explorer de manière arborescente l'ensemble des solutions possibles avec à chaque étape l'évaluation d'une fonction adéquate permettant d'éviter intelligemment des explorations inutiles afin d'accélérer la recherche pour ne pas sombrer dans l'explosion combinatoire, c'est-à-dire l'examen d'un nombre exponentiel de cas.

Problème du Voyageur de Commerce - 3

À défaut de trouver le chemin le plus court, car trop coûteux en temps de calcul, on peut se contenter de construire rapidement (c'est-à-dire en un temps de calcul raisonnable) un itinéraire que l'on espère assez bon. Se pose alors la question de mettre au point de telles méthodes appelées algorithmes d'approximation ou heuristiques.

Problèmes de Partitions et de Chargements - 1

Le problème de la partition ou du partage équitable est le suivant : étant donné n tâches indépendantes dont les durées respectives sont les entiers $d_1, \cdots, d_n$, comment les répartir sur deux machines afin de terminer au plus vite ?

Il s'agit donc de partitionner l'ensemble des durées en deux sous-ensembles dont les sommes soient les plus proches possibles.

Problèmes de Partitions et de Chargements - 2

Le célèbre problème du sac à dos s'énonce comme suit.

Données

On dispose de n objets numérotés de 1 à n, d'utilités et de volumes respectifs $u_1, \cdots, u_n$ et $v_1, \cdots, v_n$. On a également un sac à dos de volume maximum V.

Question

Comment remplir au mieux le sac ? Autrement dit, quels objets doit-on prendre dans le sac pour en maximaliser l'utilité sans dépasser le volume V ?

Problèmes de Partitions et de Chargements - 3

Hypothèses

On suppose bien sûr que

- $0 < u_i$ et $0 < v_i \leq V$ pour tout $i \in \{1; \cdots; n\}$.
- $\sum_{i=1}^{n} v_i > V$.

Problèmes de Partitions et de Chargements - 4

On peut montrer que ces deux problèmes sont équivalents : si l'on sait résoudre l'un, on sait résoudre l'autre moyennant un travail négligeable. Néanmoins, ils sont aussi difficiles l'un que l'autre malgré leurs simplicités d'énoncés. Il n'existe pas, à l'heure actuelle de méthode satisfaisante de résolution exacte dans le sens où tous les algorithmes connus sont exponentiels dans le plus mauvais cas, c'est-à-dire peuvent être très coûteux en temps de calcul. On peut donc raisonnablement les qualifier de « difficiles à résoudre rapidement ».

Problèmes de Partitions et de Chargements - 5

Ces problèmes se résolvent comme le problème du voyageur de commerce à savoir par des méthodes arborescentes. Une autre technique, la programmation dynamique, permet aussi de résoudre très élégamment, et même rapidement si les valeurs de données ne sont pas trop grandes.

Plan

Un exemple introductif - 1

Face à un problème de Recherche Opérationnel, hormis la Modélisation et la Résolution, un des aspects fondamentaux est aussi de savoir évaluer l'efficacité de l'algorithme choisi. Cette notion d'efficacité est à comprendre non seulement dans le sens de sa justesse mais aussi de l'estimation des ressources nécessaires à son exécution, c'est-à-dire de son coût. Il nous faut donc définir et quantifier cette efficacité.

Connaître la difficulté d'un problème, c'est-à-dire l'efficacité de ses possibles algorithmes de résolution, question souvent difficile, ou bien ne serait-ce que d'avoir une idée de cette difficulté, est aussi un paramètre vital dans la pratique.

Un exemple introductif - 2

Pour mieux appréhender ces questions, examinons d'abord un problème très simple : le produit C de deux matrices A et B, chacune de taille $n \times n$.

Le principe d'un algorithme de résolution nous est donné par la définition mathématique des coefficients $C_{i,j}$ de la matrice C à calculer :

$$C_{i,j} := \sum_{k=1}^{n} A_{i,k} B_{k,j} .$$

Un exemple introductif - 3

Il vient l'algorithme suivant écrit en pseudo langage :

```
Pour i=1 à n faire
{
    Pour j=1 à n faire
    {
        C_{i,j} := A_{i,1} × B_{1,j}
        Pour k=2 à n faire
        {
            C_{i,j} := C_{i,j} + A_{i,k} × B_{k,j}
        }
    }
}
```

Un exemple introductif - 4

La preuve de cet algorithme, finitude et justesse, est immédiate puisqu'il est l'application directe sous forme de trois boucles imbriquées de la formule précédente.

