Sciences orientation logicielle 2

Mathématiques Discrètes

Chapitre 2

Optimisation Multi-Critère

Problème d’optimisation

Un problème d’optimisation consiste à trouver la meilleure solution possible à un problème selon une mesure qualitative.

Nous appellerons :

- ***Problème*** le problème à résoudre
- ***Solution*** est une solution admissible au problème
- ***Fonction objectif*** la mesure qualitative d’une solution
- ***Solution optimale*** une solution telle qu’il n’exsiste aucune solution avec une «meilleure» valeur de la fonction objectif.

Notations

- P est le problème en question,

- s est une solution du problème d'optimisation et S est l'ensemble de toutes les solutions «admissibles» (légales) au problème P,

- f dénote la fonction objectif et $f(s)$ est la valeur «qualitative» d'une solution,

- La solution optimale au problème P est écrite comme $s^* \in S$ et vérifie

$$f(s^*) = \min_{s \in S} f(s)\,.$$

Minimisation ou maximisation

Notez que tout problème de MAXIMISATION peut s'écrire sous forme de minimisation :

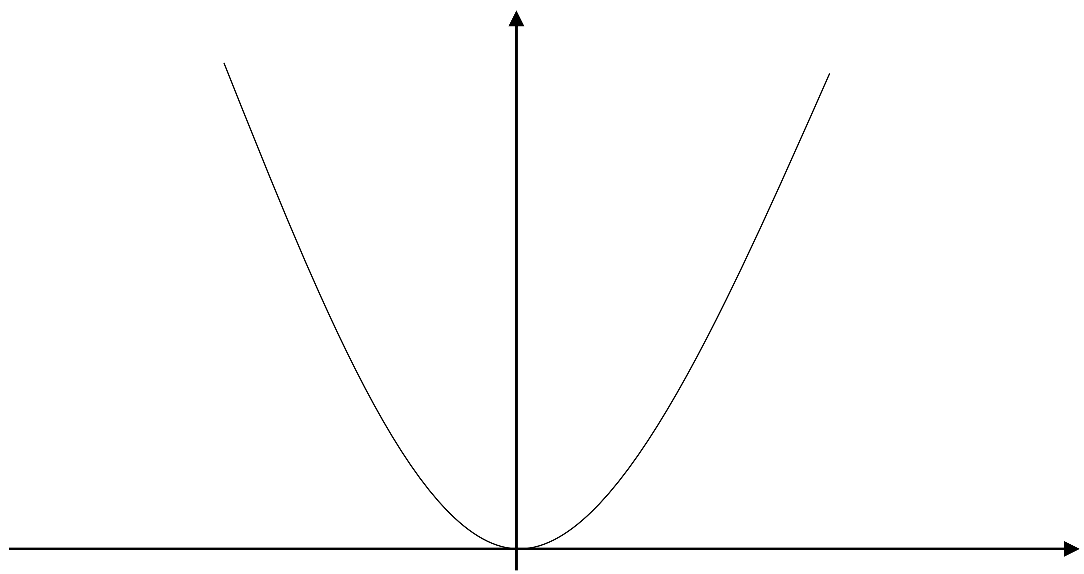
$$\max_{s \in S} f(s) = \min_{s \in S}(-f(s)) .$$

Sans perte de généralité, nous parlerons donc toujours de problèmes de minimisation (sauf précision).

Exemples

Minimiser la fonction suivante $f(s) = s^2$: $\min_{s \in \mathbb{R}} f(s)$

Nous pouvons résoudre ce problème graphiquement :



La solution optimale est $s^* = 0$ avec $f(s^*) = 0$.

Exemples

Minimiser la fonction suivante $f(s) = \frac{s}{2}$.

Dans ce cas, $\min_{s \in \mathbb{R}} f(s)$ n'est pas défini, en effet la solution serait $s^* = -\infty$!

Dans ce cas, nous pouvons en revanche ajouter des «contraintes» à notre problème, comme par exemple

$$\min_{s \in [-5, +\infty]} f(s) \, .$$

Dans ce cas, la solution est $s^* = -5$ et $f(s^*) = -2.5$!

