M2ISN, RO: Optimisation linéaire

Bernhard Beckermann
Labo Painlevé (AN-EDP), ULille
Bernhard.Beckermann@univ-lille.fr

Villeneuve d'Ascq, 12 septembre 2025

Les enjeux de la recherche opérationnelle

La création d'un modèle (relations mathématiques entre les composants d'un problème) permet d'accroître la connaissance d'un processus, d'une gestion,...

Un tel modèle fournit, le plus souvent, à l'entreprise la possibilité de réexaminer l'ensemble de ces procédures : la recherche opérationnelle peut devenir un véritable *outil d'aide à la décision*.

Une telle démarche de modélisation met souvent en question les démarches habituelles en entreprise, elle devrait donc s'appuyer sur la direction générale, et ne peut être réaliste si elle est le résultat d'un véritable travail commun avec les utilisateurs (marketing, techniciens, etc).

Les étapes dans une démarche de modélisation

1. Enquêtes auprès des techniciens du process, des gestionnaires, des décideurs, pour comprendre
   - l'enjeu du problème rencontré, les réponses attendues;
2. Mise en équation (souvent des équations continues comme les EDP provenant de la physique);
3. Discrétisation : écriture de relations par des variables et équations dans un espace "représentable", en mettant en question
   - la pertinence de la représentation d'un problème;
   - la portée des approximations prévues;
   - la réactivité aux ajustements du modèle, l'adéquation aux objectifs;
   - les contraintes de traitement numérique (coût machines/temps, stabilité numérique);
4. Retour au point 1.
5. Traduction des résultats mathématiques en langage d'utilisateur/de décideur;
6. Retour au point 1.

Problèmes discutés dans ce cours et leurs origines

1. Problèmes d'optimisation linéaire
   - en variables continues ($\in \mathbb{R}$)
   - en variables bivalentes ($\in \{0, 1\}$) dites "de décision"
   - en variables entières ($\in \mathbb{Z}$),... ou un mélange des trois.

   L'accent est mis sur la mise en équation (sous AMPL), et l'interprétation économique des résultats (evt. après variation de données).
2. Problèmes d'optimisation combinatoire (les graphes)
   - problèmes de parcours/de tournées
   - problèmes de coloration
   - problèmes d'ordonnancement

Parmi les domaines d'applications on trouve

- l'industries pétrolières ou chimiques (allocation de ressources);
- manufacturières (allocations de capacités, matières premières);
- agriculture et industries alimentaires animales;
- énergie et transport (problèmes de flot/affectation/transbordement);

...typiquement des problèmes de grande taille car étude sur des grands intervalles de temps.

Rappel cadre théorique d'OL en variables continues

On suppose donné des entiers $p, q \geqslant 0$, $n > 0$, $m = p + q > 0$, et

- $f = (f^1, ..., f^n) \in \mathbb{R}^{1\times n}$ un vecteur ligne (l'objectif)
- $\underline{x} = (\underline{x}_1, ..., \underline{x}_n)^t, \overline{x} = (\overline{x}_1, ..., \overline{x}_n)^t \in \mathbb{R}^n$ les bornes (evt. $\underline{x}_j = -\infty$, $\overline{x}_j = +\infty$)
- $B = (B_j^k) = \begin{bmatrix} B_1^1 & \cdots & B_1^n \\ \cdot & & \cdot \\ B_p^1 & \cdots & B_p^n \end{bmatrix} \in \mathbb{R}^{p\times n}, b = (b_1, ..., b_p)^t \in \mathbb{R}^p$ pour les contraintes d'égalité
- $C = (C_j^k) = \begin{bmatrix} C_1^1 & \cdots & C_1^n \\ \cdot & & \cdot \\ C_q^1 & \cdots & C_q^n \end{bmatrix} \in \mathbb{R}^{q\times n}, c = (c_1, ..., c_q)^t \in \mathbb{R}^q$ pour les contraintes d'inégalité

On se pose le problème mathématique

$$(*) \qquad \begin{cases} \min fx \\ Bx = b,\ Cx \leqslant c,\ x \geqslant \underline{x},\ x \leqslant \overline{x}. \end{cases}$$

En introduisant une variable d'écart (slack variable) $y \geqslant 0$ avec $Cx + y = c$ on peut se ramener au cas d'une *forme standard*

$$(**) \qquad \{\ \min fx,\ Ax = a,\ x \geqslant \underline{x},\ x \leqslant \overline{x}.$$

Un petit exemple

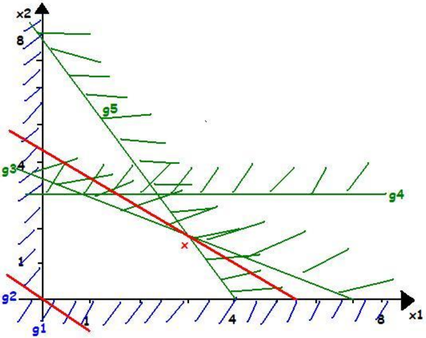
Un fabricant d'emballages envisage l'achat de machines à plier le carton de deux types différents : le modèle A et le modèle B. Le modèle A peut plier 30 boîtes par minute et doit être alimenté et contrôlé par une personne, tandis que le modèle B peut plier 50 boîtes par minute et requiert 2 personnes. Le fabricant doit mettre en forme au moins 120 boîtes par minute et ne peut pas consacrer plus de 12 employés pour l'opération de pliage. Si une machine du modèle A coûte 15000 E et une machine du modèle B coûte 20000 E, combien de machines de chaque type doit-il acheter pour minimiser son investissement ?

