Cours de Recherche Opérationnelle (RO)

Dr. Cheikh GUEYE
Université Cheikh Anta Diop de Dakar (UCAD, Sénégal)
Université Claude Bernard Lyon 1 (UCBL, France)
Titulaire d'un Doctorat en Mathématiques appliquées (UCBL-Lyon)
Titulaire d'un Doctorat en Analyse, Statistiques et Applications (UCAD-Dakar)
Laboratoire de Mathématiques et Applications (LMA-UCAD)
cheikh.gueye1990@yahoo.fr, *cheikh39.gueye@ucad.edu.sn*

Institut Supérieur d'Informatique **Dép:** Génie Info

Niveau: Licence 2 Génie Logiciel **Année:** 2025 - 2026

20 décembre 2025

Plan

Informations générales

1. **Intitulé du cours :** Recherche Opérationnelle (RO)
2. **Niveau :** Licence/Master (selon approfondissement)
3. **Volume horaire :** 20h Cours + 10h TD/TP
4. **Pré requis :**
    - Mathématiques (algèbre linéaire, logique, probabilités de base)
    - Algorithmique et structures de données
    - Programmation (Python/R/Java/C++ ou langage équivalent)
    - Tableur Excel.
    - Logiciel Géogébra.
5. OS1 : Déterminer le modèle mathématique associé à un problème de Programmation Linéaire (PL).
6. OS2 : Résoudre analytiquement un problème de PL.
7. OS3 : Résoudre graphiquement un problème de PL.
8. OS4 : Simuler un problème de PL avec le Solveur d'Excel et Python.

Objectifs du cours

Ce cours est une initiation à la modélisation et à la résolution dans le cadre de la programmation linéaire. L'objectif poursuivi est de fournir des éléments aux étudiants pour :

1. Comprendre les fondements de la Recherche Opérationnelle et leurs applications au développement logiciel.
2. Apprendre à construire des modèles mathématiques.
3. Modéliser des prob de gestion, d'optimisation et d'ingénierie logicielle.
4. Connaître le fonctionnement de l'algorithme du simplexe.
5. Utiliser des méthodes quantitatives et algo pour la prise de décision.
6. Développer des applications logicielles intégrant des modèles RO (planification, affectation, optimisation des ressources).
7. Interpréter les principaux résultats d'une analyse de sensibilité et en connaître les limites.
8. Maîtriser un logiciel d'optimisation.

Introduction à la recherche opérationnelle

- La recherche opérationnelle peut être définie comme l'ensemble des méthodes et techniques rationnelles orientées vers la recherche du meilleur choix dans la façon d'opérer en vue d'aboutir au résultat visé ou au meilleur résultat possible.
- Traite des problèmes d'optimisation :
  - **Maximisation** d'un profit, d'une performance, d'un rendement,...
  - **Minimisation** d'un coût, d'une dépense, ...
- La démarche méthodologique utilisée pour résoudre un problème de décision à l'aide de la RO :
  - Identification des variables de décision ;
  - décrire les contraintes ;
  - et établir la fonctions objectif ou la fonction économique.
  - Passage du problème réel (ou un problème posé en entreprise) au modèle mathématique (programmation linéaire ou modèle linéaire).

Types de problèmes traités

Lorsqu'on est confronté à un problème

- Qui comprend un nombre fini (souvent très grand) de solutions admissibles parmi lesquelles on cherche la solution optimale ou proche de l'optimum : **Problème combinatoire**
- Qui consiste à trouver une solution optimale face à un problème dont les termes dépendent de l'interrelation entre entre ses propres agissements et ceux d'autres décideurs : **Problème concurrentiel**.
- Qui se pose en terme incertain : **Problème aléatoire**

Origines

- Pendant la Seconde Guerre mondiale : Pour fournir des techniques d'optimisation des ressources militaires.

Le premier sucés obtenu en 1940 par le Prix Nobel de physique **Patrick Blackett** qui résolut un problème d'implantation optimale de radars de surveillance.

- Le qualificatif « opérationnelle » vient du fait que les premières applications de cette discipline avait trait aux opérations militaires.

Expansion

- Après la seconde guerre mondiale (1950), la recherche opérationnelle fait son entrée dans les entreprises.
- Pendant les années 50 et 60, l'intérêt et le développement de la Recherche Opérationnelle a agrandi dans la suite de l'application sur le commerce et l'industrie. Prenons l'exemple du problème de calcul du projet optimal de transport de sable de la construction aux travaux d'édification à Moscou, où il y avait 10 points d'origine et 230 destinations. Pour la résolution ils ont fait usage d'un ordinateur Strena dans le mois de juin de 1958, et après 10 jours de travail, il a fourni une réduction du 11 % de coûts par rapport aux coûts originaux prévus.

Grâce à l'informatique, elle a pris un essor remarquable dans presque tous les domaines.

- Développement de la politique national d'administration de l'eau, y compris les nouvelles installations, les procédés d'opérations et les coûts (Ministère hollandais de l'infrastructure et de l'environnement (The Netherlands Rijkswaterstaat), 1985).
- Distribution optimal des ressources hydriques et thermiques dans le système national de génération d'énergie (Electrobras/CEPAL Brasil, 1986).
- Restructuration total de la chaîne d'approvisionnement entre les fournisseurs, les usines, les centres de distribution, les endroits potentiels et les zones de commerce (Digital Équipement Corp., 1995).
- Optimisation du mélange d'ingrédients disponibles pour que les combustibles obtenus atteindraient aux demandes de vente et de qualité (Texaco, Inc., 1989).
- etc.

Définition du problème

Objectif : Comprendre clairement la situation à étudier.

1. Identifier le contexte et les objectifs (maximiser, minimiser, équilibrer. . .).
2. Déterminer les ressources disponibles (moyens, contraintes, données).
3. Identifier les décideurs, les acteurs et les variables du problème.

Exemple : Une entreprise veut minimiser ses coûts de transport entre plusieurs dépôts et clients.

La recherche opérationnelle est une discipline des mathématiques appliquées qui s'intéresse à l'application du savoir mathématique aux autres domaines.

La programmation linéaire, la programmation mathématique, la programmation floue, la programmation multicritère, la programmation stochastique, la programmation dynamique, l'optimisation et la recherche opérationnelle ; la théorie des graphes ; la théorie des jeux ; la théorie du contrôle optimal, l'analyse numérique, les bio-mathématiques, la bio-informatique, la théorie de l'information ; les probabilités et les statistiques, les mathématiques financières et l'actuariat ; la cryptologie et, jusqu'à un certain point, la combinatoire et la géométrie finie ; telle qu'appliquée à l'analyse des réseaux, ainsi qu'une bonne partie de ce qu'on appelle l'informatique sont autant de domaines d'application des mathématiques.

Le boum de la recherche opérationnelle coïncide avec la seconde guerre mondiale. Historiquement, le premier terme introduit fut celui de **programmation linéaire** inventé par George Dantzig dans les années 1940. Le terme programmation vient de l'usage du mot programme par les forces armées américaines pour établir des horaires de formation et des choix logistiques que Dantzig étudiait à l'époque. L'emploi du terme « programmation » avait également un intérêt pour débloquer des crédits en une époque où la planification devenait une priorité des gouvernements.

Outils scientifiques :

- Mathématiques Appliquées (Optimisation, Probabilités, Algèbre, Graphes, Jeux, Décision,...) ;
- Informatique (Algorithmique, Complexité, Contraintes)
- La recherche opérationnelle est avant tout un outil d'aide à la décision.

L'approche de la recherche opérationnelle face à un problème applicatif consiste à :

- élaborer un modèle (résultat d'un consensus entre le demandeur et le chercheur) ;
- développer un algorithme de résolution exacte ou approchée ;
- évaluer la qualité des solutions produites par l'algorithme dans l'environnement réel du problème.

Le chercheur opérationnel cherchera à fournir :

- un outil (logiciel) aussi générique que possible, i.e : utilisable et performant sur un ensemble d'instances ;
- non une solution d'une instance particulière.

La programmation linéaire occupe une place centrale de l'optimisation, car les problèmes de PL sont les problèmes d'optimisation les plus faciles - toutes les contraintes y étant linéaires. Beaucoup de problèmes réels de RO peuvent être exprimés comme un problème de PL. Pour cette raison un grand nombre d'algorithmes pour la résolution d'autres problèmes d'optimisation sont fondés sur la résolution de problèmes linéaires.

La programmation linéaire est essentiellement appliquée pour résoudre des problèmes d'optimisation à moyen et long terme (problèmes stratégiques et tactiques, dans le vocabulaire de la recherche opérationnelle).

Les domaines d'application de ces problèmes sont très nombreux aussi bien dans la nature des problèmes abordés (planification et contrôle de la production, distribution dans les réseaux) que dans les secteurs d'industrie : industrie manufacturière, énergie (pétrole, gaz, électricité, nucléaire), transport (aériens, routiers et ferroviaires), télécommunications, industrie forestière, finance. La recherche opérationnelle a aussi des applications dans le domaine de l'énergie.

Elle est couramment utilisée dans l'industrie pétrolière, principalement dans l'établissement des plans de production à long terme, à moyen terme, annuel, trimestriel et mensuel. Les résultats permettent aux décideurs d'avoir un guide pour faire les meilleurs choix dans les investissements, dans l'approvisionnement des bruts, dans l'utilisation des unités de raffinage, dans les canaux de distribution les plus rentables.

De même, les opérateurs du marché de l'électricité font largement appel à la recherche opérationnelle tant pour des problèmes stratégiques (par exemple des investissements sur le réseau) que pour des questions plus opérationnelles (stabilité du réseau, prévisions. . .).

Dans le cadre de l'industrie manufacturiere, la recherche opérationnelle permet notamment de trouver des plans de productions (ordonnancement de production), de disposer au mieux les machines dans un atelier, de diminuer le gaspillage des matières premières (problèmes de découpe) ou de l'énergie ou bien encore d'optimiser le conditionnement et la livraison des produits intermédiaires ou finis.

La recherche opérationnelle peut aider le décideur lorsque celui-ci est confronté à un problème combinatoire, aléatoire ou concurrentiel.

Un problème est dit combinatoire lorsqu'il comprend un grand nombre de solutions admissibles parmi lesquelles on cherche une solution optimale ou proche de l'optimum.

La plupart des problèmes combinatoires de la R.O. ont un graphe comme support :

- TSP généralisé (graphe des communications) ;
- Ordonnancement (graphe des précédences) ;
- Transport (graphe des liaisons) ;
- Emplois du temps (graphe des incompatibilités) ;
- Flots et circulations (graphe des liaisons)

La théorie des graphes sert de support à la résolution d'un vaste échantillon de problèmes, notamment certains issus de l'algorithmique classique, tels que la recherche du plus cour chemin entre deux endroits, le problème du voyageur de commerce (dans lesquels on cherche le chemin le plus court passant par n points), les problèmes d'ordonnancement de tâche, les problèmes de planning ou encore les problèmes d'optimisation de flux (algorithme de Ford-Fulkerson). Elle s'est également développée dans diverses disciplines telles que la chimie, la biologie, les sciences sociales. Depuis le début du XXe siècle, elle constitue une branche à part entière des mathématiques, grâce aux travaux de König, Menger, Cayley puis de Berge et d'Erdös.

De manière générale, un graphe permet de représenter la structure, les connexions d'un ensemble complexe en exprimant les relations entre ses éléments : réseau de communication, réseaux routiers, interaction de diverses espèces animales, circuits électriques,. . .

Les graphes constituent donc une méthode de pensée qui permet de modéliser une grande variété de problèmes en se ramenant à l'étude de sommets (ou noeud) et d'arcs (ou arête).

Les derniers travaux en théorie des graphes sont souvent effectués par des informaticiens, du fait de l'importance qu'y revêt l'aspect algorithmique.

Application 1 : Problème de production

Une entreprise fabrique plusieurs produits, ce qui exige des ressources particulières (matières premières, machines, personnel ...) en quantités limitées.

Chaque produit rapporte un certain bénéfice (connu) à l'entreprise.

Question

Quels produits l'entreprise doit-elle fabriquer et en quelle quantité pour réaliser le bénéfice total le plus élevé ?

- **Modélisation ou formulation du problème :** variable de décision, fonction objectif et contraintes (modèle mathématique = PL).
- **Résolution manuelle (graphique et simplexe) :** pour des problème de petite taille.
- **Résolution informatique (problème de grande taille) :** Solveur Excel, LpSolve, Python, R, ...

Application 2 : Problème de transport

Une entreprise dispose de plusieurs dépôts contenant chacun un certain nombre de containers.

Différents magasins commandent des containers.

On connaît le coût de transport de chaque dépôt aux magasins.

Question

Quelle est l'organisation des livraisons des containers pour minimiser le coût total de transport ?

Application 3 : Problème d'affectation

n tâches doivent être affectées à n machines (1 tâche par machine).

Le coût d'exécution de chaque tâche par chacune des machines est connu.

Question

Comment trouver l'affectation qui minimise le coût total ?

Application 4 : Problème du voyageur de commerce

Un voyageur de commerce doit visiter n villes.

La distance entre chaque ville est donnée.

Question

Trouver le plus court trajet passant par les n villes.

Application 5 : Problème du sac à dos

On dispose de n objets donc chacun d'eux a une masse (ou volume) et une valeur connues.

Question

Comment transporter certains objets, afin de maximiser la valeur totale sans dépasser la capacité massique (ou volumique) du sac ?

On l'utilise aussi pour modéliser les situations suivantes, quelquefois en tant que sous-problème :

- dans des systèmes d'aide à la **gestion de portefeuille** : pour équilibrer la sélectivité et la diversification dans le but de trouver le meilleur rapport entre rendement et risque pour un capital placé sur plusieurs actifs financiers (actions. . .) ;
- dans le **chargement de bateau ou d'avion** : tous les bagages à destination doivent être amenés, sans être en surcharge ; ou l'optimisation d'un navire de charge (cargo polyvalent ou porte-conteneurs) en fonction de plusieurs critères : poids, volume,

destination, marchandises spécifiques (dangereuses ou frigorifiques), importance des clients etc.

Face a un problème concret de type R.O : Les étapes suivantes sont nécessaires

- **La modélisation.** Une transformation du problème concret par :
  - des équations et/ou inéquations mathématiques ;
  - des graphes (schématisation).
- **Résolution.** Techniques principales
  - Programmation linéaire.
  - Programmation dynamique.
  - Programmation en nombres entiers.
  - Optimisation dans les réseaux.
  - Programmation non linéaire.
  - Modèles stochastiques
- **Interprétation**
- **Prise de décision**

Objectif : **apprendre à modéliser les problèmes réels et à résoudre les programmes linéaires**.

- De nombreux problèmes réels peuvent être exprimés comme des programmes linéaires.
- Les programmes linéaires peuvent être résolus efficacement par certains algorithmes.
- Ce sont d'excellents exemples de questions pratiques dont la résolution nécessite une combinaison de méthodes algorithmiques, de mathématiques élémentaires et de bon sens.

1. Formulation d'un programme linéaire (Modélisation).
2. Introduction + une approche naïve : la méthode graphique.
3. L'algorithme du simplexe.
4. Simplexe : comment éviter les pièges.
5. Dualité et interprétation économique.
6. Programmation linéaire en nombres entiers.