Optimisation – solution optimale

Une solution optimale s^* satisfait

$$f(s^*) \leq f(s), \forall s \in S$$

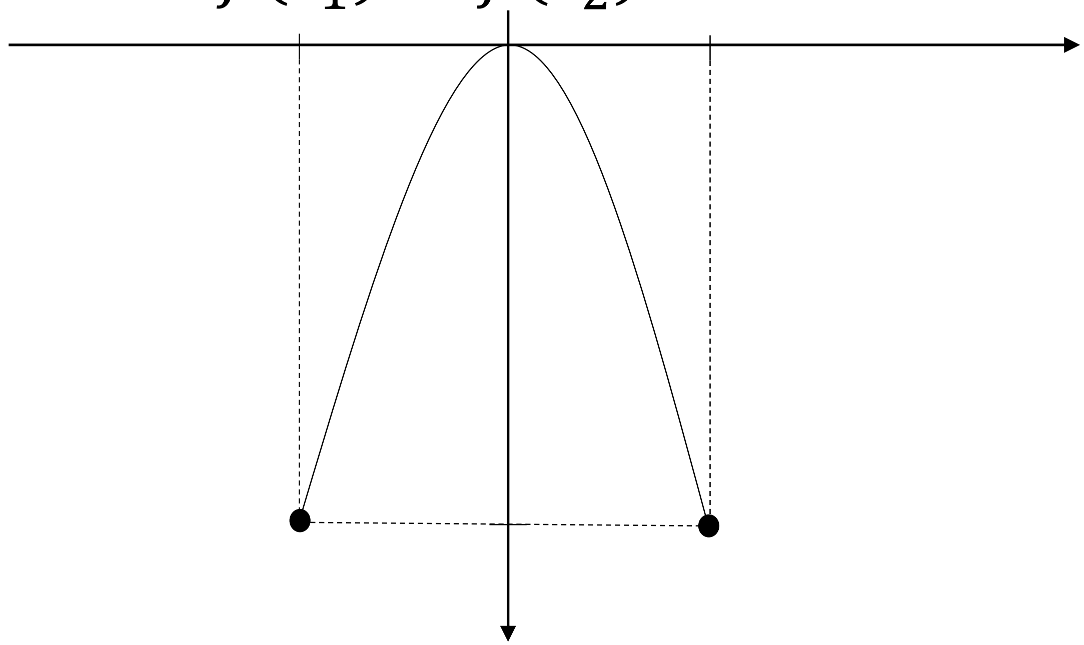
En revanche, il se peut parfaitement que s^* ne soit pas unique.

Exemple : $\min_{s \in [-5,5]} -s^2$

Optimisation – solution optimale

Exemple : $\min\limits_{s\in[-5,5]} -s^2$ a 2 solutions optimales, $s_1^* = -5$ et $s_2^* = +5$ avec

$$f(s_1^*) = f(s_2^*) = -25$$



Optimisation - Modélisation

Les problèmes d'optimisation doivent être ***modélisés*** sous forme mathématique.

Il faut donc :

1. Déterminer les ***variables de décision***,
2. Déterminer la ***fonction objectif***,
3. Déterminer les contraintes sur les variables.

Exemple

Soit le problème d'optimisation de portefeuille suivant :

Soit un nombre d'actions $i = 1, \dots, n$ dont le prix d'achat est de $p_i, i = 1, \dots, n$ et dont le profit estimé est de $\rho_i, i = 1, \dots, n$.

Nous disposons d'un budget maximum de B unités à investir.

Comment modéliser ce problème sous forme mathématique ?

Exemple – optimisation de portefeuille

<u>**Variables de décision :**</u>
$x_i =$ nombre d'actions achetées pour l'action $i = 1, \dots, n$

<u>**Fonction objectif :**</u>
L'objectif est de maximiser le profit. Pour chaque action, le profit est donné par $\rho_i \times x_i, i = 1, \dots, n.$

Cela donne donc la fonction suivante (à maximiser)

$$\sum_{i=1}^{n} \rho_i \times x_i \,.$$

Exemple – optimisation de portefeuille

Contraintes :

Nous avons un budget fixe. Le prix total d'achat de chaque action est de $p_i \times x_i, i = 1, \dots, n.$

Nous avons donc que

$$\sum_{i=1}^{n} p_i \times x_i \leq B.$$

Exemple – optimisation de portefeuille

Problème d'optimisation

Le problème d'optimisation est donc

$$\min\left(-\sum_{i=1}^{n} \rho_i \times x_i\right)$$

$$\textit{Sous contraintes :}$$

$$\sum_{i=1}^{n} p_i \times x_i \leq B.$$

$$x_i \geq 0, \forall i = 1, \dots, n.$$

Exemple – optimisation de portefeuille

Il s'agit donc d'un problème de maximisation avec n variables et une seule contrainte.

Comment résoudre ce problème ?