- Introduction des variables: $x = \begin{bmatrix} x_A \\ x_B \end{bmatrix}$, x_A (et x_B, respectivement), désignant le nombre de machines du type A (et du type B) à acheter;
- Objectif : minimiser coût total $15000x_A + 20000x_B$;
- Contraintes naturelles liées à la nature des variables: $x \geqslant 0$ (c'est-à-dire, $x_A \geqslant 0$, $x_B \geqslant 0$);
- Contrainte production par minute : $30x_A + 50x_B \geqslant 120$
- Contrainte ressources humaines : $x_A + 2x_B \leqslant 12$.
- Formulation matricielle (forme (*)): $f = [15000, 20000]$,

$$C = \begin{bmatrix} -30 & -50 \\ 1 & 2 \end{bmatrix}, \quad c = \begin{bmatrix} -120 \\ 12 \end{bmatrix}, \quad B = b = \emptyset, \quad \underline{x} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \overline{x} = \begin{bmatrix} +\infty \\ +\infty \end{bmatrix}.$$

Le théorème principal

Nous devons donc minimiser une forme linéaire sur un polyèdre (=intersection fini de demi-espaces):



Théorème : si la forme linéaire est bornée inférieurement sur le polyèdre, alors une solution optimale est atteinte en un point extrême du polyèdre.

En conséquence, l'optimisation linéaire consiste à rechercher un meilleur point dans un ensemble discret (paramétré à l'aide de l'algèbre linéaire).

Points de base et arêtes

Pour le problème $(**)$ on définit

- **Base :** Toute partition (I, B^+, B^-) de l'ensemble d'indices colonnes de A avec $A^I = (A^j)^{j\in I}$ inversible
- **Point de base :** $x = x(I, B^+, B^-)$ solution de $Ax = a$ avec $x_{B^+} = (x_j)_{j\in B^+} = \overline{x}_{B^+}$ et $x_{B^-} = \underline{x}_{B^-}$.
- **Point de base réalisable :** Point de base ou de plus $\underline{x}_I \leqslant x_I$ et $x_I \leqslant \overline{x}_I$.

On peut montrer que l'ensemble des points de base réalisables co'ncide avec l'ensemble des points extrêmes (les points "intéressants" pour l'optimisation linéaire).

La méthode Simplex

Le principe de Simplex, à partir d'un point extrême $x(I, B^-, B^+)$, est alors de rechercher un point extrême adjacent (c'est-à-dire, relié à $x(I, B^-, B^+)$ par une arête de dimension 1) et meilleur pour l'objectif. L'équation d'une telle arête est donnée par ($s \notin I$)

$$x(\theta) = x(I, B^-, B^+) + \theta c(s), \quad \theta \in \mathbb{R}, \quad c(s)_I = -(A^I)^{-1} A^s, \quad c(s)_k = \delta_{s,k} \text{ pour } k \notin I.$$

On obtient un nouveau point de base réalisable pour une base dite adjacente, i.e., on échange au plus deux indices, en prenant $x(\theta_{\max})$ où la valeur critique $\theta = \theta_{\max}$ est déterminée par le système d'inégalités $x(\theta) \geqslant \underline{x}$, $x(\theta) \leqslant \overline{x}$. L'arête d'indice s est choisie de sorte que l'objectif diminue : $fc(s) < 0$ si $\theta_{\max} \geqslant 0$, et $fc(s) > 0$ si $\theta_{\max} \leqslant 0$.

- **Coûts réduits :** $d(I) = f - f^I (A^I)^{-1} A$, pour $j \notin I : d(I)^j = fc(j)$
- **C.S.O :** $x(I, B^+, B^-)$ est solution optimale si $d(I)^{B^-} \geqslant 0$, $d(I)^{B^+} \leqslant 0$.