Formulation du modèle mathématique (Modélisation)

Objectif : traduire le problème réel en langage mathématique

1. Définir les variables de décision (ex : x_1, x_2, ..., x_n ou bien $x_{ij}=$ quantité transportée de i vers j).
2. Écrire la fonction objectif ou la fonction économique (ex : minimiser le coût total $\sum c_i x_i$ ou bien $\sum c_{ij} x_{ij}$).
3. Établir les contraintes (ex : respect des capacités, de la demande, etc.).

C'est ici qu'on obtient un modèle mathématique de type linéaire, non linéaire, entier, etc.

La programmation linéaire est un cas particulier de la programmation mathématique, où l'on cherche à trouver un optimum (maximum ou minimum) d'une fonction linéaire à plusieurs variables, ces variables étant soumises à un système d'inéquations (d'équations) linéaires.

Modélisation (ou formulation) sous la forme d'un Programmation Linéaire

La formulation d'un problème d'optimisation comporte toujours les trois étapes suivantes :

- La première étape consiste à **choisir les variables** du problème.
  - On appelle variable toute quantité utile à la résolution du problème dont le modèle doit déterminer la valeur.
  - Cette définition permet de différencier les variables des paramètres, qui sont des données qui peuvent varier, par exemple d'une période à l'autre ou d'un scénario à l'autre.
- L'étape 2 consiste à **formuler mathématiquement l'objectif**.
  - On appelle fonction objectif (ou fonction économique) d'un problème d'optimisation le critère de choix entre les diverses solutions possibles.
- La troisième étape c'est la formulation des **contraintes** du problème.
  - On appelle contraintes du problème toutes les relations limitant le choix des valeurs possibles des variables. Ces relations peuvent être de simples bornes sur les variables.

1. **Fonction objectif (ou fonction économique) linéaire**

   Min le coût / Max le profit

   $$min/max\ Z = c_1x_1 + c_2x_2 + \cdots + c_nx_n$$

2. **contraintes inégalité ou égalité linéaire**

   - **satisfaire la demande**

     $$a_1x_1 + a_2x_2 + \cdots + a_nx_n \geq b_1$$

   - **avec des ressources limitées**

     $$a'_1x_1 + a'_2x_2 + \cdots + a'_nx_n \leq b'_1$$

     ou

     $$a''_1x_1 + a''_2x_2 + \cdots + a''_nx_n = b''_1$$

3. **quantités produites :** $x_1, x_2, \ldots, x_n \geq 0$. x_i : variable de décision du problème

La programmation linéaire nous permet de résoudre un programme linéaire composé d'une fonction linéaire (fonction objectif) à optimiser (maximiser ou minimiser) sous certaines contraintes exprimées en un système d'équations ou d'inéquations linéaires.

Un fois le problème posé, la démarche peut s'effectuer en 3 étapes :

1. **Modélisation**
   - **Variables de Décision.** Les variables de décision sont les inconnues que nous cherchons à déterminer.
   - **Fonction Objectif (ou fonction économique).** La fonction objectif est une fonction linéaire à maximiser ou minimiser.
   - **Contraintes.** Les contraintes sont des systèmes d'inégalités ou d'égalités linéaires que les variables doivent respecter.
2. **Résolution**
3. **Solution**

$\boxed{\text{Problème réel}} \Longrightarrow \boxed{\text{Modèle}} \Longrightarrow \boxed{\text{Méthode}} \Longrightarrow \boxed{\text{Solution}}$

Exemple 1

Une entreprise spécialisée dans la fabrication de matériels informatiques, propose à son catalogue d'ordinateurs des centaines de référence. Pour simplifier, on ne s'intéresse ici qu'a deux types d'ordinateurs : le Core i3 et le Core i7. Chacun d'eux comporte un processeur (le même) mais les deux modèles différent en particulier par le nombre de barrettes mémoires. Plus précisément, le Core i3 comporte 2 barrettes alors que le Core i7 en comporte 6. Le marché pour ces composants est tel qu'on ne peut espérer acheter auprès des fournisseurs habituels plus de 10 000 processeurs pour le trimestre à venir et plus de 48 000 barrettes. Une autre limitation intervient sur la production : l'assemblage est caractérisé, en particulier, par une opération délicate, qui pour le Core i3 est de 3 minutes alors que pour le Core i7 elle n'est que d'une minute. On ne dispose a priori pour l'assemblage de ces deux types de machines que de 24 000 minutes pour le trimestre à venir.

Enfin, compte tenu des conditions actuelles du marché, on peut espérer retirer un profit de 40 000 FCFA sur le Core i3 et de 80 000 FCFA sur le Core i7.

L'objectif est de déterminer les quantités de chacun des deux types d'ordinateurs à fabriquer de manière à obtenir le plus grand profit possible.

Questions

1. Modéliser ce programme sous forme d'un programme linéaire en précisant les variables de décision, la fonction objectif et les contraintes.
2. Écrire la forme matricielle.
3. Résoudre graphiquement le modèle obtenu.
4. Tp : Solveur d'Excel.

Exemple 2

Une usine fabrique des chaises et des tables. Chaque semaine l'usine commande la même quantité de bois, connaît le temps de travail de ses employés et la durée de fonctionnement de ses machines. Sachant qu'une table rapporte 6 euros et une chaise 4 euros, quel est le mix chaises-tables qui maximise le revenu de l'usine ?

	Tables	Chaises	Quantité disponible
Équipement	3	9	81
Main d'oeuvre	4	5	55
Bois	2	1	20

1. Modéliser ce programme sous forme d'un programme linéaire.
2. Écrire le programme obtenu sous la forme matricielle.
3. Résoudre graphiquement le problème linéaire.
4. Tp : Solveur d'Excel

Exemple 3

Une entreprise de relations publiques souhaite réaliser un sondage d'opinion. Chaque employé peut effectuer, par jour, soit 80 interviews par téléphone (IT), soit 40 interviews directes (ID). Un employé ne peut effectuer qu'un seul type de sondage par jour. Afin d'obtenir un échantillon représentatif, trois critères doivent être respectés :

- Au moins 3000 interviews.
- Au moins 1000 interviews par téléphone (IT).
- Au moins 800 interviews directe (ID).

L'employé effectuant les IT perçoit une rémunération de 150 UM, tandis que celui réalisant les ID perçoit 170 UM.

Question

1. Modéliser ce problème
2. Écrire le programme obtenu sous la forme matricielle.
3. Résoudre graphiquement le problème linéaire.
4. Tp : Solveur d'Excel.

Représentation matricielle d'un PL

- **Exemple 1. Considérons le modèle linéaire suivant :**

$$(P_1)\begin{cases} Max\ z = 5x_1 + 4x_2 \\ 2x_1 + x_2 \leq 40 \\ x_1 + 2x_2 \leq 24 \\ x_1, x_2 \geq 0 \end{cases}$$

On définit :

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}; \quad c = \begin{pmatrix} 5 \\ 4 \end{pmatrix}; \quad b = \begin{pmatrix} 40 \\ 24 \end{pmatrix}; \quad A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}.$$

Alors on obtient la forme matricielle suivante :

$$(P_1)\begin{cases} Max\ \ z = c^T x \\ Ax \leq b \\ x \geq 0 \end{cases}$$

- **Exemple 2. Considérons le modèle linéaire suivant :**

$$(P_2)\begin{cases} Max\ z = 3x_1 + 5x_2 \\ 2x_1 + x_2 \geq 8 \\ x_1 + 2x_2 \geq 6 \\ x_1, x_2 \geq 0 \end{cases}$$

On définit :

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}; \quad c = \begin{pmatrix} 3 \\ 5 \end{pmatrix}; \quad b = \begin{pmatrix} 8 \\ 6 \end{pmatrix}; \quad A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}.$$

Alors on obtient la forme matricielle suivante :

$$(P_2)\begin{cases} Max\ \ z = c^T x \\ Ax \geq b \\ x \geq 0 \end{cases}$$

- **Exemple 3. Considérons le modèle linéaire suivant :**

$$(P_3)\begin{cases} Min\ z = 4x_1 + 3x_2 \\ x_1 + 2x_2 \geq 8 \\ 3x_1 + x_2 \geq 6 \\ x_1, x_2 \geq 0 \end{cases}$$

On définit :

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}; \quad c = \begin{pmatrix} 4 \\ 3 \end{pmatrix}; \quad b = \begin{pmatrix} 8 \\ 6 \end{pmatrix}; \quad A = \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix}.$$

Alors on obtient la forme matricielle suivante :

$$(P_3)\begin{cases} Min\ \ z = c^T x \\ Ax \geq b \\ x \geq 0 \end{cases}$$

- **Exemple 4. Considérons le modèle linéaire suivant :**

$$(P_4)\begin{cases} Min\ z = 5x_1 + 4x_2 \\ x_1 + 2x_2 \leq 6 \\ 2x_1 + x_2 \leq 6 \\ x_1 + 3x_2 \leq 9 \\ x_1, x_2 \geq 0 \end{cases}$$

On définit :

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}; \quad c = \begin{pmatrix} 5 \\ 4 \end{pmatrix}; \quad b = \begin{pmatrix} 6 \\ 6 \\ 9 \end{pmatrix}; \quad A = \begin{pmatrix} 1 & 2 \\ 2 & 1 \\ 1 & 3 \end{pmatrix}.$$

Alors on obtient la forme matricielle suivante :

$$(P_4)\begin{cases} Min\ \ z = c^T x \\ Ax \leq b \\ x \geq 0 \end{cases}$$

- **Exemple 5. Considérons le modèle linéaire suivant :**

$$(P_5)\begin{cases} Max\ z = 4x_1 + 3x_2 \\ x_1 + 2x_2 \geq 8 \\ 3x_1 + x_2 \leq 6 \\ x_1, x_2 \geq 0 \end{cases} \Leftrightarrow \begin{cases} Max\ z = 4x_1 + 3x_2 \\ -x_1 - 2x_2 \leq -8 \\ 3x_1 + x_2 \leq 6 \\ x_1, x_2 \geq 0 \end{cases}$$

On définit :

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}; \quad c = \begin{pmatrix} 4 \\ 3 \end{pmatrix}; \quad b = \begin{pmatrix} -8 \\ 6 \end{pmatrix}; \quad A = \begin{pmatrix} -1 & -2 \\ 3 & 1 \end{pmatrix}.$$

Alors on obtient la forme matricielle suivante :

$$(P_5)\begin{cases} Max\ \ z = c^T x \\ Ax \leq b \\ x \geq 0 \end{cases}$$

Les hypothèses de la programmation linéaire

Comme toute opération visant à modéliser des phénomènes concrets et réels de notre quotidien, la programmation linéaire suppose que certaines hypothèses soient vérifiées afin que le programme linéaire final décrivant le problème soit valable :

- La contribution de chaque variable à la valeur de la fonction objectif est proportionnelle à la valeur de cette variable.
- La contribution de chaque variable est indépendante de la valeur des autres variables.
- Les variables de décision du problème $x_1, x_2 \ldots$sont positives et ils prennent des valeurs réelles non entières.
- Les paramètres du problème autres que les variables de décision ont une valeur connue avec certitude.
- La fonction objectif est décrite par une fonction linéaire des variables de décision.

Résolution du modèle, analyse et interprétation des résultats

Objectif : trouver la ou les meilleures solutions possibles.

1. Choisir une méthode de résolution :
   - Méthode du simplexe (pour la programmation linéaire).
   - Méthodes graphiques (pour deux variables).
   - Branch and Bound (pour les entiers).
   - ou utilisation de logiciels (Excel Solver, etc.).
2. Calculer la solution optimale et vérifier la faisabilité.

Objectif : donner un sens concret aux résultats numériques

1. Traduire les valeurs des variables dans le contexte réel du problème.
2. Vérifier la cohérence avec la situation pratique.
3. Réaliser une analyse de sensibilité (impact des variations de données).

Mise en œuvre de la solution (prise de décision)

Objectif : appliquer concrètement les recommandations

1. Présenter la solution aux décideurs.
2. Intégrer les résultats dans le processus décisionnel.
3. Mettre en place le suivi et l'évaluation des résultats réels.

Révision du modèle si nécessaire

Si le contexte change (nouvelles contraintes, nouvelles données), on met à jour le modèle et on recommence le processus.

En résumé, les étapes clés sont : Définition du problème, Modélisation mathématique, Résolution du modèle, Interprétation des résultats, Mise en œuvre et suivi.

Exemple 1. Optimisation du planning de développement logiciel

Problème : Une équipe de développement dispose de trois développeurs, pouvant travailler chacun 40 heures par semaine. Elle doit déterminer le nombre d'heures allouer à trois modules :

- Module A : Interface utilisateur
- Module B : Base de données
- Module C : API backend

Chaque module génère une valeur ajoutée (gain) et nécessite un nombre d'heures de travail selon les compétences des développeurs.

Modules	Gain par h (unités)	Temps min (h)	Temps max (h)
A	5	10	40
B	8	20	50
C	6	10	30

L'objectif est de déterminer la répartition optimale des heures à allouer à chaque module de manière à maximiser la valeur totale du projet, tout en respectant les contraintes de temps disponibles pour chaque développeur.

Exemple 2. Production de deux types de logiciels

Problème : Une entreprise de développement souhaite produire deux types de logiciels :

- x_1 : nombre de logiciels standards,
- x_2 : nombre de logiciels sur mesure.

Chaque type nécessite un certain nombre d'heures de travail et rapporte un profit.

Produit	heure de Conception	heures de test	Profit unitaire
Logiciel standard	2 h	1 h	40
Logiciel sur mesure	1 h	2 h	50

L'entreprise dispose au maximum de :

- 8 heures pour la conception,
- 6 heures pour les tests.

1. Modéliser ce problème.
2. Résoudre graphiquement le modèle linéaire obtenu.

Exemple 3. Installation des serveurs

Problème : Un administrateur réseau souhaite installer des serveurs dans deux centres de données afin de répondre à la demande de trois régions.

Centres	Coût par unité	Région 1	Région 2	Région 3
Centre 1	4	1	3	2
Centre 2	6	2	1	4

Chaque région doit recevoir au moins les unités suivantes de services :

$$R_1 \geq 30, \quad R_2 \geq 20, \quad R_3 \geq 40.$$

1. Modéliser ce problème.
2. Résoudre graphiquement le modèle linéaire obtenu.

Exemple 4. Problème de ressources

Problème : Un responsable de projet souhaite déterminer la quantité minimale de deux types de ressources nécessaires pour atteindre un certain niveau de performance.

Chaque ressource contribue positivement à cette performance.

Ressources	Contribution performance	Coût
R_1	4	5
R_2	3	3

On veut au moins 24 unités de performance totale.

1. Modéliser ce problème.
2. Résoudre graphiquement le modèle linéaire obtenu.

Exemple 5. Problème de maximisation

On note :

- x_1 : quantité de produits P_1 à fabriquer,
- x_2 : quantité de produits P_2 à fabriquer.

$$(P_1)\begin{cases} Max \quad z = 3x_1 + 2x_2 \\ x_1 + 2x_2 \leq 8 \\ 3x_1 + 2x_2 \geq 12 \\ x_1, x_2 \geq 0 \end{cases}$$

1. Identifier les étapes de la modélisation (variables, fonction objectif, contraintes) de P_1.
2. Déterminer la nature du problème.
3. Donner la forme matricielle.
4. La résolution graphique étape par étape si c'est un problème à deux variables.