Exemple – optimisation de portefeuille

La solution optimale est de prendre le plus d'actions possibles de l'action ayant le meilleur rapport qualité-prix. Il faut donc trouver une action $i^* \in \{1, \dots, n\}$ telle que

$$\frac{\rho_{i^*}}{p_{i^*}} = \min_{i=1,\dots,n} \left(\frac{\rho_i}{p_i}\right).$$

Notez que i^* n'est pas forcément unique…

Une solution optimale sera alors

$$\begin{cases} x_{i^*} = B/p_{i^*} \\ x_i = 0, \forall i \in \{1, \dots, n\} \setminus \{i^*\}. \end{cases}$$

La fonction objectif aura pour valeur $\rho_{i^*} \times x_{i^*}$.

Exercice numérique

Trouvez un portefeuille optimal avec les actions suivantes et un budget de $B = 1000$:

$\boldsymbol{i}$	$\boldsymbol{\rho_i}$	$\boldsymbol{p_i}$
1	115	75
2	103	40
3	206	90
4	32	38
5	115	74
6	74	35
7	154.5	60

Exercice numérique

Résolution – Calculons les rapports $\frac{\rho_i}{p_i}$

$\boldsymbol{i}$	$\boldsymbol{\rho_i}$	$\boldsymbol{p_i}$	$\boldsymbol{\rho_i / p_i}$
1	115	75	1.533
2	103	40	**2.575**
3	206	90	2.289
4	32	38	0.842
5	115	74	1.554
6	74	35	2.114
7	154.5	60	**2.575**

Exercice numérique

Il y a donc 2 actions «optimales», 2 et 7, avec retour sur investissement optimal

$$\frac{\rho_2}{p_2} = \frac{\rho_7}{p_7} = 2.575$$

Un solution optimale sera alors
$x_2^* = \frac{1000}{p_2} = 25.00$ et $x_1^* = x_3^* = x_4^* = \cdots = x_7^* = 0$.
Une autre :
$x_7^* = \frac{1000}{p_7} = 16.67$ et $x_1^* = x_2^* = x_3^* = \cdots = x_6^* = 0$.

Sa «valeur» est la valeur de la fonction objectif, qui dans ce cas vaut 2575 dans les deux cas.

Exercice numérique

Dans notes cas, il y a en fait une infinité de solutions car nous pouvons allouer une part du budget à l'action 2 et le reste à l'action 2, nous aurons toujours le même valeur de la fonction objectif (2575).

En fait, les solutions optimales sont la droite

$$\frac{x_2}{p_2} + \frac{x_7}{p_7} = 1000.$$

Optimisation en nombres entiers

Nous avons supposé que nous pouvions acheter un nombre quelconque d'actions de chaque action. En particulier, une fraction.

Que se passe-t-il si nous imposons devoir acheter un nombre ENTIER d'actions ?

Cela revient à transformer la contrainte

$$x_i \geq 0, \forall i = 1, \dots, n$$

en une nouvelle contrainte :

$$x_i \in \mathbb{N}, \forall i = 1, \dots, n.$$

Optimisation en nombres entiers

Quelle en est la conséquence sur les solutions optimales ?

Etant donné que la solution
$x_2^* = \frac{1000}{p_2} = 25.00$ et $x_1^* = x_3^* = x_4^* = \cdots = x_7^* = 0$

satisfait la contrainte des nombres entiers, c'est aussi une solution optimale du second problème.

Optimisation en nombres entiers

Notons que toute solution du second problème ($x_i \geq \mathbf{0}$) est également solution du premier.

Nous appelons le premier problème la ***relaxation*** du problème en nombre entier (car nous «relâchons» la contrainte des nombres entiers).

Optimisation en nombres entiers

Toute solution du problème initial est solution du problème relaxé.

Par conséquent

$$\min_{s \in S_{relax}} f(s) \leq \min_{s \in S_{\mathbb{N}}} f(s).$$

Cela signifie que la solution optimale du problème relaxé est une ***borne inférieure*** du problème initial.

Algorithme vs heuristique

Il est souvent plus facile de créer une solution à un problème qu'en prouver l'optimalité.

Un ***algorithme d'optimisation*** est une méthode dite «exacte», dans le sens qu'il fournit une solution qui est prouvée être optimale.

Une ***heuristique***, elle, permet de trouver une «bonne solutions», mais sans pour autant prouver que la solution est optimale.

Algorithme vs heuristique

La raison de l’existence des heuristiques est que dans la grande majorité des problèmes d’optimisation sont difficiles (NP-complet voir même non-polynomiaux).