Par conséquent, si la CSO (condition d'arrêt) n'est pas encore valable, on trouve $s \in B^-$ avec $d(I)^s < 0$ ou $s \in B^+$ avec $d(I)^s > 0$, et

$$\widehat{x} = x(I, B^-, B^+) + \theta_{\max} c(s)$$

est notre nouveau point de base réalisable (si on passe le test que le polyèdre soit vide). De plus $f\widehat{x} \leqslant fx(I, B^+, B^-)$, avec généralement inégalité stricte (sauf dégénérescence). $\Longrightarrow$ *convergence finie* (sauf si cyclage)

La mise en oeuvre de SIMPLEX se fait souvent à l'aide d'une décomposition LU (en creux) de A^I, une itération consiste alors de résoudre trois systèmes avec matrice A^I, et de mettre à jour cette décomposition (complexité $\leqslant \mathcal{O}(m^2)$ pour une itération). Ainsi (et avec quelques précautions pour la stabilité numérique), la version révisée de Simplex permet de résoudre des problèmes de grande taille ($\approx$ 50000 inconnues). De plus, l'interprétation économique de certaines quantités auxiliaires calculés par SIMPLEX le rend compétitif avec des méthodes de points intérieur plus modernes.

Il existe des variantes Simplex dual (on construit une suite de bases vérifiant la CSO, mais seulement la dernière vérifiant les contraintes d'inégalités) et simplex primal-dual (permettant de donner un encadrement de la valeur optimale, et donc une condition d'arrêt alternative).

Exercice 1

Une huile de consommation est fabriquée par le raffinage d'huiles brutes : 2 huiles végétales (VG1 et VG2) et 3 non végétales (01,02,03).
Le produit final est un mélange des 5 huiles brutes et est vendu 150E/tonne. Il doit satisfaire à un indice de dureté (entre 3 et 6) qui est une fonction linéaire des duretés des huiles brutes (par exemple, mélanger 1 tonne d'une huile avec indice 4 et 2 tonnes d'une huile avec indice 7 donne une huile finale avec indice $6 = (1 * 4 + 2 * 7)/3$). Les duretés des matières premières sont données dans la table suivante

huile	$VG1$	$VG2$	$O1$	$O2$	$O3$
dureté	8.8	6.1	2.0	4.2	5.0

La chaîne de raffinage peut traiter chaque mois pas plus que 200 tonnes d'huile végétal et 250 tonnes de non végétal. Il n'y a pas de perte de poids dans le process de raffinage et le coût du raffinage est négligeable.
Il est possible de stocker chaque matière première (dans la limite de 1000 tonnes) à un coût de 5E par tonne et par mois. Le produit final ne peut pas être stocké, ainsi que les huiles raffinées avant mélange.

La gestion comprend 6 mois successives (janvier à juin). Chaque mois il est possible d'acheter les huiles brutes selon les coûts suivants (en E par tonne)

mois/huile	*VG*1	*VG*2	*O*1	*O*2	*O*3
janvier	110	120	130	110	115
février	130	130	110	90	115
mars	110	140	130	100	95
avril	120	110	120	120	125
mai	100	120	150	110	105
juin	90	100	140	80	135

1. Quelles sont les statégies d'achat et de traitement sur les 6 mois de manière à maximiser les profits? On suppose qu'au début il y a 500 tonnes de stock sur chaque huile brute et on souhaite terminer la période de gestion avec un stock équivalent.
2. Discuter l'impact d'un petit changement de dureté/prix d'achat.
3. On souhaite simuler comment le profit, les achats et la stratégie de raffinage évoluent pour des changements de prix majeurs dans les mois à venir. On supposera que l'augmentation de prix unitaires soit, par mois et à partir du mois de février, de λ Euro sur les végétales et de 2λ Euro sur les autres. Analyser l'impact pour des valeurs différents de λ (à l'aide des commande AMPL dans le fichier `*.run` en utilisant `let`).

Exercice 2

Une entreprise dispose de deux usines : une à Liverpool et une à Brighton. Elle dispose aussi de quatre dépôts à Newcastle, Birmingham, Londres et Exeter. Elle vend des produits à six clients C1 à C6. Elle peut fournir ses clients aussi bien à partir des dépôts qu'à partir des usines. Les coûts de distribution sont connus. Ils sont donnés (en livres par tonne) dans la table ci–dessous (un blanc à l'intersection de la colonne *F* et de la ligne *C* signifie que le fournisseur *F* ne livre pas à *C*).

fournit à	usine Liverpool	usine Brighton	dépôt Newcastle	dépôt Birmingham	dépôt Londres	dépôt Exeter
Newcastle	0.5					
Birmingham	0.5	0.3				
Londres	1.0	0.5				
Exeter	0.2	0.2				
C1	1.0	2.0		1.0		
C2			1.5	0.5	1.5	
C3	1.5		0.5	0.5	2.0	0.2
C4	2.0		1.5	1.0		1.5
C5				0.5	0.5	0.5
C6	1.0		1.0		1.5	1.5

Certains clients préfèrent être livrés par les fournisseurs auxquels ils sont habitués. Voici les préférences :

C1	l’usine de Liverpool	C2	le dépôt de Newcastle
C3	pas de préférence	C4	pas de préférence
C5	le dépôt de Birmingham	C6	les dépôts d’Exeter et de Londres

Chaque usine a une capacité de production limitée :

Liverpool	150000 tonnes	Brighton	200000

Chaque dépôt a un débit mensuel limité :

Newcastle	70000 tonnes	Birmingham	50000
Londres	100000	Exeter	40000

Les clients ont des besoins mensuels à satisfaire absolument :

C1	50000 tonnes	C2	10000
C3	40000	C4	35000
C5	60000	C6	20000

1. Quel schéma de distribution minimiserait le coût total ?
2. Quel effet sur les coûts un accroissement des capacités des usines et des dépôts aurait–il ? Impact sur le schéma de distribution?
3. Est–il possible de satisfaire les préférences des clients ? Combien cela coûterait–il ?