- **Étape 1** : Tracer les droites associées aux contraintes.
- **Étape 2** : Déterminer la région (ou domaine) admissible D_0.
- **Étape 3** : Trouver les sommets (ou points extrêmes) de la région admissible.
- **Étape 4** : Calculer z aux sommets de la région faisable (ou évaluer z pour chaque point extrême).

Projets : Modéliser le problème suivant

Projet 1 : Problème de production

Une entreprise produit n biens différents. Pour cela, elle dispose de m types de ressources. Elle possède b_j unités de la ressource j pour $j = 1, \dots, m$. La production d'une unité du bien i entraîne un profit égal à c_i, pour $i = 1, \dots, n$. Pour produire une unité du bien i, elle a besoin de a_{ij} unités de chaque ressource j.

Quelle quantité de chaque bien doit produire l'entreprise afin de maximiser le profit ?

Projet 2 : Problème de rangement

On considère n objets que l'on doit ranger dans un sac dont la capacité massique (volumique) est P. Chaque objet i a un poids p_i et une valeur v_i. On cherche un rangement de valeur maximale.

Projet 3 : Problème d'alimentation

On se propose de réaliser une alimentation économique pour des bestiaux, qui contient obligatoirement 4 sortes de composants nutritifs, A, B, C et D. L'industrie alimentaire produit précisément deux aliments M et N qui contiennent ces composants : 1 Kg d'aliment M contient 100 g de A, 100 g de C, 200 g de D ; 1 Kg d'aliment N contient 100 g de B, 200 g de C, 100 g de D. Un animal doit consommer par jour au moins : 0.4 Kg de A ; 0.6 Kg de B ; 2 Kg de C ; 1.7 Kg de D. L'aliment M coûte 10 euros le Kg et N coûte 4 euros le Kg.

Quelles quantités d'aliments M et N doit-on utiliser par jour et par animal pour réaliser l'alimentation la moins coûteuse ?

Projet 4 : Problème de médecine

Un spécialiste en médecine a fabriqué un médicament (des pilules) pour guérir les sujets atteints d'un rhume. Ces pilules sont fabriquées selon deux formats :

- Petite taille : elle contient 2 grains d'aspirine, 5 grains de bicarbonate et 1 grain de codéine.
- Grande taille : elle contient 1 grain d'aspirine, 8 grains de bicarbonate et 6 grains de codéine.

Pour guérir la maladie, le sujet a besoin de 12 grains d'aspirine, 74 grains de bicarbonate et 24 grains de codéine.

Déterminer le nombre de pilules minimales à prescrire au sujet pour qu'il soit guérit.

Projet 5 : Problème de recrutement

Une société industrielle doit recruter trois techniciens afin de renforcer certain de ses quatre groupes de travail dont le but est de maximiser le nombre de pièces fabriquées par l'ensemble de ses employés. La rentabilité de ces techniciens se mesure par le nombre de pièces additionnelles que va produire chaque groupe.

Nombre de techniciens	1	2	3
Groupe 1	50	80	90
Groupe 2	40	50	85
Groupe 3	20	30	80
Groupe 4	30	40	100

Déterminer le meilleur recrutement.

Projet 6 : Politique d'achat optimale

Un marchant d'un certain bien dispose d'une capacité de stockage (pour ce bien) de K unités et on admet que le coût de stockage est nul. Sur n périodes ($n \in \mathbb{N}^*$), on connaît les demandes d_i et les prix unitaires d'achat p_i de ce bien. Sachant que les achats se font en début de période, on démarre et finit avec un stock nul, les quantités sont entières, le problème consiste à trouver la meilleure stratégie d'approvisionnement c'est-à-dire déterminer la politique optimale d'achat.

Période i	1	2	...	n
Demande d_i	d_1	d_2	...	d_n
Prix unitaire	p_1	p_2	...	d_n

Programmation linéaire

1. Tout problème mathématique qui consiste à optimiser (maximiser un profit, minimiser un coût) une fonction linéaire de plusieurs variables liées par des relations linéaires est dit programmation linéaire.
2. L'importance de la PL résulte du fait qu'un très grand nombre de modèles constituent des extensions de programmes linéaires. Ce qui fait qu'elle est rencontrée dans diverse domaines.
   - Pb d'affectation (Programme Linéaire en Variables Binaires (PLVB))
   - Pb du sac 'a dos (Programme Linéaire en Nonbres Entiers (PLNE) ou en VB)
   - etc.
3. **Méthode de résolution**
   - Méthode graphique (uniquement 2 variables de décisions)
   - Méthode du simplexe
   - Dualité et interprétation économique
   - Applications : allocation de ressources, optimisation de coûts logiciels

   De manière générale, la résolution d'un problème de PL nécessite la mise en oeuvre d'un algorithme.

Modélisation

La formulation d'un problème mathématique d'un programme linéaire suit trois étapes.

1. **Choix des variables de décision ou structurelles :** les inconnus qui interviennent dans la fonction linéaire à optimiser.
2. **Identification de la fonction économique ou fonction objectif** : la fonction à optimiser. Puis spécification de la nature d'optimisation c'est à dire s'il s'agit d'une maximisation ou d'une minimisation.
3. **Détermination des contraintes** : se sont les restrictions des variables (relations entre variables). Elles peuvent s'interpréter comme les disponibilités des ressources (contraintes technologiques).

Exemple Problème de production

Une entreprise produit n biens différents. Pour cela, elle dispose de m types de ressources. Elle possède b_j unités de la ressource j pour $j = 1, \ldots, m$. La production d'une unité du bien i entraîne un profit égal à c_i, pour $i = 1, \ldots, n$. Pour produire une unité du bien i, elle a besoin de a_{ij} unités de chaque ressource j.

Quelle quantité de chaque bien doit produire l'entreprise afin de maximiser le profit ?

Modélisation

- **Choix des variables de décision :** Soit x_i la quantité de bien i qu'il faut produire ;
- **Identification de la fonction économique** z **:** si on produit une quantité x_i du bien i, alors cela rapporte un profit de $c_i x_i$. Donc

$$z = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

- **Détermination des contraintes :**
  - Contraintes technologiques qui limitent les unités de fabrication (disponibilité des ressources) :

$$\sum_{i=1}^{n} a_{ij}x_i \leq b_j,\ avec\ j = 1, \ldots, m$$

  - Contraintes de non négativité (ou la positivité des quantités produites) des quantités produites : $x_i \geq 0$, $i = 1, \ldots, n$.

On obtient ainsi le programme linéaire suivant :

$$(P) : \begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \leq b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \leq b_m \\ x_1 \geq 0\ o\ i = 1, 2, \ldots, n \end{cases}$$

Représentation matricielle

où sous forme matricielle

$$(P) : \begin{cases} Max\ \mathbf{z} = \mathbf{c}^{\mathbf{T}}\mathbf{x} \rightarrow \textit{Fonction objectif} \\ \mathbf{Ax} \leq \mathbf{b} \rightarrow \textit{Contraintes} \\ \mathbf{x} \geq \mathbf{0} \rightarrow \textit{Conrainte de positivits} \end{cases}$$

avec

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}; \quad c = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix}; \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

$$A = \begin{pmatrix} a_{11}x_1 & a_{12}x_2 & \dots & a_{1n}x_n \\ a_{21}x_1 & a_{22}x_2 & \dots & a_{2n}x_n \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1}x_1 & a_{m2}x_2 & \dots & a_{mn}x_n \end{pmatrix}$$

c^T : la transposée de c.

Exemple

Un atelier fabrique deux produits P_1 et P_2 à base de deux principales matières premières M_1 et M_2. Il dispose de 200 unités de matière M_1 et de 640 unités de M_2. Le produit P_1 nécessite une quantité de 5 unités de M_1 et de 10 unités de M_2. Quant au produit P_2, il nécessite une quantité de 4 unités de M_1 et de 16 unités de M_2. La quantité de P_1 que peut produire la machine en une journée ne peut excéder 25.

Les marges bénéficiaires de P_1 et P_2 sont respectivement 1 000 F CFA et 1 400 F CFA.

Quelle quantité doit produire l'atelier afin de maximiser le bénéfice journalier ?

Modélisation

- Variables de décision : Soit x_1 et x_2 désignent respectivement les quantités produites de P_1 et P_2.
- Fonction objectif : $z = 1000x_1 + 1400x_2$

- Les contraintes :
  - Contrainte liée à la disponibilité M_1 : $5x_1 + 4x_2 \leq 200$
  - Contrainte liée à la disponibilité M_2 : $10x_1 + 16x_2 \leq 640$
  - Contrainte liée à la machine : $x_1 \leq 25$
  - Contraintes de non négativité : $x_1 \geq 0$ et $x_2 \geq 0$

En résumé, on obtient le modèle linéaire suivant :

$$(P) : \begin{cases} Max\ z = 1000x_1 + 1400x_2 \\ 5x_1 + 4x_2 \leq 200 \\ 10x_1 + 16x_2 \leq 640 \\ x_1 \leq 25 \\ x_1, x_2 \geq 0 \end{cases}$$

Un vecteur

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \Big(x_1, x_2, \ldots, x_n\Big)^T$$

est dit solution réalisable ou admissible s'il vérifie l'ensemble des contraintes.

L'ensemble des solutions réalisables est appelé domaine réalisable (c'est un polyèdre convexe).

Lorsque celui ci est vide, on dit que le problème n'admet pas de solution.

PL : Résolution graphique

La formalisation d'un programme est une tâche délicate mais essentielle car elle conditionne la découverte ultérieure de la bonne solution. Elle comporte les mêmes phases quelles que soient les techniques requises ultérieurement pour le traitement (programmation linéaire ou programmation non linéaire) :

- La détection du problème et l'identification des variables. Ces variables doivent correspondre exactement aux préoccupations du responsable de la décision. En programmation mathématique, les variables sont des variables décisionnelles.
- La formulation de la fonction économique (ou fonction objectif) traduisant les préférences du décideur exprimées sous la forme d'une fonction des variables identifiées.
- La formulation des contraintes. Il est bien rare qu'un responsable dispose de toute liberté d'action. Le plus souvent il existe des limites à ne pas dépasser qui revêtent la forme d'équations ou d'inéquations mathématiques.

Résolution de programmes linéaires ou Approches de Résolution

1. On dispose d'un outil (la PL) pour modéliser des problèmes.
2. Comment résoudre les problèmes à l'aide de la PL ?

   - **Résolution graphique.** Pour les problèmes à deux variables, on peut tracer les contraintes sous forme de droites et identifier la région admissible.
   - **La méthode du simplexe.** Pour les problèmes à $n > 2$ variables, on navigue parmi les sommets de la région admissible pour trouver la solution optimale.
   - **La méthode des deux phases**
   - **Outils Numériques (logiciels pour résoudre des programmes linéaires).** Des logiciels comme Excel Solveur, Python (PuLP, SciPy) ou MATLAB permettent de résoudre des problèmes de grande taille.

- Lorsqu'il n'y a que **deux variables de décision**, un problème linéaire peut être résolu de manière purement graphique. Cette résolution graphique permet de mettre en évidence certaines propriétés que possède n'importe quel problème de programmation linéaire.

  $\rightarrow$ A chaque couple de variables x_1 et x_2, on associe un point du plan dont les coordonnées correspondent aux valeurs des variables.

  $\rightarrow$ Les variables étant positives, ces points sont situés dans l'orthant positif (le quart Nord-Est du plan) ou

$$\mathbb{R}^+ = \left\{(x_1, x_2) \in \mathbb{R}^2 : x_1 \geq 0,\ x_2 \geq 0\right\}$$

- La résolution graphique ne concerne que des problèmes avec 2 variables alors que les problèmes réels peuvent en avoir plusieurs milliers, voire centaine de milliers.

Lorsque le nombre de variables est au plus égal à deux, on peut envisager une résolution graphique du problème. Cette technique nous permettra de comprendre la méthode générale du simplexe.

Chaque contrainte est représentée par un demi plan.

Nous déterminons le domaine réalisable en vérifiant le sens des inégalités de chacune des contraintes. Ce domaine est un ensemble convexe.

La résolution graphique d'un problème linéaire consiste à tracer la droite qui sépare les demi-plans pour chaque contrainte tout en conservant le demi-plan acceptable, c'est-à-dire le demi-plan des solutions réalisables pour la contrainte. L'intersection des différents demi-plans de toutes les contraintes sans oublier les contraintes de positivité forme le polygone des solutions, appelé aussi "région des solutions admissibles".

Représentation graphique des solutions d'un problème de PL (2 variables)

Contexte

- La représentation graphique est possible uniquement pour 2 variables (x_1 et x_2).
- Objectif : visualiser la zone de faisabilité et identifier la solution optimale.

Étapes de la représentation graphique.

▷ **Étape 1 : Définir le problème**

- Fonction objectif : $Z = c_1x_1 + c_2x_2$ (à maximiser ou minimiser).
- Contraintes linéaires :

$$\begin{gathered} a_1x_1 + a_2x_2 \leq b_1 \\ a'_1x_1 + a'_2x_2 \leq b'_1 \\ x_1, x_2 \geq 0 \end{gathered}$$

▷ **Étape 2 : Tracer les contraintes**

- Convertir chaque contrainte en équation de droite : $a_1x_1 + a_2x_2 = b_1$ et $a'_1x_1 + a'_2x_2 = b'_1$.
- Tracer ces droites dans le plan x_1, x_2.
- Déterminer le demi-plan valide selon le sens de l'inégalité ($\leq$ ou $\geq$).

▷ **Étape 3 : Déterminer la zone de faisabilité**

- La zone de faisabilité est l'intersection des demi-plans de toutes les contraintes, souvent un polygone convexe.
- Chaque point à l'intérieur de cette zone respecte toutes les contraintes.

▷ **Étape 4 : Identifier la solution optimale**

- **Pour maximisation** : déplacer la droite de Z vers le maximum possible dans la zone de faisabilité.
- **Pour minimisation** : déplacer vers le minimum possible.
- La solution optimale se trouve toujours sur un sommet (vertex) du polygone de faisabilité

Exemple 1 : Problème de maximisation ($\leq$)

Le responsable d'une décision ne dispose que de sa compétence pour réaliser une formalisation correcte du problème posé car il n'existe pas de méthode en la matière. Un moyen d'acquérir cette compétence est l'apprentissage comme proposé dans les exemples suivants :

$$(P_1) : \begin{cases} Max\ z = 1000x_1 + 1400x_2 \\ 5x_1 + 4x_2 \leq 200 \\ 10x_1 + 16x_2 \leq 640 \\ x_1 \leq 25 \\ x_1, x_2 \geq 0 \end{cases}$$

- **Représentation des contraintes.** Représenter toutes les contraintes.
- **Domaine réalisable.** La région délimitée par des segments représente la frontière entre le domaine admissible et non admissible.
- **Solution optimale.** $x_1^* = 16$ et $x_2^* = 30$. La valeur optimale est $z^* = 58000$ euros.