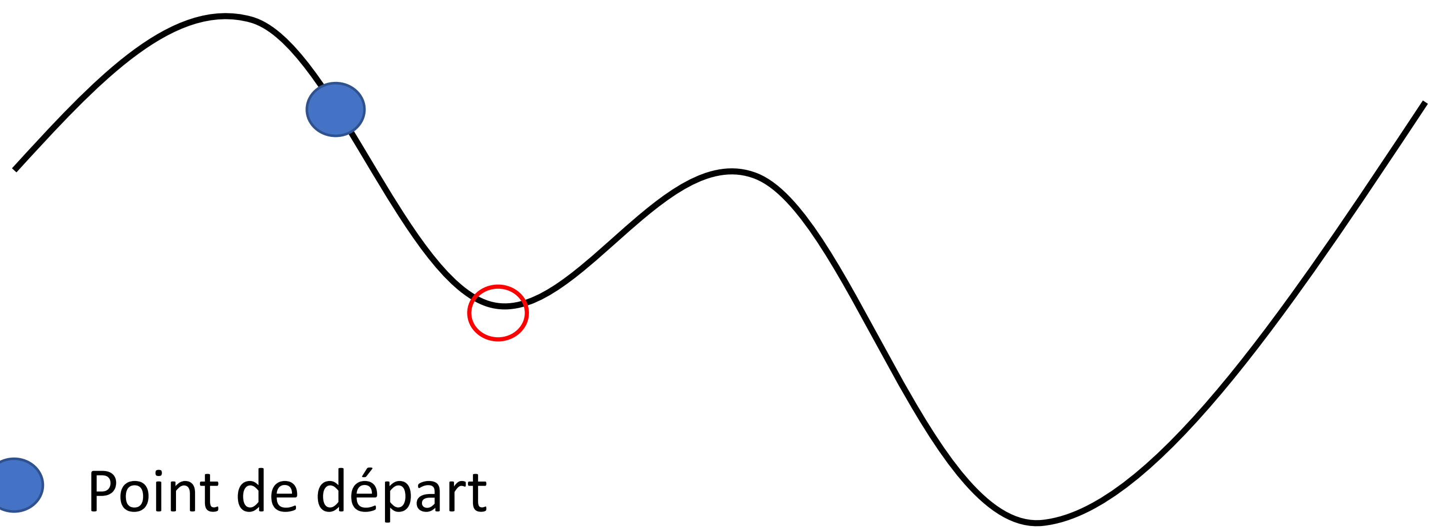
De plus, les algorithmes d’optimisation passent souvent la majorité de leur temps à prouver qu’une solution trouvée est optimale.

Il n’existe pas d’algorithme universel d’optimisation – l’énumération en serait – or nous avons vu avec la théorie de Turing que pas tous les problèmes sont décidables.

Algorithme vs heuristique - exemple

Méthode de la plus forte pente : toujours descendre dans la directions de la plus forte pente.

Résultat : un minimum ***LOCAL*** du problème.