L'entreprise a la possibilité d'ouvrir de nouveaux dépôts à Bristol et à Northampton. Elle peut aussi agrandir celui de Birmingham. Elle considère qu'il n'est pas souhaitable de maintenir plus de quatre dépôts. Elle pourrait si nécessaire fermer ceux Newcastle et d'Exeter. Voici les remboursements d'emprunts destinés à financer l'expansion de l'entreprise ainsi que les débits correspondants.

	coûts	débits
Bristol	12000	30000
Northampton	4000	25000
Birmingham (extension)	3000	20000

Les économies mensuelles occasionnées par la fermeture du dépôt de Newcastle sont de 10000 livres tandis que pour Exeter on gagnera 5000 livres. Voici les modifications à ajouter au tableau des coûts de distribution.

fournit à	usine Liverpool	usine Brighton	dépôt Bristol	dépôt Northampton
Bristol	0.6	0.4		
Northampton	0.4	0.3		
C1			1.2	
C2			0.6	0.4
C3			0.5	
C4				0.5
C5			0.3	0.6
C6			0.8	0.9

Faut–il construire de nouveaux dépôts ? Si oui, lesquels ? faut–il agrandir/fermer?

Modéliser ce problème à l'aide de variables décisionnelles (bivalentes).

Exercice 3

Un fermier souhaite planifier la gestion de sa ferme de 200 ares sur les cinq prochaines années. Il dispose d'un troupeau de 120 vaches constitué de 20 génisses et 100 laitières. Chaque génisse exige $2/3$ d'are et chaque laitière exige 1 are. Une laitière fournit en moyenne 1.1 veau par an. La moitié de ces veaux sont des mâles et sont vendus immédiatement 30 euros chacun. Les génisses restantes peuvent être vendues immédiatement 40 euros chacune ou être gardées pour devenir des laitières au bout de deux ans. Les laitières sont vendues à l'âge de dix ans 120 euros chacune. On estime à 5% la perte annuelle de génisses et à 2% la perte annuelle de laitières. Aujourd'hui, il y a 10 vaches de chaque âge dans la ferme (de zéro à neuf ans). Les génisses qui devaient être vendues immédiatement l'ont déjà été.

Le lait d'une vache fournit un revenu annuel de 370 euros. On peut loger au maximum 130 vaches dans la ferme. Au delà de ce nombre, il est nécessaire de payer des impôts de 200 euros par vache supplémentaire et par année. Chaque laitière exige 0.6 tonne de grain et 0.7 tonne de betterave par an. Le grain et les betteraves peuvent être produits à la ferme. Chaque are fournit 1.5 tonne de betterave mais seuls 80 ares permettent de cultiver le grain. Ces 80 ares sont à répartir en quatre groupes :

groupe 1	20 ares	1.1 tonnes de grain par are
groupe 2	30	0.9
groupe 3	20	0.8
groupe 4	10	0.65

Le grain peut être acheté 90 euros et vendu 75 euros la tonne. La betterave peut être achetée 70 euros et vendue 58 euros la tonne. Les temps de travail et les coûts sont les suivants :

1 génisse	10 heures par an	50 euros par an
1 laitière	42	100
1 are produisant du grain	4	15
1 are produisant de la betterave	14	10

L'entretien de la ferme coûte 4000 euros par an. On dispose de 5500 heures de travail. Toute heure supplémentaire coûte 1.2 euro. Quelle stratégie le fermier doit–il suivre pour maximiser son profit sachant que la ferme ne doit être déficitaire sur aucune année ?

Indications et extensions:

- Fixer pour chaque variable son moment de validité (début/fin de l'année).
- Faire une simulation sur 10 ans en continue (on interprétera un nombre de vaches fractionnaire comme moyenne sur plusieurs simulations). En utilisant l'interface WEB ou la version achetée, vous pouvez également faire des simulations sur des périodes plus longues.
  Dans un deuxième temps, on pourrait introduire pour le nombre de vaches deux variables, l'une réelle (où on tient compte de la mortalité) et l'autre sa partie entière (qui sera utilisée pour calculer les besoins/bénéfices)
- On tiendra pas compte du taux de mortalité pour la production annuelle de lait/veaux et les besoins de fourrure (les vaches meurent à la fin de l'année).
- Les comptes sont faites à la fin de l'année (pour les coûts et les bénéfices).
- Le résultat, change-t-il si on impose que le cheptel à la fin ne devrait pas être réduit plus que 50% ou augmenté plus que 75%?
- Le résultat, change-t-il si on impose (car le fermier doit lui-même aussi se nourrir) que chaque année les bénéfices devraient dépassent les coûts d'un montant de d Euros ($d = 19000$, $d = 19500$, $d = 20000$)?
- Supposons que le prix du lait à l'année j est à multiplier par le facteur $1 + 0.1 * (j - last(ANNEES)/2) * (j - last(ANNEES)/2)$;. Impact pour notre fermier?