Exemple 3 : Problème de minimisation

Résoudre graphiquement les problèmes linéaires suivants :

$$(P_2) : \begin{cases} Min\ z = x_1 + x_2 \\ x_1 + 2x_2 \geq 8 \\ 2x_1 + x_2 \geq 7 \\ x_1, x_2 \geq 0 \end{cases} \qquad (P_3) : \begin{cases} Min\ z = 2x_1 + 3x_2 \\ 4x_1 + x_2 \geq 5 \\ 3x_1 + 2x_2 \geq 6 \\ x_1 + 2x_2 \geq 3 \\ x_1, x_2 \geq 0 \end{cases}$$

$$(P_4) : \begin{cases} Min\ z = x_1 - x_2 \\ x_1 - x_2 \leq 3 \\ -0,5x_1 + x_2 \leq 4 \\ -2x_1 + x_2 \leq 2 \\ x_1, x_2 \geq 0 \end{cases} \qquad (P_5) : \begin{cases} Min\ z = 2x_1 - 10x_2 \\ x_1 - x_2 \geq 0 \\ x_1 - 5x_2 \geq -5 \\ x_1, x_2 \geq 0 \end{cases}$$

Exemple 3 : Problème de maximisation (type $\geq$ et $\leq$)

On considère le problème linéaire suivant :

$$(P_3) : \begin{cases} Max\ z = 2x + 3y \\ x + y \leq 4 \\ x + 2y \leq 5 \\ 4x - y \geq 2 \\ x_1, x_2 \geq 0 \end{cases}$$

1. Tracer sur un graphique, le domaine réalisable de ce modèle linéaire.
2. Calculer les coordonnées de chaque point extrême.
3. Déterminer la solution optimale de ce modèle et évaluer la fonction économique z en ce point ($x^* = 3$; $y^* = 1$ et $z^* = 9$).

Exemple 4 : Problème de minimisation (type $\geq$)

On considère le problème linéaire suivant :

$$(P_4) : \begin{cases} Min\ z = 2x - 10y \\ x - y \geq 0 \\ x - 5y \geq -5 \\ x, y \geq 0 \end{cases}$$

1. Tracer sur un graphique, le domaine réalisable de ce modèle linéaire.
2. Calculer les coordonnées de chaque point extrême.
3. Déterminer la solution optimale de ce modèle et évaluer la fonction économique z en ce point.

Exemple 5 : Problème de maximisation

On considère le problème linéaire suivant :

$$(P_5) : \begin{cases} Max\ z = \lambda x + 7y \\ x - y \leq 6 \\ 2x + y \leq 4 \\ y \leq 2 \\ x, y \geq 0 \end{cases}$$

1. Tracer sur un graphique, le domaine réalisable de ce modèle linéaire.
2. Calculer les coordonnées de chaque point extrême.
3. Déterminer la solution optimale de ce modèle et évaluer la fonction économique z en ce point.

Exemple 6 : Problème de maximisation ($\leq$)

Considérons le modèle linéaire suivant :

$$(P_6) : \begin{cases} Max\ z = 3x_1 + 5x_2 \\ x_1 \leq 4 \\ 2x_2 \leq 12 \\ 3x_1 + 2x_2 \leq 18 \\ x_1, x_2 \geq 0 \end{cases}$$

1. Tracer sur un graphique, le domaine réalisable de ce modèle linéaire.
2. Calculer les coordonnées de chaque point extrême.
3. Déterminer la solution optimale de ce modèle et évaluer la fonction économique z en ce point.

Exemple 1 : Résoudre graphiquement le PL suivant :

$$Max\ Z = 3x_1 + 2x_2$$

Sous contraintes :

$$\begin{cases} x_1 \leq 4 \\ 2x_2 \leq 12 \\ 3x_1 + 2x_2 \leq 18 \\ x_1, x_2 \geq 0 \end{cases}$$

Étape 1 : Simplifions les contraintes

$$\begin{cases} x_1 \leq 4 \\ x_2 \leq 6 \\ 3x_1 + 2x_2 \leq 18 \end{cases}$$

Étape 2 : Détermination des points d'intersection (sommets du domaine faisable)

- Intersection avec les axes.

- Vérifions les points d'intersection entre contraintes.

Étape 3 : Sommets du domaine faisable. $O = (0,0)$; $E_1 = (4,0)$; $E_2 = (4,3)$; $E_3 = (2,6)$ et $E_4 = (0,6)$.

Étape 4 : Calcul de la fonction objectif. $z = 3x_1 + 5x_2$.

Point	x_1	x_2	$z = 3x_1 + 5x_2$
O	0	0	0
E_1	4	0	12
E_2	4	3	27
E_3	2	6	36
E_4	0	6	30

La solution optimale est : $(x_1^*, x_2^*) = (2,6)$ et la **valeur optimale** est $Z_{max}^* = 36$.

Exemple 2 : Résoudre graphiquement le PL suivant :

$$(P)\begin{cases} Max\ Z = 3x_1 + 2x_2 \\ x_1 + x_2 \leq 4 \\ x_1 \leq 2 \\ x_2 \leq 3 \\ x_1, x_2 \geq 0 \end{cases} \tag{1.2}$$

Solution graphique

- Tracer les droites : $x_1 + x_2 = 4$, $x_1 = 2$, $x_2 = 3$.
- Zone de faisabilité : polygone convexe formé par l'intersection.
- Évaluer Z aux sommets : $(0,0)$, $(0,3)$, $(1,3)$, $(2,2)$, $(2,0)$.
- Maximum de $Z^* = 6 + 4 = 10$ à $x_1^* = 2$ et $x_2^* = 2$.

Remarque

- La solution optimale d'un PL à 2 variables se trouve toujours sur un sommet de la zone faisable.
- Permet d'illustrer la méthode avant d'utiliser le Simplexe pour plusieurs variables.
- Pour des problèmes avec deux variables, on peut résoudre graphiquement (aide à comprendre la structure du problème)

Terminologie

- **Solution** : affectation de valeurs aux variables.
- **Solution réalisable :** solution réalisable si les valeurs satisfont l'ensemble des contraintes.
- **Région réalisable :** ensemble des solutions réalisables.

Notion de solution optimale et faisable

1. Solution faisable

- Une solution faisable est un ensemble de valeurs des variables de décision qui respecte toutes les contraintes du problème.
- Autrement dit : c'est un point dans l'espace des variables qui satisfait les inégalités et égalités du problème.

Une solution faisable est tout vecteur de décision $x = (x_1, x_2, \ldots, x_n)$ qui satisfait toutes les contraintes du problème.

Exemple : On considère le problème :

$$\begin{cases} \textit{Maximiser} \quad Z = 3x_1 + 2x_2 \\ x_1 + x_2 \leq 4 \\ x_1 \leq 2 \\ x_1, x_2 \geq 0 \end{cases} \tag{1.3}$$

Une solution faisable est tout couple $x = (x_1, x_2)$ qui respecte ces conditions.

Exemple 1.

- $(x_1, x_2) = (2, 2)$ est faisable, car elle respecte toutes les contraintes.
- $(x_1, x_2) = (3, 2)$ est non faisable, car $x_1 + x_2 = 5 > 4$.

Exemple 2. Pour un problème PL avec $x_1 + x_2 \leq 4$, $x_1 \geq 0$, $x_2 \geq 0$:

- $x_1 = 1$, $x_2 = 2$ est faisable car $1 + 2 \leq 4$ et les variables sont non négatives.
- $x_1 = 3$, $x_2 = 2$ n'est pas faisable car $3 + 2 = 5 > 4$. L'ensemble de toutes les solutions faisables est appelé zone de faisabilité ou polyèdre faisable.
- La zone de faisabilité est convexe si toutes les contraintes sont linéaires.

La région faisable (ou ensemble des solutions réalisables) est donc l'ensemble de tous les points qui satisfont les contraintes.

2. Solution optimale

Une solution optimale est une solution faisable qui maximise ou minimise la fonction objectif, selon le problème.

- Si on **maximise**, la solution optimale donne la **valeur maximale** de la fonction objectif.
- Si on **minimise**, elle donne la **valeur minimale**.

Solution optimale (optimal solution). Une solution optimale est une solution faisable qui optimise la fonction objectif :

- **Maximisation :** la solution qui donne le plus grand Z.
- **Minimisation :** la solution qui donne le plus petit Z.

Exemple (suite du précédent) : Maximiser $z = 3x_1 + 2x_2$

Les coins de la région faisable sont : $(0, 0)$; $(0, 4)$; $(2, 2)$; $(2, 0)$.

En calculant : $z(0, 0) = 0$; $z(0, 4) = 8$; $z(2, 2) = 10$ et $z(2, 0) = 6$.

La solution optimale est $(x_1^*, x_2^*) = (2, 2)$, et la valeur optimale est $z^*_{max} = 10$

Résumé

Notion	**Définition**	**Exemple**
Solution faisable	Satisfait toutes les contraintes du problème	$(x_1^*, x_2^*) = (2, 2)$
Solution optimale	Faisable et minimise/maximise la fonction objectif	$(x_1^*, x_2^*) = (2, 2)$ avec $z = 10$

1. En PL, la solution optimale se trouve toujours sur un sommet du polyèdre faisable (théorème fondamental).
2. Il peut exister plusieurs solutions optimales si la droite de la fonction objectif est parallèle à une arête du polyèdre.
3. **Schéma mental**
   - **Zone faisable** : tous les points qui respectent les contraintes (polygone convexe pour 2 variables).
   - **Solution optimale** : point(s) dans cette zone où la fonction objectif atteint sa valeur maximale ou minimale.

Proposition

S'il en existe, il y a toujours une solution optimale sur un sommet (point extrême) de la région réalisable

Corollaire

Pour trouver l'optimum, il "suffit" d'examiner les points extrêmes de la région réalisable

Polyèdres et points extrêmes

- Un polyèdre convexe est l'ensemble des solutions d'un système fini d'inégalités linéaires.
- L'ensemble des solutions admissibles d'un PL est donc un polyèdre convexe.

Théorème

Si le polyèdre formé par l'ensemble des solutions d'un PL est borné, alors il existe au moins une solution optimale et l'une d'elles est obtenue sur un point extrême.

La méthode graphique pour la résolution d'un problème linéaire à deux variables est très facile à appliquer. Sa difficulté augmente par l'augmentation des variables dans le système. Elle devient difficile pour trois variables et, voire impossible, au delà de trois inconnus dans le système étudié.

Afin d'ôter cette difficulté, une méthode appelée méthode du simplexe a été proposée et développée par Dantzig afin de résoudre des problèmes de programmation linéaire avec plusieurs variables.

- La résolution manuelle ne peut se faire que pour des petits problèmes.
- Le recours à l'informatique est indispensable pour des problèmes de grande dimension.
- Pour les problèmes réels, les entreprises font appel à des solveurs professionnels.
- On appelle solveur un programme (ou un sous programme) informatique permettant de résoudre un problème d'optimisation sous contraintes.

Exemple 2

Le gérant d'un entrepôt souhaite renouveler le matériel de sécurité de son établissement. Il a besoin au minimum de

- 90 paires de chaussures de sécurité,
- 240 casques de sécurité,
- 240 paires de gants.

Une première entreprise de vente lui propose un lot A comprenant 2 paires de chaussures, 4 casques et 8 paires de gants pour 200 euros. Une deuxième entreprise vend pour 400 euros un lot B de 3 paires de chaussures, 12 casques et 6 paires de gants.

Pour répondre à ses besoins, le gérant acheté x lots A et y lots B.

Déterminer graphiquement, en précisant la démarche suivie, le nombres de lots A et de lots B à acheter pour avoir une dépense minimale.

Résolution d'un PL avec l'aide de Géogébra

Le Géogébra est un logiciel de mathématiques qui regroupe la géométrie, de l'algébre, de calcul formel et scientifique, de l'algorithmique, de la statistique etc. GeoGebra est un logiciel de géométrie dynamique dans le plan qui permet de créer des figures dans lesquelles il sera possible de déplacer des objets afin de vérifier si certaines conjectures ne sont pas dues uniquement à un positionnement particulier des objets. En résumé, il regroupe presque toutes les mathématiques.

$$(P1)\begin{cases} \mathbf{Max}\ \mathbf{Z_1} = \mathbf{4x} + \mathbf{5y} \\ 2x + y \leq 800 \\ x + 2y \leq 700 \\ y \leq 300 \\ x,\ y \geq 0 \end{cases} \qquad (P2)\begin{cases} \mathbf{Min}\ \mathbf{Z_2} = \mathbf{2x_1} + \mathbf{3x_2} \\ 4x_1 + x_2 \geq 5 \\ 3x_1 + 2x_2 \geq 6 \\ x_1 + 2x_2 \geq 3 \\ x_1,\ x_2 \geq 0 \end{cases}$$

1. Présentation et prise en main du logiciel Géogébra.
2. Résoudre les programmes linéaires en utilisant la méthode graphique (Géogébra)

PL : Comment utiliser le Solveur d'Excel ?

La résolution manuelle ne peut se faire que pour des **petits problèmes**. Le recours à l'informatique est indispensable pour des problèmes de **grande dimension**. Pour les problèmes réels, les entreprises font appel à des solveurs professionnels. On appelle solveur un programme (ou un sous programme) informatique permettant de résoudre un problème d'optimisation sous contraintes.

- **Solveurs libres :** Python, Scilab, LP-SOLVE (Mixed Integer Linear Programming), CLP (Linear Programming), etc.
- **Solveurs commerciaux :** Solveur Excel (Linear Programming), OOQP (Linear Programming), etc.

Pour résoudre un problème de PL avec Excel, vous devez d'abord activer le complément Solveur, puis configurer votre feuille de calcul pour représenter la fonction objectif et les contraintes. Enfin, utilisez l'outil Solveur en définissant la cellule objectif, les cellules de variables et les contraintes pour trouver la solution optimale.

1. Introduction. Le solveur d'EXCEL est un outil puissant d'optimisation et d'allocation de ressources. Il permet de trouver le minimum, le maximum ou la valeur au plus près d'une donnée tout en respectant les contraintes qu'on lui soumet. Nous pouvons donc résoudre c'est-à-dire trouver la meilleure solution pour un modèle de la programmation linéaire en utilisant ce solveur. Cet outil permet aussi de déterminer les valeurs à saisir dans certaines cellules de manière à **optimiser une ou plusieurs valeurs cibles** situées dans d'autres cellules, conformément à des contraintes définies. Il s'agit d'un véritable outil d'**optimisation intégrée** dans Excel. Nous verrons également que l'utilisation du Solveur permet de paramétrer un certain nombre de contraintes, afin d'ajuster très précisément le résultat obtenu.

2. Quand utiliser le solveur. Utilisez le solveur lorsque vous recherchez la valeur optimale d'une cellule donnée (la fonction économique) par ajustement des valeurs de plusieurs autres cellules (les variables) respectant des conditions limitées supérieurement ou inférieurement par des valeurs numériques (c'est à dire les contraintes)

Il s'agit d'un outil de simulation et d'optimisation intégré à Excel, qui permet d'effectuer une série de calculs afin de déterminer la ou les valeurs à attribuer à certaines cellules pour atteindre un résultat donné. Il permet aussi de calculer automatiquement la meilleure valeur possible d'une cellule (appelée cellule objectif), en modifiant certaines autres cellules (appelées cellules variables), tout en respectant des contraintes définies par l'utilisateur.