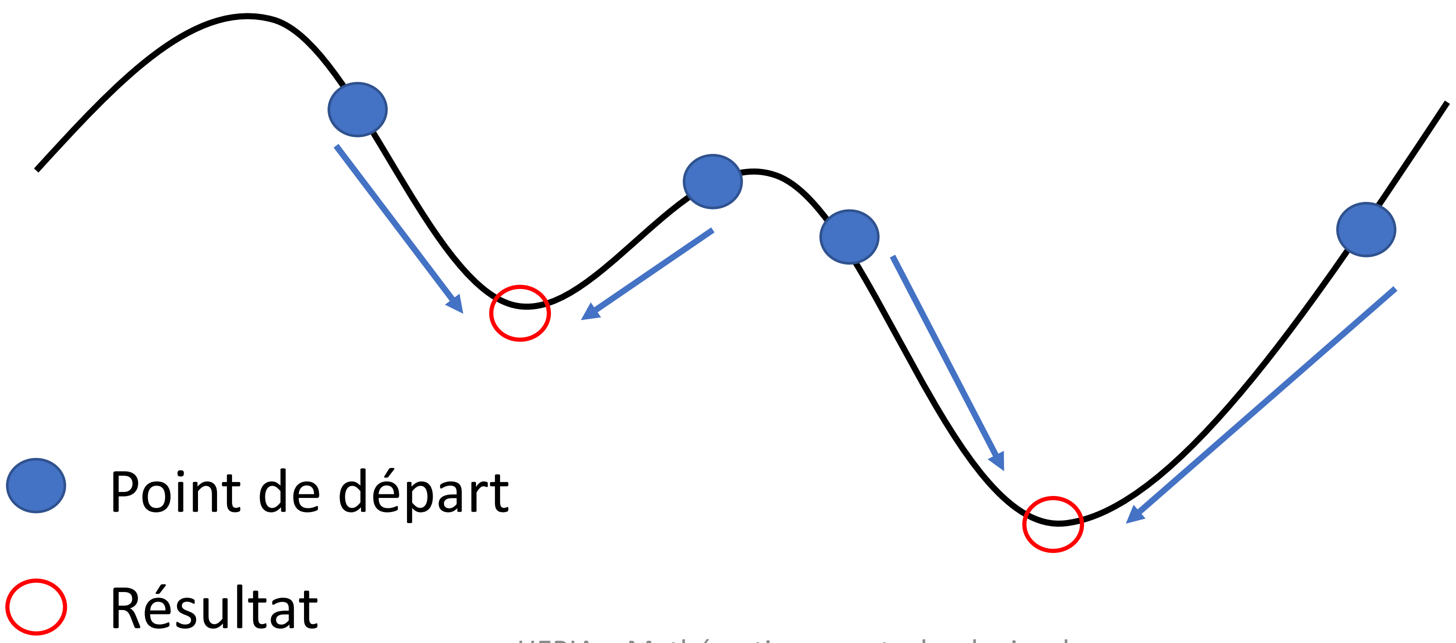


Point de départ

Résultat

Algorithme vs heuristique - exemple

La méthode de la plus forte pente donne toujours un minimum local.
Or, cela ne donne aucune garantie sur l'optimum global.
La méthode dépend fortement du point de départ...



Algorithme vs heuristique - bornes

Reprenons le problème de l'optimisation du portefeuille en nombres entiers.

Ce problème est «facile» à résoudre sous sa forme relaxée (nombres réels). De plus, nous savons que sa solution optimale est une borne inférieur au problème en nombre entiers :

$$f^* = \min_{s \in S_{relax}} f(s) \leq \min_{s \in S_{\mathbb{N}}} f(s) = F^*$$

Algorithme vs heuristique - bornes

Par conséquent, toute heuristique pour trouver un portefeuille en nombres entiers peut utiliser f^*, la borne inférieure, pour estimer l'écart à l'optimum «au pire des cas».

Soit F une trouvée admissible du problème en nombres entiers trouvé par une heuristique, nous savons que l'écart à l'optimum est au plus de $F - f^*$.

Algorithme vs heuristique – borne d'optimalité

Lorsque nous connaissons une borne inférieure à un problème, nous pourrons alors définir la ***borne d'optimalité*** (*optimality gap* en anglais) comme le ratio d'écart à l'optimum :

$$\omega = \frac{F - f^*}{f^*}$$

Si on multiple par 100, ω représente le pourcentage de perte par rapport à la borne inférieur.

Exercice

Prouvez que si $\omega = 0$, alors la solution F est prouvée optimale.

Optimisation multicritères

Reprenons l'exemple de l'optimisation de portefeuille.

L'exemple simpliste suppose que le profit d'une action est déterministe – ce qui n'est clairement pas le cas.

Supposons donc que nous disposions d'une mesure du «risque» associé à chaque action :

$$r_i, \forall i = 1, \dots n$$

Nous modélisons le risque d'un portefeuille correspond à :

$$R = \sum_{i=1}^{n} r_i \times x_i \,.$$

Exercice numérique

Trouvez un portefeuille maximisant le profit et minimisant le risque avec les actions suivantes et un budget de $B = 1000$:

$\boldsymbol{i}$	$\boldsymbol{\rho_i}$	$\boldsymbol{p_i}$	$\boldsymbol{r_i}$
1	115	75	104
2	103	40	58
3	206	90	214
4	32	38	16
5	115	74	111
6	74	35	36
7	154.5	60	123

Exercice numérique

Quelle est la solution optimale minimisant le risque :

$$R^* = \min \left(\sum_{i=1}^{n} r_i \times x_i \right)$$

Sous contraintes :

$$\sum_{i=1}^{n} p_i \times x_i \leq B.$$

$$x_i \geq 0, \forall i = 1, \dots, n.$$

Réponse

La $x_i^* = 0, \forall i = 1, \dots, n$ donne un risque $R^* = 0$…

Moralité : pas d’investissement, pas de risque !

Qu’en est-il si nous devons investir la totalité du budget ?

Dans ce cas, il s’agit de prendre l’action avec le plus faible rapport risque/coût – donc l’action 4.

La solution optimale sera alors

$$x_i^* = 0, \forall i \neq 4, x_4^* = 1000 * \frac{1}{38} = 26.316$$

Le risques associé est

$$R^* = 421.053.$$

Qu'en est-il du profit ?