Après cet argument de preuve, comment maintenant estimer l'efficacité de cet algorithme, c'est-à-dire quantifier les ressources nécessaires à son exécution ?

Pour mettre en œuvre cet algorithme, il nous faut d'une part une représentation des données et des résultats et, d'autre part, effectuer un certain nombre d'opérations sur ces représentations. Ces ressources correspondent donc à l'espace mémoire et au temps d'exécution sur une machine. Ce sont ces quantités qu'il nous faut estimer.

Un exemple introductif - 5

L'algorithme manipule trois matrices de taille $n \times n$ et un nombre constant de variables auxiliaires (i, j, k et n). L'espace mémoire est donc proportionnel à n^2.

D'autre part, il y a exactement $n^2(n-1)$ additions, n^3 multiplications, n^3 affectations et les trois boucles imbriquées nécessitent $n^2(n-1)$ étapes supplémentaires pour leurs mises en œuvre. Le temps d'exécution sera donc proportionnel à n^3. Remarquons aussi que l'algorithme aura exactement le même comportement quelle que soit la donnée.

Un exemple introductif - 6

Après avoir analysé un algorithme de résolution, intéressons-nous maintenant au problème lui-même : d'autres algorithmes sont-ils possibles ? Peut-on faire mieux, c'est-à-dire plus efficace ? Existe-t-il pour ce problème un algorithme optimal, c'est-à-dire le moins coûteux possible ?

En 1969, Strassen a publié un algorithme permettant de multiplier deux matrices de taille $n \times n$ en un temps proportionnel à $n^{\log_2(7)}$, soit environ $n^{2.81}$, donc asymptotiquement meilleur que l'algorithme usuel. Depuis, de meilleurs algorithmes ont été publiés. Le record est actuellement détenu par Coppersmith et Winograd avec un algorithme en $O\left(n^{2.376}\right)$ même si cet algorithme est peu utilisable en pratique.

Définitions - 1

L'étude du problème précédent nous a permis d'illustrer les notions et questions en jeu : tailles des structures manipulées, nombre d'opérations utilisées, existence de meilleurs algorithmes, bornes inférieures... Définissons maintenant plus précisément ces notions.

Il y a en fait deux critères pratiques principaux permettant de mesurer l'efficacité, ou le coût d'un algorithme A résolvant un problème P : le temps de calcul et l'espace mémoire nécessaire. Ils correspondent respectivement à la complexité en temps et à la complexité en espace de A.

Définitions - 2

Définition

La complexité en temps d'un algorithme A est le nombre d'opérations élémentaires nécessaires au déroulement de A.

Définition

La complexité en espace d'un algorithme A est le nombre d'objets élémentaires utilisés par le déroulement de A.

Définitions - 3

Il faut donc définir les opérations élémentaires, c'est-à-dire à temps constant, et les objets élémentaires (c'est-à-dire ceux qui sont de taille constante) autrement dit représentables par un nombre constant de cases mémoires. Cela correspond à la notion de modèle de machine.

Dans l'exemple précédent, le modèle de machine utilisé était le modèle uniforme : chaque opération (affectation, addition, multiplication, instructions de gestion de boucle) est supposée avoir un coût constant. De même, chaque nombre manipulé est supposé être représentable en taille constante.

Ces complexités s'expriment asymptotiquement en fonction de la taille du problème.

Définitions - 4

Définition

La taille du problème (aussi appelé taille de la donnée) est le nombre d'objets élémentaires nécessaires à sa représentation.

Il s'agit en gros du nombre de cases mémoires.

Dans l'exemple précédent, la taille du problème est celle des données, c'est-à-dire celle des deux matrices, soit $2n^2$, mais il est d'usage de prendre n comme paramètre de taille.

Définitions - 5

Le plus souvent, on s'intéresse à la complexité en temps dans le plus mauvais cas :

Définition

La complexité en temps dans le plus mauvais cas est le nombre d'opérations élémentaires exécutées par l'algorithme sur l'ensemble des données de taille n.