En quelques sortes cet outil est une version bien plus évoluer et bien plus puissant de la valeur cible.

Résumé

Le Solveur cherche la solution optimale (maximum, minimum ou valeur donnée) à un problème dans Excel.

• Le Solveur d'Excel est un module d'optimisation numérique intégré au tableur Microsoft Excel. Il résout des problèmes où il faut optimiser une fonction objectif (minimiser ou maximiser une cellule) sous contraintes, en ajustant des variables de décision. Il repose sur des méthodes d'optimisation linéaire, non linéaire ou entière selon la nature du problème.

• Par exemple, une entreprise peut utiliser le Solveur pour déterminer la quantité de produits à fabriquer afin de maximiser son profit, tout en respectant les contraintes de main-d'œuvre, de matières premières et de budget.

3. Comment activer le Solveur ? Le Solveur est une fonctionnalité cachée qui est présente sous la forme d'un complément Excel.

Allez dans Fichier → Options → Compléments → dans la zone "Gérer", sélectionnez "Compléments Excel" → puis cliquez sur Atteindre → Cochez la case Complément Solveur → puis cliquez sur OK.

On retrouve le Solveur dans l'onglet **DONNÉES**.

4. Configurer la feuille de calcul.

1. **Définissez les variables de décision :** Réservez des cellules pour les variables de votre problème (par exemple, les quantités à produire).
2. **Configurez la fonction objectif :** Saisissez une formule qui représente la fonction objectif (par exemple, un bénéfice ou un coût) dans une cellule dédiée. Utilisez les cellules des variables de décision dans cette formule.
3. **Saisissez les contraintes :** En utilisant d'autres cellules, configurez les formules qui représentent vos contraintes (par exemple, les limites de ressources). Ces formules devront également faire référence aux cellules des variables de décision.

5. Utiliser l'outil Solveur.

- Accédez au Solveur en allant dans l'onglet Données et en cliquant sur Solveur dans le groupe "Analyse".
- Dans la boîte de dialogue Solveur :

- **Définissez l'objectif :** Sélectionnez la cellule contenant votre fonction objectif. Choisissez si vous souhaitez Maximiser, Minimiser ou atteindre une Valeur spécifique.

- **Modifiez les variables de décision :** Sélectionnez la plage de cellules qui contiennent vos variables de décision.

- **Ajoutez les contraintes :**

Cliquez sur "Ajouter" pour ajouter chaque contrainte.

Sélectionnez la cellule de la contrainte, le symbole d'inégalité $\leq$ ou $\geq$ ou $=$ etc et la cellule de référence de votre contrainte.

- **Lancez la résolution** : Cliquez sur Résoudre. Le Solveur calculera la solution optimale et l'affichera dans vos cellules de variables.
- **Conservez la solution** : Assurez-vous de sélectionner "Résultats du Solveur" pour conserver la solution trouvée.

- Assurez-vous que l'option Rendre les variables sans contrainte non négatives est cochée (courant dans la plupart des problèmes de PL).
- Dans la zone Sélectionner la méthode de résolution, choisissez Simplexe LP (Linear Programming) pour les problèmes de programmation linéaire.
- Cliquez sur Résoudre.
- Une boîte de dialogue Résultats du solveur apparaîtra, vous permettant de conserver la solution du solveur ou de restaurer les valeurs d'origine. Vous pouvez également demander des rapports (Résultats, Sensibilité, Limites).

En suivant ces étapes, vous pourrez utiliser Excel pour trouver la solution optimale à votre problème de programmation linéaire.

6. Comment utiliser le Solveur ?

Exemple 1 (Problème de maximisation). On considère le modèle linéaire suivant :

$$(P1)\begin{cases} Max\ \ Z_1 = 10x_1 + 15x_2 + 25x_3 \\ x_1 + 2x_2 + 4x_3 \leq 20000 \\ x_1 + x_2 + 3x_3 \leq 16000 \\ 3x_1 + 5x_2 + 3x_3 \leq 48000 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

1. Résoudre le programme linéaire en utilisant le Solveur Excel.
2. En déduire les rapports de réponses, sensibilité et limites.
3. Commentez chaque rapport.

Configurer la feuille de calcul

- **1) Définir les variables du modèle.** Les cellules variables sont les valeurs respectives des différents variables (cellules jaunes).
- **2) Définir la fonction objectif.** La cellule cible est celle contenant la formule exprimant la valeur à optimiser (valeur de z) (cellules jaunes).
- **3) Établir les contraintes.** Les contraintes sont les valeurs imposées (premier membres des contraintes) (cellules jaune).

Configuration de l'outil Solveur

Afin d'optimiser la fonction économique, nous allons utiliser la commande Solveur. . .du menu Données. Il est fort probable que les commandes du solveur n'apparaissent pas encore dans le menu Données.

Lancer le Solveur d'Excel

1) Spécifications de la cellule cible (objectif à définir). Dans la zone Cellule cible à définir, tapez la référence de la cellule ($\boxed{\ldots}$ cellule en jaune) que vous voulez minimiser, maximiser (c'est à dire la fonction économique)

- Pour maximiser la cellule cible, il faut choisir le bouton Max.
- Pour minimiser la cellule cible, choisir le bouton Min.
- Pour que la cellule cible se rapproche d'une valeur donnée, choisir le bouton Valeur et indiquer la valeur souhaitée dans la zone à droite du bouton.

Remarque.

- Allez plus vite en cliquant directement sur la cellule à spécifier plutôt que de taper sa référence au clavier.
- La cellule cible doit contenir une formule dépendant directement ou indirectement des cellules variables spécifiées dans la zone Cellules variables.

2) Spécification des cellules variables. Tapez dans la zone Cellule variables les références des cellules (cellule en jaune (vide)) devant être modifiées par le solveur jusqu'à ce que les contraintes du problème soient respectées et que la cellule cible atteigne le résultat recherché.

Remarques.

- Allez plus vite, cliquons directement sur les cellules à spécifier plutôt que de taper leurs références au clavier.
- Nous pouvons spécifier jusqu'à 200 cellules variables.
- Dans le programme initial, on définit les cellules variables par des zéros.

3) Spécifications des contraintes. A l'aide des boutons Ajouter, Modifier et Supprimer de la boîte de dialogue, établir la liste de contraintes dans la zone Contraintes.

Remarques.

- Après avoir cliqué dans chaque case à compléter, il suffit de cliquer dans les cellules correspondantes directement sur la feuille Excel, puis $\boxed{\text{OK}}$ pour confirmer.
- Une contrainte peut être une limité inférieurement ($\leq$), supérieurement ($\geq$) ou limité aux nombres entiers (opérateur ent)...

- La cellule à laquelle l'étiquette Cellule fait référence contient habituellement une formule qui dépend des cellules variables.
- Le solveur gère jusqu'à 200 contraintes.

Ensuite, on choisit la méthode de résolution Simplex PL, puis on clique sur Résoudre. À ce moment, Excel effectue un certain nombre de traitements afin de déterminer le scénario de fabrication qui permet d'optimiser au maximum la marge nette.

$\Longrightarrow$ **Fenêtre. Résultat du Solveur.**

4) Les options du solveur Options. Cette boîte de dialogue permet de contrôler les caractéristiques avancées de résolution et de précision du résultat. En général, la plupart des paramètres par défaut sont adaptés à la majorité des problèmes d'optimisation.

Concentrons-nous sur quelques options plus spécifiques :

- **Échelle automatique.**

- **Afficher le résultat des itérations.** Interrompt le solveur et affiche les résultats produits par chaque itération. Cette option permet de suivre étape après étape les différents programmes de base.
- **Ignorer les contraintes du nombre entier.**

5) Résolution et résultat. Une fois tous les paramètres du problème mis en place, le choix du bouton $\boxed{\text{Résoudre}}$ amorce le processus de résolution du problème. Nous obtenons alors une de ces réponses :

- Première solution de base.
- Solution finale.

Que faire des résultats du solveur :

- Garder la solution trouvée par le solveur ou rétablir les valeurs d'origine dans votre feuille de calcul.
- Créer un des rapports intégrés du solveur en sélectionnant celui qui nous concernera.

6) Rapport des réponses (solution). Le Solveur a trouvé une solution satisfaisante respectant toutes les contraintes et les conditions d'optimalité. Au bas de l'écran, vous pouvez obtenir le rapport des réponses en sélectionnant la feuille correspondante.

Ce rapport donne l'évolution des cellules variables et de la cellule cible. On remarque donc bien qu'il y a eu une maximisation $(P1)$. Le rapport rappelle les différentes valeurs des contraintes, leurs formules, et dans quelle mesure elles ont été respectées.

- **Lié ou saturée :** La valeur finale de la cellule contenant une contrainte atteint effectivement la valeur maximale.

  La Marge (que l'on appelle temps mort) est donc égale à 0.
- **Non lié ou non saturée :** La contrainte est respectée mais la valeur finale de la cellule n'est pas égale à la valeur maximale ou minimale de la contrainte.

  La Marge (temps mort) vaut ici 0 pour les 3 contraintes.

On peut demander à Excel d'afficher un rapport en cochant Rapport de plan pour voir les détails du calcul. Ensuite, on sélectionne Réponse (rapports) puis on clique sur OK : une autre feuille de calcul s'affichera alors automatiquement.

7) Rapports de sensibilité. Les valeurs appelées admissibles d'augmentation et de diminution pour les cellules variables indiquent l'ampleur maximale des variations du coefficient de l'objectif qui laisse la solution optimale inchangée, en supposant que tous les autres coefficients restent constants. Le rapport de sensibilité associe un coût d'ombre à chacune des contraintes.

Analyse de sensibilité

Nous abordons ici l'analyse de sensibilité (pour une étude graphique, le cas général sera étudié en Travaux Pratiques) de la résolution d'un problème de programmation linéaire. Dans un premier temps nous étudions les conséquences d'une variation d'un coefficient de la fonction objectif, puis celles du second membre d'une contrainte (ressources) donnée.

Tout problème de programmation linéaire donne lieu à ce type d'analyse et les résultats que nous allons obtenir ici sont généraux.

En général, cette étude de sensibilité intervient lorsque :

- on veut évaluer les conséquences d'un changement de politique modifiant certaines données du problème,
- les données du problème ne sont pas exactement connues, où il faut déterminer dans quelle circonstance cela affecte la solution proposée.

Rapports des limites. On observe l'impact de chaque terme sur le résultat de la fonction objectif.

Je vais vous guider à travers un exemple simple de résolution d'un problème de programmation linéaire avec Excel Solveur, et vous expliquer comment créer et interpréter les rapports (Réponse, Sensibilité et Limites).

Problème de Maximisation de Profit

Une entreprise fabrique 2 produits, P_1 et P_2. Les profits unitaires sont de 40 euros pour P_1 et 30 euros pour P_2. La production est soumise à des contraintes de ressources :

- P_1 nécessite 2 heures de travail par unité, P2 nécessite 1 heure de travail par unité. Disponibilité : 100 heures.
- P_1 nécessite 1 unité de matière première, P2 nécessite 2 unités de matière première. Disponibilité : 80 unités.

L'objectif est de maximiser le profit total.

Solution

1. Modélisation

- Variables de décision :
  - x_1 = quantité de P_1 produite
  - x_2 = quantité de P_2 produite
- Fonction objectif : Maximiser $Z = 40x_1 + 30x_2$
- Contraintes :

$$2x_1 + x_2 \leq 100 \text{ (travail)}$$
$$x_1 + 2x_2 \leq 80 \text{ (matière première)}$$
$$x_1, x_2 \geq 0 \text{ (non-négativité)}$$

2. Résolution avec le Solveur d'Excel

- **Préparation des données dans Excel :**

  Dans une feuille Excel, crée un tableau.
  - Définir les variables.
  - Définir la fonction objectif.
  - Établir les contraintes.
- **Configurer le Solveur :**
  - Aller dans Données → Solveur (si Solveur n'est pas activé, active-le via les compléments d'Excel).
  - Paramètres : Objectif à Définir, Type de problème (Max ou Min ou Valeur), les variables et les contraintes.
  - Cliquer sur Résoudre.

- **Générer les Rapports :**
  - Après avoir cliqué sur Résoudre, cochez les options pour générer les rapports : Réponse, Sensibilité et Limites.
  - Excel crée ces rapports dans de nouvelles feuilles.
- Rapport de Réponse : Donne la solution optimale et l'utilisation des ressources.
- Rapport de Sensibilité : Indique comment la solution change si les coefficients (profits, ressources) varient — utile pour l'analyse "what-if".
- Rapport de Limites : Montre les intervalles de variation des variables sans perdre la réalisabilité.

Le Rapport de Sensibilité (ou "Sensitivity Report" en anglais)

C'est un outil crucial dans l'analyse de programmation linéaire. Il fournit des informations sur la stabilité de la solution optimale face à des changements dans les paramètres du modèle (coefficients de la fonction objectif, disponibilités des ressources, etc.). En d'autres termes, il indique jusqu'à quel point la solution optimale peut varier sans changer de base (c'est-à-dire sans que les variables de base et non-base changent de statut).

Rôle du Rapport de Sensibilité. Le Rapport de Sensibilité permet de répondre à des questions comme :

- Quels sont les impacts d'une modification des profits unitaires (ou des coûts) sur la solution optimale ?
- Quelle est la valeur ajoutée d'une unité supplémentaire d'une ressource (contrainte) ?
- Dans quelle mesure peut-on faire varier les coefficients sans changer la structure de la solution ?

En Résumé. Le Rapport de Sensibilité :

- Indique comment la solution optimale change (ou non) si les coefficients (profits, coûts, ressources) varient.
- Fournit des intervalles de validité pour les coefficients de la fonction objectif et les disponibilités des ressources.
- Aide à prendre des décisions en évaluant l'impact de petites modifications des paramètres.

Exemple 2 (Problème de minimisation). On considère le modèle linéaire suivant :

$$(P_2)\begin{cases} Min \;\; z_2 = 2x_1 + 3x_2 \\ 4x_1 + x_2 \geq 5 \\ 3x_1 + 2x_2 \geq 6 \\ x_1 + 2x_2 \geq 3 \\ x_1, x_2 \geq 0 \end{cases}$$

Résoudre ce problème avec le solveur d'Excel et commentez les rapports.

Implémentation pratiques

1. Résolution avec logiciels et bibliothèques
   - Solveurs (Excel, LpSolve, CPLEX, Gurobi, GLPK, etc.)
   - Outils de programmation scientifique (Python : PuLP, Numpy, SciPy, Matplotlib)
   - Interfaces avec Java/C++
2. Mini-projets pratiques
   - Optimisation de la planification d'un sprint agile
   - Gestion des dépendances d'un projet logiciel
   - Affectation des ressources informatiques dans un cloud

Algorithme du simplexe : Méthode des tableaux

Vers un algorithme de résolution

- **Méthode de résolution naïve (graphique) :** énumérer tous les sommets, calculer Z sur ces points, prendre le sommet pour lequel Z est optimisé :
  - **fonctionne** : nombre fini de sommets.
  - **limitation** : ce nombre peut être très grand en général ...

L'algorithme du simplexe (G. B. Dantzig 1947) Algorithme itératif permettant de résoudre un problème de programmation linéaire.