La solution minimisant le risque a un profit de 842.11…
C'est (sans surprise) le plus faible profit !

Nous avions 2 portefeuilles optimaux maximisant le revenu :
$x_2^* = \frac{1000}{p_2} = 25.00$ et $x_1^* = x_3^* = x_4^* = \cdots = x_7^* = 0$,
et
$x_7^* = \frac{1000}{p_7} = 16.66$ et $x_1^* = x_2^* = x_3^* = \cdots = x_6^* = 0$.

Leurs risques respectifs sont 1450 et 2050, respectivement.

Optimisation multicritères

L'exemple de l'optimisation de portefeuille «risqué» est un exemple classique d'un problème d'optimisation multicritères avec 2 objectifs conflictuels.

Dans le cas général, il peut y avoir en avoir plus (disons m). Notons que chaque objectif à optimiser sera alors noté comme

$$f_j(s), j = 1, \ldots, m.$$

Optimisation multicritères

Le problème d’optimisation multicritères devient alors

$$\min_{s \in \mathrm{S}} \{f_{1(s)}, \ldots, f_m(s)\}.$$

Quelles sont les approches possibles ?

1. Se concentrer sur un seul objectif – cela donne des bornes pour chaque objectif individuel (inférieurs s'il s'agit de les minimiser, supérieurs s'il s'agit de les maximiser) ;

2. Optimisation par contrainte : se concentrer sur un seul objectif, et ajouter des contraintes sur la qualité des autres ;

3. Optimiser une combinaison de divers objectifs (typiquement, linéaire) ;

4. Combinaison des méthodes précédentes , typiquement itératives ;

Principe de Pareto-optimalité

Une solution $s^* \in S$ est dite Pareto-optimale s'il n'existe aucune solution $s' \in S$ telle que

$$f_j(s') \leq f_j(s^*), j = 1, \ldots, m$$

Avec au moins un indice $k \in \{1, \ldots, m\}$ tel que

$$f_k(s') < f_k(s^*).$$

Principe de Pareto-optimalité

Propriétés :
1) Si $s^* \in S$ est une solution Pareto-optimale et soit $s' \in S$ une solution telle que

$$f_j(s') \leq f_j(s^*), j = 1, \dots, m$$

Alors

$$f_j(s') = f_j(s^*), j = 1, \dots, m$$

Autrement dit, la solution est «équivalente» et donc, elle-aussi, Pareto-optimale.

Principe de Pareto-optimalité

Propriétés :
2) Si $s^* \in S$ est une solution Pareto-optimale. Soit une solution $s' \in S$ une solution telle que

$$f_k(s') < f_k(s^*), k \in \{1, \dots, m\},$$

Alors il existe (au moins) un indice $l \in \{1, \dots, m\}$ tel que

$$f_l(s') > f_l(s^*).$$

Autrement dit, si une solution améliore (au moins) un objectif d'une solutions Pareto-optimale, alors (au moins) un autre objectif sera détérioré.

Théorème (condition suffisante)

Soit le problème d'optimisation dérivé du problème multicritères obtenu par une combinaison linéaire des objectifs à optimiser

$$\min_{s \in S} \sum_{j=1}^{n} \lambda_j f_j(s)\,,$$

avec $\lambda_j > 0, j = 1, \ldots, m$.

Alors toute solution optimale du problème ci-dessus est une solution Pareto-optimale du problème multicritère initial.

Principe de solution dominée

Soit le problème d'optimisation multicritères suivant :

$$\min_{s \in \mathrm{S}} \{f_{1(s)}, \dots, f_m(s)\},$$

et soient deux solution $s_1, s_2 \in \mathrm{S}$ telles que

$$f_j(s_1) \leq f_j(s_2), j = 1, \dots, m.$$

Nous dirons alors que s_1 ***domine*** s_2 (ou s_2 ***est dominée*** par s_1).

Si, en plus, il existe au moins un indice $k \in \{1, \dots, m\}$ tek que

$$f_k(s_1) < f_k(s_2)$$

Alors, s_1 sera toujours la solution choisie car elle est toujours aussi bien voir meilleure (selon les critères optimisés) que s_2.

Dans ce cas, nous dirons que s_1 domine ***strictement*** la solution s_2.

Domination - exemple

Reprenons l'exemple d'optimisation de portefeuille où il faut ***minimiser*** le risque et ***maximiser*** le profit.