Informations marginales en un point

Conditions nécessaires et suffisantes d'optimalité de Kuhn et Tucker
Soit $\widehat{x}$ réalisable pour un problème de type $(*)$. Alors il y a l'équivalence entre les deux propriétés:
(i) $\widehat{x}$ est une solution optimale de $(*)$
(ii) il existent $u \in \mathbb{R}^{1\times p}$, $v \in \mathbb{R}^{1\times q}$, $\overline{w}, \underline{w} \in \mathbb{R}^{1\times n}$ avec

$$v, \overline{w}, \underline{w} \geqslant 0, \quad uB - vC - \overline{w} + \underline{w} = f, \quad v(c - C\widehat{x}) = 0, \quad \underline{w}(\widehat{x} - \underline{x}) = 0, \quad \overline{w}(\overline{x} - \widehat{x}) = 0.$$

Les dernières 3 relations sont dites *relations d'exclusion* ou *de complémentarité*.

Corollaire : Avec $u, v, \overline{w}, \underline{w}$ comme dans (i),(ii) (dites *multiplicateurs de Kuhn et Tucker* ou *variables duales*) et aussi $\widehat{x}$, nous avons pour la valeur optimale

$$V := V(b, c, \underline{x}, \overline{x}) := f\widehat{x} = ub - vc + \underline{w}\,\underline{x} - \overline{w}\,\overline{x}$$

Corollaire : A condition que $\widehat{x} = x(I, B^+, B^-)$ et (I, B^+, B^-) reste aussi réalisable après une "petite" perturbation $\Delta b, \Delta c, \Delta \underline{x}, \Delta \overline{x}$, nous avons

$$V(b + \Delta b, c + \Delta c, \underline{x} + \Delta \underline{x}, \overline{x} + \Delta \overline{x}) - V(b, c, \underline{x}, \overline{x}) = u\Delta b - v\Delta c + \underline{w}\,\Delta \underline{x} - \overline{w}\,\Delta \overline{x}.$$

Les variables duales sont disponibles sous AMPL...
(I, B_-, B_+) réalisable et CSO $\Longrightarrow (u, -v) = f^I(A^I)^{-1}$, $\underline{w}_j = \max\{0, d(I)^j\}$,
$\overline{w}_j = \max\{0, -d(I)^j\}$.

L'exemple de notre usine sur plusieurs semaines

On a légèrement reformulé le problème de l'usine sidérurgique, ici sur *deux* semaines, avec deux produits *bands, coils*. Aussi, on a ajouté un paramètre `inv_max` désignant la quantité maximum qu'on puisse stocker d'une semaine à l'autre.

Voir steelTbis

L'objectif: minimiser la perte totale

```
minimize total_cost:
  sum {p in PROD, t in 1..T} (- revenue[p,t]*Sell[p,t] +
     prodcost[p]*Make[p,t] + invcost[p]*Inv[p,t]);
```

On obtient une valeur optimale de -248575.8571 (et donc des bénéfices de 248575.8571 euros).

Contraintes $Cx \leqslant c$ et v

```
subject to time {t in 1..T}:
   sum {p in PROD} (1/rate[p]) * Make[p,t] <= avail[t];
```

donne lieu à des informations suivantes (voir steelTbis)

```
:    time   time.slack    :=
 1    -2660       0
 2    -3160       0 ;
```

Interprétation: Les deux composantes de la variable d'écart `time.slack` $(= c - C\widehat{x})$ s'annulent. Donc la variable duale $v = [2660, 3160]$ est en dualité avec la variable d'écart. Pour trouver $v \geqslant 0$ on regarde `time`, attention au changement de signe !

Augmenter par exemple `avail[1]` par une unité (petite pour préserver la réalisibilité) donnera un changement d'objectif de $-v\Delta c = -2660$ unités (on y gagne).

Les bornes sup et inf et $\underline{w}$, $\overline{w}$

Les contraintes $x \leqslant \overline{x}$ et $x \geqslant \underline{x}$ ont été modélisées sous la forme (par exemple)

```
subject to sell_max {p in PROD, t in 1..T} : Sell[p,t]<=market[p,t];
subject to sell_min {p in PROD, t in 1..T} : Sell[p,t] >= 0;
```

La post optimisation donne lieu à des informations suivantes (voir steelTbis)

```
:          sell_min sell_min.slack:=   |  :          sell_max sell_max.slack
 bands 1      0          6000          |   bands 1      -1.7          0
 bands 2      0          6000          |   bands 2      -0.2          0
 coils 1      0           307          |   coils 1         0       3693
 coils 2      0          2500  ;       |   coils 2 -1.42857          0  ;
```

Interprétation: Comme on voit dans la colonne `sell_min.slack`, la partie correspondante de $\widehat{x} - \underline{x}$ n'a aucune composante zéro. Donc, par complémentarité, la partie correspondante de $\underline{w} \geqslant 0$ s'annule. Ceci est assez logique: un changement "petit" de la borne inférieure 0 ne changera pas la perte totale de l'entreprise car on produit des quantités non nuls.