- La méthode du simplexe est une méthode de résolution des problèmes de programmation linéaire. Elle permet de trouver la solution optimale (maximum ou minimum) d'une fonction objectif, tout en respectant un ensemble de contraintes linéaires.
- La méthode du simplexe est une technique algorithmique utilisée pour résoudre les problèmes de programmation linéaire, c'est-à-dire les problèmes d'optimisation dans lesquels la fonction objectif et les contraintes sont des expressions linéaires.

Le principe du simplexe repose sur une idée géométrique : l'ensemble des solutions admissibles forme un polyèdre convexe (une région délimitée par des plans). La méthode du simplexe explore les sommets de ce polyèdre, en passant d'un sommet à un autre adjacent, de manière à améliorer progressivement la valeur de la fonction objectif, jusqu'à atteindre le sommet optimal.

Cette méthode, introduite par George Dantzig en 1947, est aujourd'hui l'une des plus utilisées dans les domaines de la gestion, de l'économie, de la logistique, de la production et de la planification. Elle est à la fois simple dans son principe et puissante dans son efficacité, notamment pour les problèmes de grande dimension.

L'objectif de cette méthode est de déterminer la solution optimale (maximum ou minimum) d'une fonction objectif, tout en respectant un ensemble de contraintes sous forme d'inégalités ou d'égalités linéaires.

Principe d'amélioration locale

À partir d'un sommet, chercher un sommet voisin qui améliore l'objectif.

Principe d'amélioration locale (maximisation) :

Soit x_0 sommet non optimum. Alors il existe x, un sommet voisin de x_0, tel que $Z(x) > Z(x_0)$.

Méthode de résolution : on part d'un sommet x_0 quelconque, on passe à un sommet voisin pour lequel Z augmente, et ainsi de suite.

Forme canonique. Lorsque l'ensemble des contraintes se présente sous forme d'inégalités ($\leq$ ou $\geq$) on parle de forme canonique. Toutefois, il convient de distinguer un programme canonique de type max d'un programme canonique de type min.

1. Forme standard. Un programme linéaire (PL) mis sous la forme particulière où toutes les contraintes sont des équations et toutes les variables sont non négatives est dit sous forme standard. Il est noté (PL=).

2. Variables d'écart et d'excédent. Avant que l'algorithme du simplexe puisse être utilisé pour résoudre un programme linéaire, ce programme linéaire doit être converti en un programme équivalent où toutes les contraintes technologiques sont des équations et toutes les variables sont non négatives.

- Contraintes de type ($\leq$) : Pour chaque contrainte i de ce type, on rajoute une **variable d'écart** e_i, tel que e_i est une variable positive ou nulle.

$$3x_1 + 2x_2 \leq 2 \textit{ se transforme en } 3x_1 + 2x_2 + \mathbf{e_1} = 2,\ e_1 \geq 0.$$

La variable d'écart d'une contrainte représente la quantité disponible non utilisée. C'est l'écart entre la disponibilité et le besoin.

- Contraintes de type ($\geq$) : Pour chaque contrainte i de ce type, on retranche une **variable d'excédent** e_i, tel que e_i est une variable positive ou nulle.

$$3x_1 + 2x_2 \geq 2 \textit{ se transforme en } 3x_1 + 2x_2 - \mathbf{e_1} = 2, \; e_1 \geq 0.$$

- Un programme linéaire qui contient des contraintes (technologiques) de type $\leq$ est noté (PL).
- Un programme linéaire qui contient des contraintes (technologiques) de type ($\leq, \geq, =$) est noté (PG).
- Un programme linéaire (PL) resp (PG) converti tel que toutes les contraintes technologiques sont des équations et toutes les variables sont non négatives est noté (PL=) resp (PG=).

3. Variables de base et variables hors base. Considérons un système d'équations à n variables et m équations où $n \geq m$.

Une solution de base pour ce système est obtenue de la manière suivante :

- On pose $n - m$ variables égales à 0. Ces variables sont appelées variables hors base (VHB).
- On résout le système pour les m variables restantes. Ces variables sont appelées les variables de base (VB).
- Le vecteur de variables obtenu est appelé solution de base (il contient les variables de base et les variables hors base)

Une solution de base est **admissible** si toutes les variables de la solution de base sont ≥ 0.

4. Solutions admissibles. Toute solution de base de (PL=) pour laquelle toutes les variables sont non négatives, est appelée solution de base admissible. Cette solution de base admissible correspond à un point extrême.

5. Résolution du programme linéaire (PL). On considère le modèle linéaire suivant :

$$(P_1)\begin{cases} Max \;\; Z = 3x_1 + 2x_2 \\ x_1 + x_2 \leq 7 \\ 2x_1 + x_2 \leq 9 \\ x_1, x_2 \geq 0 \end{cases}$$

Étape 1 : Forme standard. On va introduire des variables d'écarts e_1 et e_2 positive, puis on transforme le système d'inéquation à un système d'équations :

$$(P_1)\begin{cases} Max \;\; Z = 3x_1 + 2x_2 \\ x_1 + x_2 + e_1 = 7 \\ 2x_1 + x_2 + e_2 = 9 \\ x_1, x_2, e_1, e_2 \geq 0 \end{cases}$$

$n = 4$ (inconnues ou variables) et $m = 2$ (équations).

Étape 2 : Solution de base (solution initiale). si $x_1 = x_2 = 0$ alors

$$si \begin{cases} x_1 = 0 \\ x_2 = 0 \end{cases} \quad alors \quad \begin{cases} e_1 = 7 \\ e_2 = 9 \end{cases}$$

- x_1 et x_2 sont des variables hors bases (VHB).

- e_1 et e_2 sont des variables de bases (VB).

Étape 3 : Tableau initial. recherche de la solution extrême de base, les variables d'écart sont considérées au départ comme variables principales ou variables dans la base. Les variables réelles sont non principales et se trouvent dans le hors base. la fonction objectif est alors nulle.

Le tableau initial se construit de la manière suivante :

- remplir les coefficients des contraintes du (PL=).
- remplir les coefficients de la fonction objectif.

Avec le tableau initial, on prépare le tableau 2 en choisissant respectivement la variable entrante (VE), la variable sortante (VS) et le Pivot :

- **La variable entrante** correspond au coefficient le plus élevé dans la fonction objectif (ou fonction économique). La colonne de la variable entrante prend alors le nom de colonne du pivot.
- **La variable sortante** correspondra au plus petit rapport positif issu de la division de la colonne second membre par la colonne variable entrante. Cette variable sortante se lira dans la base et sur la même ligne que le plus petit rapport positif. La ligne de la variable sortante prend le nom de la ligne du pivot.
- **Le pivot** se trouvera à l'intersection de la ligne pivot et de la colonne pivot.

Résumé

- **Étape 1 : Forme standard.** on introduit des variables d'écart afin de transformer les contraintes d'inégalité en contraintes d'égalité.
- **Étape 2 : Détermination de la solution de base initiale.** Pour déterminer la solution de base initiale, on considère les variables d'écart comme variables de base (VB) et on met les variables de décision à zéro (VHB). Les valeurs des variables de base sont alors directement données par les termes constants des contraintes.
- **Étape 3 : Créer le tableau initial (ou table de simplexe).** on construit le tableau du simplexe initial en y inscrivant les coefficients de la fonction objectif ainsi que ceux des contraintes.
- **Étape 4 : Itérations.** Appliquer des itérations pour améliorer la solution en sélectionnant une variable d'entrée et une variable de sortie.
- **Conditions d'arrêt.** Vérifier les conditions d'arrêt pour déterminer si la solution optimale a été trouvée.

TD 3 : Algorithme des tableaux de simplexe

Dans les exercices suivants, appliquer l'algorithme tableaux de simplexe pour trouver la solution optimale de programmation linéaire.

Exercice 1 : Problème de maximisation

Soit le problème d'optimisation suivant :

$$(P1)\begin{cases} \text{Maximiser } Z = 240x_1 + 160x_2 \\ \textbf{sc} \\ x_1 + 2x_2 \leq 150 \\ 4x_1 + 2x_2 \leq 400 \\ \text{avec } x_1, x_2 \geq 0 \end{cases}$$

Solution optimale : $x_1 = 83,33$, $x_2 = 33,33$ et la valeur optimale $Z = 2533,33$.

Exercice 2 : Soit le problème d'optimisation suivant :

$$(P2)\begin{cases} \textit{Maximiser } Z = 3x_1 + 5x_2 \\ \textbf{sc} \\ x_1 + 2x_2 \leq 10000 \\ 2x_1 + 3x_2 \leq 12000 \\ x_1 + 4x_2 \leq 15000 \\ \textit{avec } x_1, x_2 \geq 0 \end{cases}$$

Solution optimale : $x_1 = 600$, $x_2 = 3600$ et la valeur optimale $Z = 19800$.

Exercice 3 : Soit le problème d'optimisation suivant :

$$(P3)\begin{cases} \textit{Maximiser } Z = 3x_1 + 4x_2 + 2x_3 \\ \textbf{sc} \\ x_1 + x_2 - x_3 \leq 10 \\ x_1 - 2x_2 + 3x_3 \leq 14 \\ \textit{avec } x_1, x_2, x_3 \geq 0 \end{cases}$$

La solution optimale est : $x_1 = 0$, $x_2 = 44$, $x_3 = 34$ et la valeur optimale $Z = 244$.

Exercice 4 : Soit le problème d'optimisation suivant :

$$(P4)\begin{cases} \quad Maximiser\ Z = 4x_1 + 6x_2 + 3x_3 \\ \qquad\qquad\qquad\quad \textbf{sc} \\ x_1 + 6x_2 + 2x_3 \leq 24 \\ x_1 + 3x_2 - 2x_3 \leq 9 \\ \ avec\ x_1, x_2, x_3 \geq 0 \end{cases}$$

La solution optimale est : $x_1 = 0$, $x_2 = 3,83$, $x_3 = 16,74$ et la valeur optimale $Z = 78,10$.

Problématique 1 : Tous les coefficients de la fonction objectif sont égaux.

1. **Problème :** Lorsque deux ou plusieurs coefficients de la fonction objectif sont égaux et sont les plus positifs (pour un problème de maximisation), il existe un choix multiple de la variable entrante.
2. **Implications :**
   - **Choix arbitraire de la variable entrante** : Si plusieurs variables ont le même coefficient maximal, n'importe laquelle peut être choisie pour entrer dans la base.
   - **Solutions multiples :** Cette situation peut conduire à des solutions optimales alternatives, car différentes variables entrant dans la base peuvent produire le même optimum.
   - **Faisabilité** : Il faut s'assurer que la solution reste faisable à chaque itération, en vérifiant les ratios pour choisir la ligne pivot.

Conclusion : L'égalité des coefficients dans la fonction objectif ne bloque pas la méthode du simplexe mais nécessite de prendre une décision judicieuse lors du choix de la variable entrante et peut révéler l'existence de solutions multiples optimales.

Problématique 2 : Égalité de plusieurs rapports positifs lors du choix de la ligne pivot

1. **Analyse de la Problématique 2 : Égalité de plusieurs rapports positifs.** Lorsqu'on calcule le minimum des rapports entre les termes du membre de droite et les coefficients correspondants dans la colonne pivot, il peut arriver que plusieurs lignes donnent le même rapport minimal positif. Cette situation indique que plusieurs variables candidates peuvent sortir de la base simultanément.
2. **Implications dans la méthode du simplexe :**
   - Le choix de la ligne pivot doit être effectué de manière arbitraire parmi celles présentant le rapport minimal.
   - Cette décision peut conduire à des solutions alternatives optimales, car le tableau final peut dépendre de la ligne pivot choisie.
   - Il est important de vérifier après chaque itération que la solution reste faisable et que l'optimalité est atteinte.

Conclusion : L'égalité de plusieurs rapports positifs ne bloque pas la méthode du simplexe, mais elle nécessite une attention particulière dans le choix de la ligne pivot et peut conduire à des solutions multiples optimales.

Application

Trouver la solution optimale des problèmes linéaires suivants en appliquant la méthode du simplexe :

1. **Les coefficients de la fonction objectif sont égaux :**

$$(P1)\begin{cases} Max\ Z_1 = 12x_1 + 12x_2 \\ x_1 + x_2 \leq 7 \\ 2x_1 + x_2 \leq 9 \\ x_1, x_2 \geq 0 \end{cases} \quad et \quad (P2)\begin{cases} Max\ Z_2 = 3x + 3y \\ x + 2y \leq 8 \\ 3x + y \leq 9 \\ x, y \geq 0 \end{cases}$$

2. **Égalité des plus petits rapports positifs :**

$$(P3)\begin{cases} Max\ Z_1 = 8x_1 + 5x_2 \\ x_1 + x_2 \leq 7 \\ 2x_1 + x_2 \leq 14 \\ x_1, x_2 \geq 0 \end{cases} \quad et \quad (P4)\begin{cases} Max\ Z_2 = 3x + 2y \\ 2x + 2y \leq 4 \\ 3x + y \leq 6 \\ x, y \geq 0 \end{cases}$$

Solution : Méthode de résolution par le simplexe (cas d'égalité des coef de la fonction objectif)

$$(P1)\begin{cases} Max\ Z_1 = 12x_1 + 12x_2 \\ x_1 + x_2 \leq 7 \\ 2x_1 + x_2 \leq 9 \\ x_1, x_2 \geq 0 \end{cases}$$

Étape 1 : Forme standard

- Transformer toutes les contraintes en égalités en introduisant les variables d'écart (et d'excès si nécessaire).
- S'assurer que toutes les variables sont non négatives.
- Vérifier que le problème est bien sous forme maximisation (ou convertir si besoin).

$$(P1)\begin{cases} Max\ Z_1 = 12x_1 + 12x_2 \\ x_1 + x_2 + e_1 = 7 \\ 2x_1 + x_2 + e_2 = 9 \\ x_1, x_2, e_1, e_2 \geq 0 \end{cases}$$

- Les variables x_1 et x_2 sont appelées des variables initiales.
- Les variables e_1 et e_2 sont appelées des variables d'écarts.
- Les coefficients des variables d'écarts dans le fonction objectif sont nuls c'est-à-dire

$$Max\ Z_1 = 12x_1 + 12x_2 + 0e_1 + 0e_2$$

Étape 2 : Solution de base initiale

La solution de base est donnée à l'origine $x_1 = x_2 = 0$ et $e_1 = 7,\ e_2 = 9$.
La valeur de la fonction objectif $Z_1 = 0$.

- Les variables x_1 et x_2 sont appelées des variables hors bases (V.H.B).
- Les variables e_1 et e_2 sont appelées des variables de bases (V.B).

Étape 3 : Construction du tableau initial (premier tableau)

Écrire le premier tableau du simplexe contenant :

- Les variables de base (première colonne),
- Les variables hors base (première ligne puis ajouter les variables de base)
- Les coefficients des contraintes,
- Les valeurs du second membre (C),
- Une colonne supplémentaire pour calculer les rapports,
- Les coefficients de la fonction objectif (la dernière ligne).

	x_1	x_2	e_1	e_2	C	K
e_1	**1**	**1**	1	0	7	K_1 K_1'
e_2	**2**	**1**	0	1	9	K_2 K_2'
Coef de Z_1	**12**	**12**	0	0	0	

On constate que les coefficients de la fonction objectif sont égaux. Dans ce cas, nous avons deux possibilités pour choisir la colonne du pivot, car les

Lorsque les coefficients de la fonction objectif sont identiques, le choix de la variable entrante n'est pas unique. Dans cette situation, plusieurs colonnes peuvent être candidates au pivot. En particulier, les variables x_1 et x_2 ont le même coefficient dans la fonction objectif ; elles constituent donc toutes les deux des options valides pour la variable entrante au cours de l'itération du simplexe.