Prenons les deux solutions optimales pour le profit :

$$s' = \{x_2' = 25,.00\ x'_1 = x'_3 = \cdots = x'_7 = 0\}$$

et

$$s'' = \{x_7'' = 16.66, x_1'' = x_2'' = \cdots = x_6'' = 0\}.$$

Rappelons les valeurs du profit et du risque :

$[\rho', r'] = [2575, 1450]$ et $[\rho'', r''] = [2575,2050]$.

Nous voyons que $[\rho', r']$ domine strictement $[\rho'', r'']$, car

$$\rho' = 2575 \geq 2575 = \rho''$$

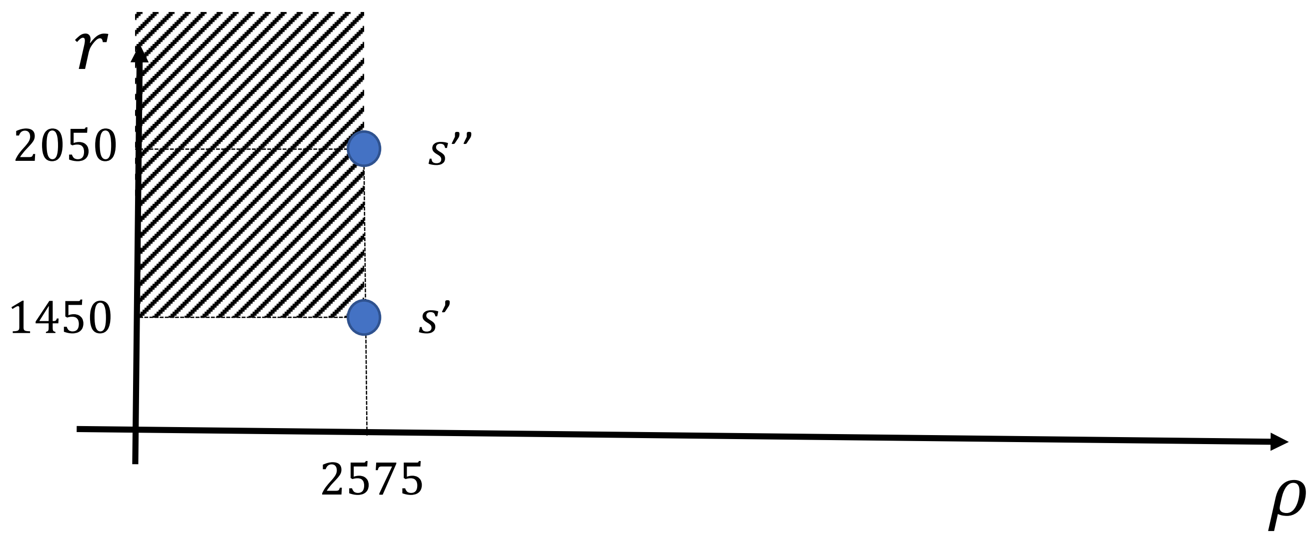
Et

$$r' = 1450 < 2050 = r''$$

NOTE: ici, nous MAXIMISONS ρ, ce qui explique l'inversion de l'inégalité dans le critère de domination pour l'objectif 1 (le profit ρ).