Par contre, $\overline{w} = [1.7, 0.2, 0, 1.42857] \geqslant 0$ (encore attention au changement de signe). Le zéro s'explique par le fait que la restriction du marché `market[coils,1]` n'est pas atteinte. Par contre, si une autre restriction augmente, alors $-\overline{w}\Delta\overline{x}$ diminue, on y gagne.

```
var Inv {PROD,0..T} >= 0, <= inv_max;
```

Si les bornes inf et sup d'une variable apparaissent directement dans la définition comme pour `Inv`, on obtient

```
:            Inv      Inv.uslack     Inv.urc  Inv.lslack    Inv.lrc :=
 bands 0      10               0           0           0          0
 bands 1   142.857       857.143           0     142.857          0
 bands 2       0            1000           0           0       28.3
 coils 0       0               0           0           0          0
 coils 1     1000              0   -0.571429        1000          0
 coils 2       0            1000           0           0    36.5714 ;
```

Donc $\overline{w} = [0, 0, 0, 0, .571429, 0] \geqslant 0$ et $\overline{w} = [0, 0, 28.3, 0, 0, 36.5714] \geqslant 0$ (attention encore une fois aux changements de signe).

Notons que la semaine 0 est un peu particulière car les deux variables `Inv.uslack, Inv.lslack` s'annulent car on a fixé (par l'autre contrainte `initial`) la valeur. Notons aussi que les variables $\overline{w}_j$ sont non nuls seulement si `Inv` atteint la borne sup `inv_max =1000`, donc une variable d'écart `Inv.uslack` égale à zéro (voir par exemple `bands` en première semaine). En général, ceci implique que la composante correspondante de `Inv.lslack` est non nul et celle de $\underline{w}_j$ s'annule.

Exemple: si on augmente la taille de l'inventaire d'une unité, on obtient un changement d'objectif $-\overline{w}\Delta\overline{x} = -(0 + 0 + 28.3 + 0 + 0 + 36.5714) < 0$, on y gagne.

Contraintes $Bx = b$ et u

```
subject to initial {p in PROD}:  Inv[p,0] = inv0[p];
```

donne lieu à des informations suivantes (voir steelTbis)

```
initial [*] := bands  -23.3 coils  -30 ;
```

Interprétation: Ici on a une contrainte d'égalité et donc "la variable d'écart" n'a pas de sens. La variable duale vaut $u = [-23.3, -30]$. Si par exemple on aura un stock initial `inv0[bands]` d'une unité de moins (il n'y avait pas de `coils` au départ, comparer avec le fichier de données), on y perdra $u\Delta b = u[0, -1]^t = 23.3$ unités.

Exercice: donner l'interprétation en termes de pertes en stock de

```
         balance :=
bands 1    23.3
bands 2    25.8
coils 1      30
coils 2 33.5714 ;
```

La post optimisation

Ici on considère l'optimum du problème en fonction de toutes les données

$$V = V(f, b, c, \underline{x}, \overline{x})$$

(la dépendance de V des variations des matrices B, C étant plus compliqué).
Si on varie f, on ne perde pas la réalisibilité, mais pour les grandes perturbations (ou en cas de dégénérescences) on risque de perdre la C.S.O..
Conclusion: si on dispose de $\widehat{x} = x(I, B^+, B^-)$ et de $u, v, \underline{w}, \overline{w}$ comme avant alors pour un scalaire θ

$$\theta \mapsto V(f + \theta\Delta f, b, c, \underline{x}, \overline{x}) \leqslant V(f, b, c, \underline{x}, \overline{x}) + \theta\Delta f\widehat{x}.$$

En général cette fonction est concave en θ (et affine par morceaux là ou la base de change pas). En particulier, on peut obtenir inégalité stricte dans un voisinage de zéro au cas de dégénérescences.
D'une manière similaire, pour un scalaire θ

$$\begin{aligned}\theta \mapsto\ & V(f, b + \theta\Delta b, c + \theta\Delta c, \underline{x} + \theta\Delta\underline{x}, \overline{x} + \theta\Delta\overline{x}) \\ & \geqslant V(f, b, c, \underline{x}, \overline{x}) + \theta[u\Delta b - v\Delta c + \underline{w}\,\Delta\underline{x} - \overline{w}\,\Delta\overline{x}].\end{aligned}$$

En général cette fonction est convexe en θ (et affine par morceaux là ou la base de change pas). $\Longrightarrow$ *... nous obtenons des sous-gradients.*