Voici la méthode à suivre

- **On suppose que x_1 est la variable entrante, puis on calcule les rapports afin de déterminer le pivot.**

$$K_1 = \frac{7}{1} = 7 \quad et \quad K_2 = \frac{9}{2} = 4,5$$

$min(K_1; K_2) = K_2 = \frac{9}{2}$. Ainsi, K_2 représente le plus petit rapport positif, ce qui détermine la variable sortante et identifie l'élément pivot dans le tableau du simplexe.

- **On suppose que x_2 est la variable entrante, puis on calcule les rapports afin de déterminer le pivot.**

$$K_1' = \frac{7}{1} = 7 \quad et \quad K_2' = \frac{9}{1} = 9.$$

$min(K_1'; K_2') = K_1' = 7$. Ainsi, K_1' représente le plus petit rapport positif, ce qui détermine la variable sortante et identifie l'élément pivot dans le tableau du simplexe.

Dans le cas où plusieurs choix de variables entrantes sont possibles, on calcule les rapports positifs associés à chaque hypothèse. On compare ensuite les deux plus petits rapports positifs obtenus. La règle consiste alors à retenir le plus grand de ces deux rapports, afin de déterminer la variable sortante selon la stratégie retenue pour éviter les ambiguïtés ou les cycles.

Conclusion. $max(K_2; K_1') = max\left(7; \frac{9}{2}\right) = 7$. Ainsi, la variable x_2 est choisie comme variable entrante, tandis que la variable e_1, devient la variable sortante. L'élément situé à l'intersection de la colonne de x_2 et de la ligne de e_1 constitue le pivot, dont la valeur est égale à 1.

Étape 4 : Présentation du deuxième tableau du simplexe

	x_1	x_2	e_1	e_2	C	K
x_2	1	1	1	0	7	
e_2	$L_1 = 1$	0	$L_2 = -1$	1	$L_3 = 2$	
Coef de Z_1	$L_4 = 0$	0	$L_5 = -12$	0	$L_6 = -84$	

L'optimalité est atteinte dès lors que tous les coefficients de la fonction objectif (ligne de Z_1) sont devenus négatifs ou nuls. Cela signifie qu'aucune amélioration supplémentaire de la valeur de Z_1 n'est possible, et que la solution courante constitue donc la solution optimale.

Conclusion. La résolution du programme linéaire par la méthode du simplexe conduit à la solution optimale suivante :

$$x_1^* = 0, \quad x_2^* = 7, \quad e_1 = 0, \quad e_2 = 2.$$

La valeur optimale de la fonction objectif s'élève à : $Z_1^* = 84$.

Cette solution respecte l'ensemble des contraintes du modèle et maximise la fonction objectif. Elle correspond donc à l'unique solution optimale obtenue au terme de l'algorithme du simplexe.

Solution : Égalité des rapports

$$(P3)\begin{cases} Max\ Z_1 = 8x_1 + 5x_2 \\ x_1 + x_2 \leq 7 \\ 2x_1 + x_2 \leq 14 \\ x_1, x_2 \geq 0 \end{cases}$$

Étape 1 : Forme standard

$$(P3)\begin{cases} Max\ Z_1 = 8x_1 + 5x_2 \\ x_1 + x_2 + e_1 = 7 \\ 2x_1 + x_2 + e_2 = 14 \\ x_1, x_2, e_1, e_2 \geq 0 \end{cases}$$

Étape 2 : Solution de base initiale

La solution de base est donnée à l'origine $x_1 = x_2 = 0$ et $e_1 = 7$, $e_2 = 14$. La valeur de la fonction objectif $z = 0$.

- Les variables x_1 et x_2 sont appelées des variables hors bases (V.H.B).
- Les variables e_1 et e_2 sont appelées des variables de bases (V.B).

Étape 3 : Construction du tableau initial (premier tableau)

	x_1	x_2	e_1	e_2	C	K
e_1	**1**	**1**	**1**	**0**	7	$K_1 = 7$
e_2	**2**	**1**	**0**	**1**	14	$K_2 = 7$
Coef de Z_1	$\boxed{8}$	5	0	0	0	

- L'élément 8 identifié dans le tableau constitue le pivot, c'est-à-dire l'élément utilisé pour réaliser le changement de base. Par conséquent, la variable x_1 est sélectionnée comme variable entrante pour l'itération du simplexe.

Comment choisir le plus petit rapport positif ?

1. On fait la somme des coefficients de la ligne considérée sauf les contraintes C
2. On divise cette somme par le coefficient de la variable entrante dans cette ligne (la colonne du pivot).

Ce calcul permet de déterminer le rapport positif, qui est utilisé pour identifier la variable sortante et l'élément pivot du tableau. Cette étape garantit que la nouvelle solution reste faisable après le pivotage.

- 1+1+1+0=3 et on divise cette somme par le coefficient correspondant de la colonne du pivot $\frac{3}{1} = 3$.
- 2+1+0+1=4 et on divise cette somme par le coefficient correspondant de la colonne du pivot $\frac{4}{2} = 2$.

Lorsque plusieurs rapports positifs sont calculés, la règle consiste à sélectionner le plus grand rapport parmi eux. Donc $max(3; 2) = 3$.

Le rapport correspondant à cette valeur détermine la variable sortante et l'élément pivot à utiliser pour la prochaine itération du simplexe.

1 est le pivot et e_1 est la variable sortante.

Étape 4 : Deuxième tableau du simplexe

	x_1	x_2	e_1	e_2	C	K
x_1	1	1	1	0	7	
e_2	0	-1	-2	1	0	
Coef de Z_1	0	-3	-8	0	-56	

La condition d'optimalité dans la méthode du simplexe est satisfaite lorsque tous les coefficients de la ligne de la fonction objectif Z_1 sont négatifs ou nuls.

Dans ce cas :

- Aucune variable entrante ne peut améliorer la valeur de Z_1.
- La solution courante est donc optimale.

Cette règle garantit que l'on a atteint la valeur maximale de la fonction objectif, sous les contraintes données.

Conclusion. La résolution du programme linéaire par la méthode du simplexe conduit à la solution optimale suivante :

$$x_1^* = 7, \quad x_2^* = 0, \quad e_1 = 0, \quad e_2 = 0.$$

La valeur optimale de la fonction objectif s'élève à : $Z_1^* = 56$.

Résumé : Principe de l'algorithme du simplexe

Voici l'idée principale de l'algorithme du simplexe :

1. Pour commencer, on sélectionne un sommet initial ;
2. On teste si ce sommet est l'optimum ;
3. Si le sommet que l'on vient d'examiner n'est pas optimal, on se déplace sur un sommet voisin pour lequel la fonction objectif s'améliore et on repasse à l'étape précédente.

Le sommet optimal est atteint lorsqu'aucun des sommets voisins ne permet plus l'amélioration de la fonction objectif.

En s'appuyant sur le principe précédent, voici l'aspect algorithmique de la méthode du simplexe :

Début :

- Saisir la fonction objectif.
- Saisir le type initial.
- Standardiser le programme.
- Convertir le programme en tableau ;

Tant que : Les coefficients de la fonction objectif ne sont pas tous nuls ou négatifs.

• **Sélection de la variable entrante (VE) :** Choisir le coefficient le plus grand (positif) de la fonction objectif.

• **Sélection de la variable sortante (VS) :** Choisir le rapport minimum (le plus petit rapport positif).

• **Définir le pivot.**

Multiplier la ligne du pivot par 1/pivot.

Étendre l'opération aux autres lignes.

Fin Tant que.

Fin.

Dualité

La méthode du simplexe permet d'obtenir une solution optimale à tout programme linéaire en variables continues (non entières). La théorie de la dualité donne d'autres informations caractérisant une solution optimale.

Programme Dual (D)

A tout programme linéaire, on peut associer un autre programme linéaire appelé programme dual. Considérons un programme linéaire sous forme canonique, ce programme sera appelé programme primal (P).

Si un modèle de programmation linéaire possède une solution optimale, il en est de même pour son dual, et les valeurs optimales des deux modèles sont égales.

La solution optimale du dual correspond aux multiplicateurs optimaux. Nous pouvons les lire directement dans le tableau optimal du simplexe : ce sont les coefficients dans la ligne correspondant à l'objectif.

Le coût réduit (correspondant à l'opposé du coefficient dans la ligne de l'objectif) mesure la variation de l'objectif entraîné par une augmentation d'une unité de la valeur de la variable hors-base associée.

Analyse de sensibilité

En général, le coût réduit d'une variable hors-base indique le changement dans l'objectif apporté par une augmentation d'une unité de la valeur de cette variable. Pour les variables d'écart, ce principe peut se formuler ainsi : le coût réduit d'une variable d'écart hors-base indique le changement dans l'objectif apporté par une diminution d'une unité du terme de droite associé.

Ceci est un exemple d'analyse de sensibilité : un paramètre (ici, un terme de droite) est modifié et nous mesurons la sensibilité de la solution optimale à ce changement.

Nous pouvons aussi mesurer la sensibilité de la solution optimale à un changement d'un terme de droite ou d'un coefficient dans l'objectif en résolvant à nouveau le modèle modifié.

Primal	Dual
Variable	Contrainte
Contrainte	Variable
Minimisation	Maximisation
Profit unitaire	Terme de droite
Terme de droite	Coût unitaire
Ligne	Colonne
Colonne	Ligne
Contrainte $\geq$	Contrainte $\leq$

Exemple : Résoudre le PL en utilisant la méthode du simplexe

$$(P)\begin{cases} Min\ Z = 2x_1 + 3x_2 \\ 4x_1 + x_2 \geq 5 \\ 3x_1 + 2x_2 \geq 6 \\ x_1 + 2x_2 \geq 3 \\ x_1, x_2 \geq 0 \end{cases}$$

Exercice : Formuler le problème dual (D) de chacun des programmes linéaires suivants :

$$(P1)\begin{cases} Max\ Z = 2x_1 + 4x_2 + 3x_3 \\ 3x_1 + 4x_2 + 2x_3 \leq 60 \\ 2x_1 + x_2 + 2x_3 \leq 40 \\ x_1 + 3x_2 + 2x_3 \leq 80 \\ x_1, x_2, x_3 \geq 0 \end{cases} \qquad (P2)\begin{cases} Max\ Z = 3x_1 + x_2 - 2x_3 \\ x_1 + 2x_2 \geq 10 \\ 3x_1 - x_2 + x_3 + 7 \\ x_1 + 3x_3 \leq 8 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

$$(P3)\begin{cases} Max\ Z = 10x_1 + 14x_2 \\ x_1 + x_2 \geq 12 \\ x_1 \geq 8 \\ x_2 \leq 6 \\ x_1, x_2 \geq 0 \end{cases} \qquad et\ (P4)\begin{cases} Max\ Z = 400x_1 + 350x_2 + 450x_3 \\ 2x_1 - 3x_2 + 2x_3 \leq 120 \\ 4x_1 + 3x_2 = 160 \\ 3x_1 - 2x_2 + 4x_3 \geq 100 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

Exercice : Utiliser la méthode du duale du simplexe pour résoudre les problèmes suivants :

$$
(P1)\begin{cases} Min\ Z = 35x_1 + 34x_2 \\ 4x_1 + 3x_2 \geq 504 \\ 5x_1 + x_2 \geq 256 \\ 2x_1 + 5x_2 \geq 420 \\ x_1, x_2 \geq 0 \end{cases} \quad et \quad (P2)\begin{cases} Min\ Z = 3x_1 + x_2 + x_3 \\ x_1 + 2x_2 \geq 8 \\ 3x_1 - 2x_2 - x_3 \geq 6 \\ -x_1 - x_2 + 4x_3 \geq 2 \\ x_1, x_2, x_3 \geq 0 \end{cases}
$$

Introduction à la programmation linéaire avec Pyhon en utilisant PuLP/SciPy

La programmation linéaire est un ensemble de techniques utilisées en programmation mathématique, parfois appelée optimisation mathématique, pour résoudre des systèmes d'équations linéaires et d'inéquations tout en maximisant ou en minimisant une fonction linéaire. C'est important dans des domaines tels que le calcul scientifique, l'économie, les sciences techniques, la fabrication, les transports, l'armée, la gestion, l'énergie, etc.

L'écosystème Python propose plusieurs outils complets et puissants pour la programmation linéaire. Vous pouvez choisir entre des outils simples et complexes ainsi qu'entre des outils gratuits et commerciaux. Tout dépend de vos besoins.

- Qu'est-ce que la programmation linéaire et pourquoi c'est important ?
- Quels outils Python conviennent à la programmation linéaire ?
- Comment créer un modèle de programmation linéaire en Python ?
- Comment résoudre un problème de programmation linéaire avec Python ?

- La méthode de base pour résoudre les problèmes de programmation linéaire est appelée la méthode du simplexe, qui comporte plusieurs variantes. Une autre approche populaire est la méthode du point intérieur.
- Les problèmes de PL en nombres entiers mixtes sont résolus à l'aide de méthodes plus complexes et gourmandes en calcul, comme la méthode de branchement et de liaison, qui utilise la programmation linéaire sous le capot. Certaines variantes de cette méthode sont la méthode de branchement et de coupe, qui implique l'utilisation de plans de coupe, et la méthode de branchement et de prix.
- Il existe plusieurs outils Python adaptés et bien connus pour la PL et la programmation linéaire en nombres entiers mixtes. Certains d'entre eux sont open source, tandis que d'autres sont propriétaires. Le fait que vous ayez besoin d'un outil gratuit ou payant dépend de la taille et de la complexité de votre problème ainsi que du besoin de rapidité et de flexibilité.

- Il convient de mentionner que presque toutes les bibliothèques de programmation linéaire (PL) et de programmation linéaire à nombres entiers (PLNE) mixtes largement utilisées sont natives et écrites en Fortran ou C ou C++. En effet, la programmation linéaire nécessite un travail de calcul intensif avec des matrices (souvent volumineuses). De telles bibliothèques sont appelées solveurs. Les outils Python ne sont que des enveloppes autour des solveurs.
- Python convient à la création de wrappers (ou "enveloppe" en français) autour de bibliothèques natives car il fonctionne bien avec C/C++. Nous n'aurons pas besoin de C/C++ (ou Fortran) pour cette séquence, mais si nous souhaitons en savoir plus sur cette fonctionnalité intéressante, consultons les ressources suivantes :
  - Construire un module d'extension Python C.
  - Composants internes de CPython.
  - Extension de Python avec C ou C++

Fondamentalement, lorsque nous définissons et résolvons un modèle, nous utilisons des fonctions ou des méthodes Python pour appeler une bibliothèque de bas niveau qui effectue le travail d'optimisation proprement dit et renvoie la solution à votre objet Python.