On peut aussi s'intéresser à la complexité en moyenne, c'est-à-dire au comportement moyen de l'algorithme en ayant au préalable défini une loi de probabilité sur l'ensemble des données de taille n.

Définitions - 6

Dans l'exemple de la multiplication des matrices, la complexité en temps est en $O(n^3)$ et la complexité en espace est en $O(n^2)$.

Voici quelques autres exemples :

- Recherche d'un élément précis dans une liste de n éléments : $O(n)$, possible en $O(\log(n))$.
- Recherche du plus grand élément d'un tableau de n entiers : $O(n)$, c'est optimal.
- Tri de n nombres : $O(n^2)$, possible en $O(n\log(n))$ qui est optimal.
- Fusion de deux tableaux triés de tailles n : $O(n)$.

Ces notions de complexités vont donc nous permettre de comparer des algorithmes, donc de comparer des méthodes de résolution d'un problème.

Définitions - 7

Pour montrer l'importance pratique de ces notions, considérons le temps d'exécution de trois programmes codant trois algorithmes A_1, A_2 et A_3 de complexités respectives égales à n, n^3 et 2^n pour des données de taille n sur une machine M effectuant 10^9 opérations par seconde :

n	10	50	100
$A_1(n)$	$10^{-8}s$	$5 \times 10^{-8}s$	$10^{-7}s$
$A_2(n^3)$	$10^{-6}s$	$2.25 \times 10^{-4}s$	$10^{-3}s$
$A_3(2^n)$	$10^{-6}s$	13 *jours*	4×10^{13} années

Définitions - 8

Les algorithmes A_1 et A_2 de complexités polynomiales sont parfaitement utilisables même si A_1 est nettement meilleur. Mais, A_3, de complexité exponentielle, est inutilisable même pour des tailles très raisonnables.

De plus, toute évolution technologique ne fait que confirmer cette différence : calculer les améliorations en termes de taille maximale de données traitables en un certain temps sur une machine mille fois plus rapide.

Définitions - 9

Nous avons vu qu'en ayant un algorithme de résolution d'un problème, deux questions naturelles se posent :

- Peut-on faire mieux ?
- Comment déterminer si tel algorithme est le meilleur possible ? En d'autres termes, comment montrer que l'algorithme a une complexité inférieure à tout autre algorithme de résolution ?

Répondre à la seconde question, c'est déterminer la complexité du problème. C'est une tâche en général très difficile et peu de résultats sont connus dans ce domaine.

Cette problématique constitue la théorie de la complexité des problèmes. Très sommairement, cette théorie met en évidence plusieurs classes de problèmes :

- La classe **P** : problèmes polynomiaux, c'est-à-dire des problèmes admettant un algorithme de complexité polynomiale.
- La classe **NP** : problèmes des décisions (à réponse « oui » ou « non ») à preuve polynomiale, c'est-à-dire pour lesquels on peut vérifier en un temps polynomial qu'une solution donnée ou proposée est à réponse « oui ».
- Les problèmes **NP - Complets** : classe d'équivalence des problèmes les plus difficiles de **NP** dans le sens où toute méthode de résolution de l'un d'eux permettrait de résoudre efficacement tout problème de **NP**.
- Les problèmes **NP - difficiles** : problèmes au moins aussi difficiles que les problèmes **NP - Complets**.

Plan

Plan et buts du cours - 1

Le cours se divise en trois parties. On commencera par étudier la programmation linéaire (la méthode géométrique et la méthode du simplexe). Puis, on parlera de programmation dynamique. Enfin, on étudiera les graphes.

Il ne s'agit pas dans ce cours de faire une présentation très fouillée de la Recherche Opérationnelle mais de sensibiliser, au travers de quelques domaines et problèmes classiques ainsi que des méthodes générales de résolution, à la nature particulière, originale, très diversifiée et en perpétuelle évolution de cette matière qui est quasiment un passage obligé dans l'acquisition d'une culture scientifique générale que doit acquérir tout ingénieur.

Plan et buts du cours - 2

Le but n'est pas de devenir un spécialiste de la Recherche Opérationnelle mais d'en bien connaître la problématique générale (le type de problèmes traités, les méthodes utilisées, ce que l'on peut en attendre, les limitations actuelles) et de prendre conscience que c'est un outil qui a une part importante dans toute prise de décision quant à la réalisation d'un projet substantiel.