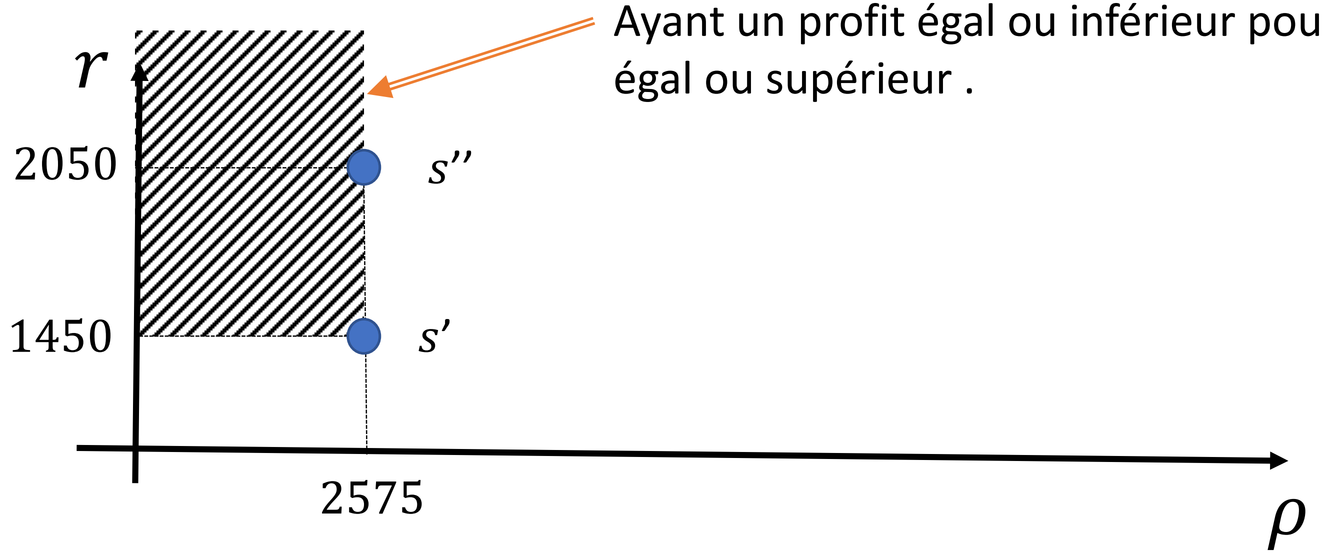
Domination – graphiquement

Si nous représentons sur l’axe X la valeur du profit ρ et sur l’axe Y la valeur du risque r d’un portefeuille, chaque solution correspond à un point unique.



Domination – graphiquement

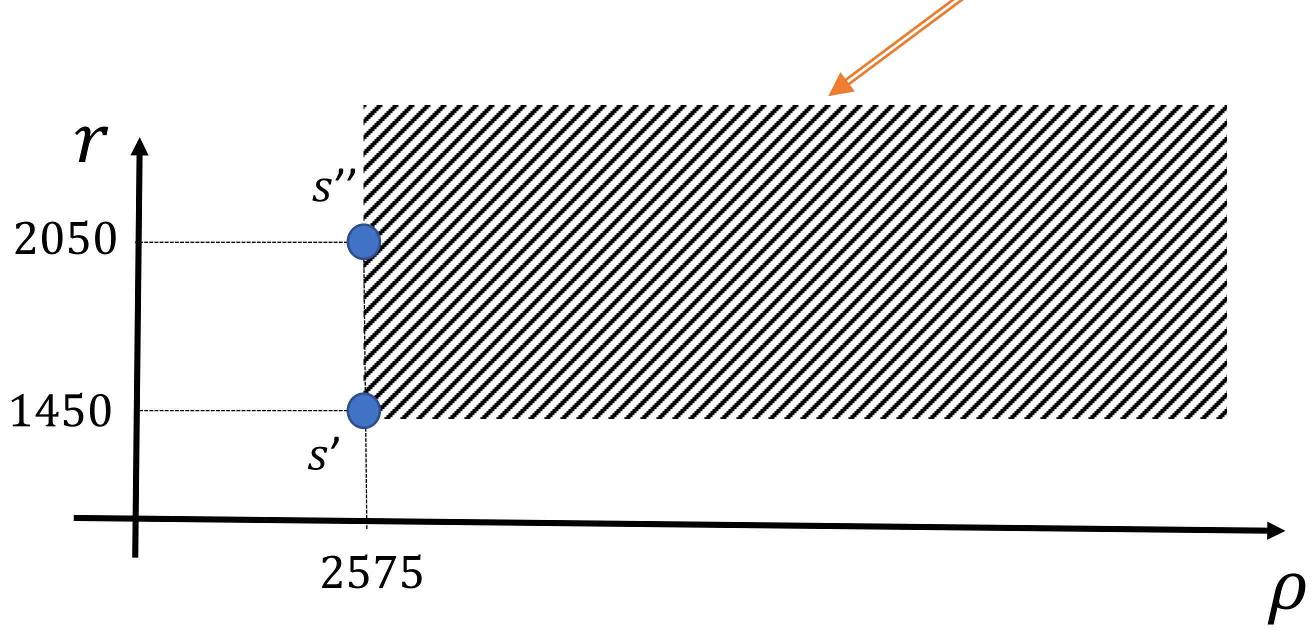
s' domine s''. En fait, tout solution dans le rectangle haché est dominée par s' (cela inclut s'') !



Domination – graphiquement

NOTE IMPORTANTE : il faut impérativement tenir compte du sens de l'optimisation (min ou max) pour définir de quel côté se situe la zone de domination !

Zone de domination s'il fallait le profit est à minimiser !



Définition – frontière de Pareto

Toute solution Pareto-optimale est une solution qui n’est strictement dominée par aucune autre solution.

L’ensemble des solution Pareto-optimales s’appelle la ***Frontière de Pareto***.

Exercice

Etudiez la frontière de Pareto du problème suivant :

$$\min_{x} \{10 + |x - 5|,\ 15 - x, 20 - 2x\}$$

s.c.

$$x \in [0, 10]$$