Exemple avec impôt par pièce dans l'entrepôt

Regardons le nouvel objectif

```
minimize total_cost:
  sum{p in PROD, t in WEEKS} ( - revenue[p,t]*Sell[p,t] +
  prodcost[p]*Make[p,t] + (invcost[p] + impot_inventaire)*Inv[p,t]);
```

Ici `impot_inventaire` qui va être notre paramètre θ peut être interprété comme un impôt par pièce par semaine pour chaque objet mis en entrepôt. Logiquement, si les impôts augmentent trop, on n'utilise plus l'entrepôt. Voici la valeur optimale en fonction de θ variant entre 0 et 1.1 (on observe une courbe concave, avec trois pentes différentes).

	impot_inventaire_total_cost
0	-248576
0.05	-248519
0.1	-248462
0.15000000000000002	-248404
0.2	-248347
0.25	-248297
0.30000000000000004	-248247
0.35000000000000003	-248197
0.4	-248147
0.45	-248097
0.5	-248047
0.55	-247997
0.6000000000000001	-247947
0.65	-247897
0.7000000000000001	-247847
0.75	-247797
0.8	-247747
0.8500000000000001	-247697
0.9	-247690
0.9500000000000001	-247690
1	-247690
1.05	-247690
1.1	-247690

Pour mieux comprendre ce comportement, il faudra afficher le tableau `Inv` en semaines 1 et 2 (en semaine 0 c'est fixé).

impot_inventaire_Inv	bands,1	bands,2	coils,1	coils,2
0	142.857	0	1000	0
0.05	142.857	0	1000	0
0.1	142.857	0	1000	0
0.15000000000000002	142.857	0	1000	0
0.2	142.857	0	1000	0
0.25	0	0	1000	0
0.30000000000000004	0	0	1000	0
0.35000000000000003	0	0	1000	0
0.4	0	0	1000	0
0.45	0	0	1000	0
0.5	0	0	1000	0
0.55	0	0	1000	0
0.6000000000000001	0	0	1000	0
0.65	0	0	1000	0
0.7000000000000001	0	0	1000	0
0.75	0	0	1000	0
0.8	0	0	1000	0
0.8500000000000001	0	0	1000	0
0.9	0	0	0	0
0.9500000000000001	0	0	0	0
1	0	0	0	0
1.05	0	0	0	0
1.1	0	0	0	0

On obtient alors des bases (et donc des valeurs pour `Inv`) constants par morceaux, et l'entrepôt n'est plus utilisé à partir de `impot_inventaire` $= 0.9$ (attention au fait que l'on a tracé un polygone).

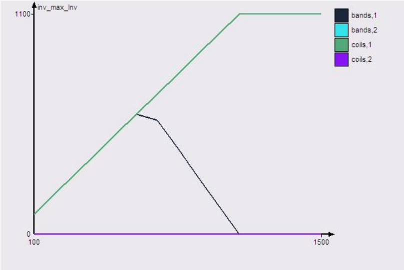
Sous monampl : après avoir appuyé sur `Make`, on clique dans le menu sur `AMPL/Make Post Optimisation loop`, dans la fenêtre on choisit `impot_inventaire`, une boucle de 0 à 1.1 par pas de 0.05, et on affiche par exemple `total_cost`. Appuyer sur `run` fait le calcul. Pour l'afficher, cliquer avec le bouton droite de la souris sur le paramètre `impot_inventaire_total_cost` et lire dans un tableau.

Exemple avec limitation des pièces en entrepôt

Varions la borne sup `inv_max` pour la variable

```
var Inv {PROD,WEEKS_WITH} >= 0, <= inv_max;
```

entre 100 et 1500 par pas de 100. Pour valeur optimale `total_cost` et la quantité à stocker `Inv` on trouve



et donc bien une courbe convexe pour la valeur optimale, avec par morceaux des points de base variant linéairement avec `total_cost`. A partir de `total_cost`= 1100, on n'utilise plus pleinement la capacité de l'entrepôt.

Programmes linéaires multi-objectifs

Etant donné un polyèdre P, par exemple $P = \{x \in \mathbb{R}^n : Ax = a, x \geqslant 0\}$, la programmation multi-objectif consiste à résoudre

$$(MP): \quad \min\{Cx : x \in P\}, \quad C \in \mathbb{R}^{\ell \times n},$$

c'est-à-dire, on veut rendre simultanément ℓ objectifs $C_j x$ le plus petit possible (ce qui est bien entendu conflictuel).

Un peu de théorie: L'ensemble $E = \{Cx : x \in P\}$ est un polyèdre, on dit que $\overline{x}$ est **optimum au sens de Pareto** si pour tout $x \in P$ avec $Cx \neq C\overline{x}$ il existe au moins un $j \in \{1, ..., \ell\}$ avec $C_j x > C_j \overline{x}$ (au moins un objectif est moins bon).