- **Plusieurs bibliothèques Python gratuites sont spécialisées pour interagir avec des solveurs de programmation linéaire ou linéaire à nombres entiers mixtes :**
    - Optimisation SciPy et recherche de racine.
    - Pulp
    - Pyomo
    - CVXOPT

Dans cette séquence, nous utiliserons les deux bibliothèques SciPy et PuLP pour définir et résoudre des problèmes de programmation linéaire.

Implémentation en Python d'un PL

Dans cette séquence, nous utiliserons deux packages Python pour résoudre le problème de programmation linéaire décrit ci-dessous :

- **PuLP** est une API (Application Programming Interface) de programmation linéaire en Python permettant de formuler des problèmes et d'appeler des solveurs externes.
- **SciPy** est un package à usage général destiné au calcul scientifique avec Python.

où **API :** C'est une interface qui permet à deux systèmes (ou applications) de communiquer entre eux de manière standardisée et sécurisée.

En résumé : Une API est un pont entre nous (ou notre application) et un service.

- **PuLP** permet de choisir différents solveurs et de formuler des problèmes de manière plus naturelle. Le solveur par défaut utilisé par PuLP est le COIN-OR Branch and Cut Solver (CBC). Il s'appuie sur le solveur de programmation linéaire COIN-OR (CLP) pour les relaxations linéaires et sur la bibliothèque de génération de coupes COIN-OR (CGL) pour la production des coupes.

  PuLP est une bibliothèque open source pour la programmation linéaire en Python. Elle met à disposition tous les outils pour modéliser un problème et les résoudre en faisant appel à différents types de solveurs standards utilisant des algorithmes de résolution différents. C'est en fait un wrapper qui permet la formulation du problème en Python.
- **SciPy** est simple à configurer. Une fois installé, il fournit tout le nécessaire pour commencer. Son sous-package scipy.optimize peut être utilisé pour l'optimisation linéaire et non linéaire.

- Un autre excellent solveur open source est le kit de programmation linéaire GNU (GLPK). Certaines solutions commerciales et propriétaires bien connues et très puissantes sont Gurobi, CPLEX et XPRESS.
- En plus d'offrir une flexibilité dans la définition des problèmes et la possibilité d'exécuter différents solveurs, PuLP est moins compliqué à utiliser que des alternatives comme Pyomo ou CVXOPT, qui nécessitent plus de temps et d'efforts pour être maîtrisées.

Installation de SciPy et PuLP

Deux outils Python pour l'optimisation et le calcul scientifique.

- **PuLP (Python Linear Programming) :**
  - Utilité : Modélisation et résolution de problèmes de programmation linéaire (LP, MILP).
  - Installation : !pip install pulp
  - Vérification : import pulp
  - Importation : from pulp import LpProblem, LpMaximize/LpMinimize, LpVariable
- **SciPy (Scientific Python) :**
  - Utilité : Optimisation, algèbre linéaire, statistiques, traitement du signal, etc.
  - Installation : !pip install scipy
  - Vérification : import scipy
  - Dépendance : Nécessite numpy (installé automatiquement avec scipy).
  - Importation : Pour définir et résoudre des problèmes d'optimisation avec SciPy, nous devon importer scipy.optimize.linprog() : from scipy.optimize import linprog

Maintenant que nous avons importé linprog(), nous pouvons commencer à l'optimiser.

linprog() ne résout que les problèmes de minimisation (pas de maximisation) et n'autorise pas les contraintes d'inégalité avec le signe supérieur ou égal à ($\geq$). Pour contourner ces problèmes, nous devons modifier notre problème avant de lancer l'optimisation :

- Au lieu de maximiser $z = 2x_1 + 3x_2$ (par exemple), nous pouvons minimiser son négatif ($-z = 2x_1 - 3x_2$).
- Au lieu d'avoir le signe supérieur ou égal à, nous pouvons multiplier l'inégalité par -1 et obtenir le signe opposé inférieur ou égal à ($\leq$).
- **Exemple.** On considère le problème de programmation linéaire suivant :

$$\begin{cases} Max \quad Z = 3x_1 + 2x_2 \\ 2x_1 + x_2 \leq 18 \\ 2x_1 + 3x_2 \geq 42 \\ 3x_1 + x_2 \leq 24 \\ -x_1 + 5x_2 = 15 \\ x_1, x_2 \geq 0 \end{cases}$$

Après avoir introduit ces changements, vous obtenez un nouveau système :

$$\begin{cases} Min \quad -Z = -3x_1 - 2x_2 \\ 2x_1 + x_2 \leq 18 \\ -2x_1 - 3x_2 \leq -42 \\ 3x_1 + x_2 \leq 24 \\ -x_1 + 5x_2 = 15 \\ x_1, x_2 \geq 0 \end{cases}$$

Ce système est équivalent à l'original et aura la même solution. La seule raison d'appliquer ces changements est de surmonter les limitations de SciPy liées à la formulation du problème.

Étape 2 : L'étape suivante consiste à définir les valeurs d'entrée

- $obj = [-3, -2]$: cette variable *obj* contient les coefficients de la fonction objectif.

- $Cont1\_ineq = \Big[[2,1],[-2,-3],[3,1]\Big]$: cette variable contient les coefficients du côté gauche des contraintes d'inégalité
- $Cont2\_ineq = [18,-42,24]$: cette variable contient les coefficients du côté droit des contraintes d'inégalité.
- $Cont3\_eq = \Big[[-1,5]\Big]$: cette variable contient les coefficients du côté gauche de la contrainte d'égalité.
- $rhs\_eq = [15]$: cette variable contient les coefficients du côté droit de la contrainte d'égalité.

Remarques : Attention à l'ordre des lignes et des colonnes !

- L'ordre des lignes pour les côtés gauche et droit des contraintes doit être le même. Chaque ligne représente une contrainte.
- L'ordre des coefficients de la fonction objectif et des côtés gauches des contraintes doit correspondre. Chaque colonne correspond à une seule variable de décision

Étape 3 : L'étape suivante consiste à définir les limites de chaque variable dans le même ordre que les coefficients. Dans ce cas, ils sont tous deux compris entre zéro et l'infini positif :

- bnd = [(0, float("inf")), (0, float("inf"))] : Ce sont les contraintes sur les valeurs que peuvent prendre les variables de décision x dans un problème d'optimisation.
- Chaque variable x_i a souvent des bornes inférieures (lower bound) et supérieures (upper bound).
- Variables non négatives (cas courant en optimisation) :
  - $x_1 \geq 0 \longrightarrow$ Bounds (bnd) : $[0, \infty]$
  - $x_2 \geq 0 \longrightarrow$ Bounds (bnd) : $[0, \infty]$

  Cette instruction est redondante car linprog() prend ces limites (de zéro à l'infini positif) par défaut.

En résumé : Bounds (bnd) of x = Limites (min/max) autorisées pour chaque variable.

Remarque : Au lieu de float("inf"), nous pouvons utiliser math.inf, numpy.inf, ou scipy.inf.

Étape 4 : Enfin, il est temps d'optimiser et de résoudre notre problème qui nous intéresse. nous pouvons le faire avec linprog() :

- opt (ou res) = linprog (c=obj, $A\_ub = Cont1\_ineq$, $b\_ub = Cont2\_ineq$, $A\_eq = Cont3\_eq$, $b\_eq = rhs\_eq$, bounds=bnd, method="revised simplex")

Le paramètre c fait référence aux coefficients de la fonction objectif. A_ub et b_ub sont liés respectivement aux coefficients des côtés gauche et droit des contraintes d'inégalité. De même, A_eq et b_eq font référence à des contraintes d'égalité. Nous pouvons utiliser des limites pour fournir les limites inférieure et supérieure des variables de décision.

Nous pouvons utiliser le paramètre **method** pour définir la méthode de programmation linéaire que nous souhaitons utiliser. Il existe trois options :

- **method="interior-point"** sélectionne la méthode du point intérieur. Cette option est définie par défaut.
- **method="revised simplex"** sélectionne la méthode révisée du simplexe en deux phases.
- **method="simplex"** sélectionne l'ancienne méthode du simplexe en deux phases.

linprog() renvoie une structure de données avec ces attributs :

- .con correspond aux résidus des contraintes d'égalité.
- .fun est la valeur de la fonction objectif à l'optimum (si trouvée).
- .message est le statut de la solution.
- .nit est le nombre d'itérations nécessaires pour terminer le calcul.
- .slack correspond aux valeurs des variables slack, ou aux différences entre les valeurs des côtés gauche et droit des contraintes.

- .status est un entier compris entre 0 et 4 qui indique l'état de la solution, tel que 0 lorsque la solution optimale a été trouvée.
- .success est un booléen qui indique si la solution optimale a été trouvée.
- .x est un tableau NumPy contenant les valeurs optimales des variables de décision.

Nous pouvons accéder à ces valeurs séparément :

- opt.fun
- opt.success
- opt.x

C'est ainsi que nous obtenons les résultats de l'optimisation.

Les capacités de programmation linéaire de SciPy sont principalement utiles pour les petits problèmes. Pour des problèmes plus importants et plus complexes, nous pourrions trouver d'autres bibliothèques plus adaptées pour les raisons suivantes :

- SciPy ne peut pas exécuter divers solveurs externes.
- SciPy ne peut pas fonctionner avec des variables de décision entières.
- SciPy ne fournit pas de classes ou de fonctions facilitant la création de modèles. Nous devons définir des tableaux et des matrices, ce qui peut s'avérer une tâche fastidieuse et sujette aux erreurs en cas de problèmes importants.
- SciPy ne nous permet pas de définir directement les problèmes de maximisation. Nous devons les convertir en problèmes de minimisation.
- SciPy ne nous permet pas de définir des contraintes en utilisant directement le signe supérieur ou égal à. Nous devons plutôt utiliser la valeur inférieure ou égale à.

Heureusement, l'écosystème Python propose plusieurs solutions alternatives pour la programmation linéaire qui sont très utiles pour des problèmes plus importants. L'un d'eux est PuLP, que nous verrons en action dans la section suivante.

Utiliser PuLP

PuLP dispose d'une API de programmation linéaire plus pratique que SciPy. Nous n'avons pas besoin de modifier mathématiquement notre problème ni d'utiliser des vecteurs et des matrices. Tout est plus propre et moins sujet aux erreurs.

Comme d'habitude, nous commençons par importer ce dont nous avons besoin :

- from pulp import LpMaximize/LpMinimize, LpProblem, LpStatus, lpSum, LpVariable

Maintenant que PuLP est importé, nous pouvons résoudre notre problème.

Exemple.

On considère le problème de programmation linéaire suivant :

$$\begin{cases} Max \;\; Z = 3x_1 + 2x_2 \\ 2x_1 + x_2 \leq 18 \\ 2x_1 + 3x_2 \geq 42 \\ 3x_1 + x_2 \leq 24 \\ -x_1 + 5x_2 = 15 \\ x_1, x_2 \geq 0 \end{cases}$$

La première étape consiste à initialiser une instance de LpProblem pour représenter votre modèle :

- **installer la bibliothèque pulp.** !pip install pulp
- **Importer le package pulp.**

  from pulp import LpMaximize, LpProblem, LpVariable
- **Création du problème**

prob = LpProblem(name="*Programmation_Linaire*", sense=LpMaximize)

Nous utilisons le paramètre sense pour choisir d'effectuer une minimisation (LpMinimize ou 1, qui est la valeur par défaut) ou une maximisation (LpMaximize ou -1). Ce choix affectera le résultat de notre problème.

- **Initialiser les variables de décision.** Une fois que nous avons le modèle, nous pouvons définir les variables de décision comme instances de la classe LpVariable :
  - x_1 = LpVariable(name="x1", lowBound=0)
  - x_2 = LpVariable(name="x2", lowBound=0)

Nous devons fournir une limite inférieure avec lowBound=0 car la valeur par défaut est l'infini négatif. Le paramètre upBound définit la limite supérieure, mais nous pouvons l'omettre ici car sa valeur par défaut est l'infini positif.

- **Fonction économique.** prob += 3*x1 + 2*x2, "Z"

- **Définir les contraintes du problème**

  prob += 2 x1 + x2 <= 18, "contrainte1"

  prob += 2 x1+ 3 x2 <= 42, "contrainte2"

  prob += 3 x1 + x2 <= 24, "contrainte3"

  prob += -x1 + 5x2 = 15, "contrainte4"
- **Résolution du problème.** prob.solve()
- **Affichage des résultats.**
  - print("Status : la solution optimale est donnée par :", prob.status)
  - print("La valeur de la variable x1=", round(x1.value(), 2))
  - print("La valeur de la variable x2=", round(x2.value(), 2))
  - print("la valeur optimale z=", round(prob.objective.value(), 2))

Conclusion

Nous savons maintenant ce qu'est la PL et comment utiliser Python pour résoudre des problèmes de programmation linéaire. Nous avons également appris que les bibliothèques de programmation linéaire Python ne sont que des wrappers autour de solveurs natifs. Lorsque le solveur termine son travail, le wrapper renvoie l'état de la solution, les valeurs des variables de décision, les variables de marge, la fonction objectif, etc.

Dans cette séquence, nous avons appris à :

- Définir un modèle qui représente votre problème
- Créer un programme Python pour l'optimisation
- Exécuter le programme d'optimisation pour trouver la solution au problème.
- Récupérer le résultat de l'optimisation

Nous avons utilisé SciPy avec son propre solveur ainsi que PuLP avec CBC et GLPK, mais nous avons également appris qu'il existe de nombreux autres solveurs de programmation linéaire et wrappers Python. Nous sommes maintenant prêt à plonger dans le monde de la programmation linéaire !

A) Qu'est-ce qu'un vs code ?

- Qu'est-ce qu'un fichier .py ?*
  - *Extension .py* : Indique que le fichier contient du code Python.
  - Exécution : Le code est interprété par Python (ex : python *mon_script.py*).
- VS Code : Installer l'extension Python, cliquer sur "Run Python File".

B) Qu'est-ce qu'un Notebook Jupyter ?

- Outil interactif : Mélange code, texte, images, graphiques dans un même document.
- Utilisation : Analyse de données, prototypage, enseignement, rapport interactif.
- Lancé via : Serveur web local (ex : jupyter notebook).
- *Extension .ipynb*
- Data Science : Nettoyage, visualisation (Pandas, Matplotlib).
- Enseignement : Cours interactifs (maths, programmation).

Projet

Mise en oeuvre : méthode des dictionnaires

Ouvrages de référence

- V. Chvatal - Linear Programming, W.H.Freeman, New York, 1983.
- R. J. Vanderbei - Linear Programming, Foundations and Extensions, Springer-Verlag, 2008.
- C. Guéret, C. Prins et M. Sevaux-Programmation linéaire : 65 problèmes d'optimisation modélisés et résolus avec Visual Xpress, Eyrolles, 2000.
- C. Prins et M. Sevaux - Programmation linéaire avec Excel : 55 problèmes d'optimisation modélisés pas à pas et résolus avec Excel, Eyrolles, 2011.
- Vincent ISOZ. Recherche opérationnelle (solveurs). version 4.0 Révision 10 du 2014-11-25.
- Joël M. ZINSALO. Recherche Opérationnelle. Université d'Abomey Calavi, École Polytechnique d'Abomey Calavi.