Etant donné des paramètres $\lambda^1, ..., \lambda^s > 0$, toute solution optimale de

$$(P(\lambda)): \quad \min\{\sum_{j=1}^{\ell} \lambda^j C_j x : x \in P\}$$

est optimale au sens de Pareto pour (MP). Réciproquement, l'ensemble des solutions optimales au sens de Pareto pour (MP) est la réunion sur tout paramètre λ à composantes > 0 des ensembles des solutions optimales pour $(P(\lambda))$ (mais pas forcément des points de base trouvés par SIMPLEX).

Goal programming

Ici on remplace l'objectif peu précis $\min Cx$ par des contraintes à respecter au mieux. L'utilisateur fixe $c_j \in \mathbb{R}$ et des pénalités $\pi_j > 0$ et résout par exemple

$$(GP): \quad \min\{\sum_{j=1}^{m} \lambda^j : x \in P, \forall j = 1, ..., \ell : \lambda^j \geqslant 0, C_j x - c_j \leqslant \pi_j \lambda^j\}.$$

Ici on minimise la somme des dépassements des bornes à atteindre. Les pénalités π_j traduisent, à la fois, l'importance de l'objectif j, l'intensité de la perte unitaire sur cet objectif et l'ordre de grandeur, lié aux unités de mesures, de cet objectif !
NB: toute solution optimale $\overline{x}$ de (GP) avec contraintes $C_j x - c_j \leqslant \pi_j \lambda^j$ actives est aussi optimale pour (MP) au sens de Pareto.

On peut également remplacer l'objectif $\sum_j \lambda_j$ par $\max_j \lambda_j$ pour minimiser l'écart maximum. De plus, des contrainte de la forme

$$C_j x - \widehat{c_j} \geqslant \widehat{\pi}_j \widehat{\lambda^j} \quad \text{ou} \quad C_j x - c_j = \pi_j \lambda^j - \widehat{\pi}_j \widehat{\lambda^j}$$

traduisent le souhait de respecter une borne inférieure $\widehat{c_j}$, ou alors d'atteindre une valeur but c_j.

La méthode STEM (Ph. Vincke, 1976)

Cette approche nécessite l'intervention d'un décideur qui relâche un objectif dans l'espoir de gagner sur les autres.

Etape 0: On calcule la matrice de gain

$$P_{i,j} = C_j y^{(i)}, \quad y^{(i)} = \arg\min\{C_i x : x \in P\} \quad \text{(pas forcément unique).}$$

Calculer dispersion relative du $j$ieme objectif $t_j = \frac{\max_i P_{i,j} - P_{j,j}}{P_{j,j}}$, et pénalité $\pi_j = t_j / \sum_i t_i$. Poser $I = \{\}, r = 0$.

Etape 1: $r \leftarrow r + 1$, calcul d'un compromis $x^{(r)}$ comme solution optimale de

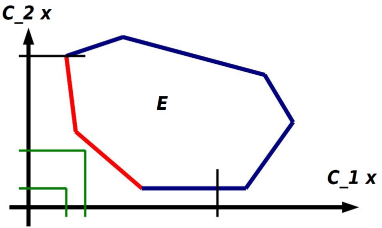
$$v^{(r)} := \min\{\lambda : x \in P, \lambda \in [0, +\infty), \forall j \notin I : C_j x \leqslant P_{j,j} + \pi_j \lambda, \forall j \in I : C_j x \leqslant c_j\}$$

Etape 2: le décideur a le choix entre

- accepter un des compromis $x^{(1)}, ..., x^{(r)}$ et arrêt;
- si card$(I) < \ell$: en considérant les variables duales associés aux contraintes $j \notin I$, choisir l'indice $s \notin I$ de la contrainte à relâcher. Choix de la borne $c_s > P_{s,s} + \pi_s v^{(r)}$, et ajout de s à I.
- en considérant les variables duales associés aux contraintes $j \in I$: ajuster un des paramètres c_j pour $j \in I$;

Retour à l'étape 1.

Un exemple pour deux objectifs



On trace l'ensemble $E = \{Cx : x \in P\}$, un polyèdre du $\mathbb{R}^2$, et en rouge l'ensemble des solutions optimales au sens de Pareto. Ici on trouve

$$Cy^{(1)} = \begin{bmatrix} 2 \\ 8 \end{bmatrix} \text{ unique, } Cy^{(2)} = \begin{bmatrix} 10 \\ 1 \end{bmatrix} \text{ non unique, } P = \begin{bmatrix} 2 & 8 \\ 10 & 1 \end{bmatrix}, \pi = \begin{bmatrix} 4/11 \\ 7/11 \end{bmatrix}$$

Le premier problème auxiliaire est alors

$$\min\{\lambda : x \in P, \lambda \in [0, +\infty), Cx \leqslant \begin{bmatrix} 2 \\ 1 \end{bmatrix} + \lambda \begin{bmatrix} 4/11 \\ 7/11 \end{bmatrix}\}.$$

On a tracé en vert les deux rectangles obtenus pour $\lambda = 0$ et pour $\lambda = 33/10$, la solution optimale étant un peu plus grande.