**Tutoriel - La méthode SÉP**

* **Leçon 1. Principes de la méthode : un exemple géométrique. ……………… 1**
	+ 1. Énoncé de l’exemple. 1
		2. Résolution géométrique de la relaxation continue. 1
		3. Résolution géométrique du modèle en nombres entiers. 1
		4. Borne pour la valeur optimale *z*\*. 2
		5. Méthode SÉP : séparation. 2
		6. Méthode SÉP : évaluation et élimination. 2
		7. Arbre d’énumération de l’exemple et vocabulaire. 3
		8. La gestion des bornes et de la liste. 4
* **Leçon 2. Utilisation du didacticiel. …………………………………………… 5**
	1. Démarrage du didacticiel : choix du problème à traiter. 5
	2. Le critère de séparation. 5
	3. Structure de l’écran type du didacticiel. 6
	4. Les itérations de la méthode SÉP. 6
	5. Résolution de l’exemple. 6
* **Leçon 3. Quelques cas particuliers. …………………………………………… 8**
	1. Autre critère de séparation : le critère du meilleur *cj*. 8
	2. Les cas d’égalité. 9
* **Tableaux, arbres et figures. …………………………………………………… 10**
* **Annexe 1. Le vocabulaire des modèles linéaires. ………………………………. 16**
* **Annexe 2. Méthode géométrique de résolution d’un modèle continu à 2 variables. 18**
* **Leçon 1. Principes de la méthode : un exemple géométrique.**

**1.1 Énoncé de l’exemple.**

Nous exposerons d’abord les principes de la méthode SÉP à partir d’un exemple comportant deux variables de décision. Par la suite, en leçon 2, nous verrons la méthode à l’œuvre sur un exemple plus complexe et plus représentatif.

Mais, pour l’instant, nous nous intéressons au modèle totalement en nombres entiers (P) suivant (tableau T1). L’intérêt de cet exemple lilliputien est que sa région admissible se représente par un graphique cartésien, ce qui permet d’illustrer géométriquement comment procède la méthode SÉP.

**1.2 Résolution géométrique de la relaxation continue.**

Oublions temporairement les contraintes d’intégrité (4). Et considérons le modèle continu (P0) consistant à maximiser *z* sous les contraintes (1) à (3). Nous dirons que (P0) est la *relaxation continue* de (P).

Nous présumons que vous êtes familier avec la méthode géométrique de résolution d’un modèle continu à deux variables. Si ce n’est pas le cas, référez-vous à l’annexe 2, ou encore à votre manuel de recherche opérationnelle. Donc, nous supposons que vous savez comment résoudre géométriquement un modèle continu à deux variables. Rappelons qu’on en construit d’abord la région admissible. Dans le cas du modèle (P0), il s’agit du polygone OABC présentement affiché (figure F2). Ensuite, on trace une droite de niveau de la fonction-objectif. La droite en pointillés présentement affichée correspond aux solutions pour lesquelles *z* prend la valeur 36 (figure F1). Enfin, comme *z* augmente dans la direction indiquée par la flèche en rouge et que nous cherchons à maximiser *z*, nous déplaçons la droite aussi loin que possible, mais sans quitter la région admissible. Nous constatons que le point le plus éloigné est le sommet B, lequel est donc l’unique optimum du modèle (P0).

Convenons de noter *z*0 la valeur optimale de la relaxation continue (P0). Ici, *z*0 est la valeur de la fonction-objectif *z* en B et *z*0 = 210.

**1.3 Résolution géométrique du modèle en nombres entiers.**

Revenons maintenant à notre modèle (P) en nombres entiers. Les solutions admissibles de (P) sont les points (*x*1;*x*2)

* tels que (*x*1;*x*2) est une solution admissible de la relaxation continue (P0);
* et tels que, de plus, *x*1et*x*2 sont des nombres entiers.

Géométriquement, l’ensemble des solutions admissibles de (P) est la grille des points à coordonnées entières situés dans le polygone OABC ou sur sa frontière (figure F2). Pour déterminer un optimum de (P), il suffit, comme dans le cas d’un modèle continu, de tracer une droite de niveau de la fonction-objectif, droite que nous poussons ensuite aussi loin que possible. Nous constatons que, dans notre exemple, l’unique solution optimale correspond au point (4; 4), pour lequel la fonction-objectif prend la valeur 192, valeur que nous noterons *z*\*.

Voici une autre façon de trouver l’optimum (4; 4) du modèle en nombres entiers. Nous résolvons d’abord la relaxation continue, obtenant ainsi, nous l’avons vu précédemment, la droite de niveau 210 qui passe par le sommet B; comme la 1re coordonnée de B n’est pas entière, nous déplaçons ensuite la droite de niveau vers le bas, jusqu’à l’obtention d’un point admissible dont les coordonnées sont entières. Nous retrouvons ainsi le point (4; 4).

**1.4 Borne pour la valeur optimale *z*\*.**

D’après ce qui précède, la fonction-objectif *z* atteint une valeur de 210 dans la région admissible de la relaxation, tandis que cette même fonction-objectif est limitée à 192 si on exige de plus que les coordonnées *x*1et*x*2 soient toutes deux entières. Ainsi, dans notre exemple, *z*\* ≤  *z*0.

Cette inéquation est valide pour tout modèle en nombres entiers de maximisation. Voyons pourquoi.

* L’exemple traité jusqu’ici et représenté dans le graphique de gauche correspond au cas le plus fréquent : une fois atteint le sommet optimal de la relaxation, nous devons revenir en arrière pour trouver un point du polygone dont toutes les coordonnées soient entières; ce retour en arrière s’accompagne d’une diminution de *z*, et *z*\* est alors inférieur à *z*0.
* Il peut également arriver que la droite de niveau *z*0 contienne un ou plusieurs points de coordonnées entières qui appartiennent à la région admissible (prendre le modèle (P), mais redéfinir la fonction-objectif ainsi : *z* = 10*x*1 + 15*x*2). Chacun de ces points constitue un optimum du modèle en nombres entiers; et *z*\* est alors égal à  *z*0. Le graphique de droite illustre cette dernière situation : les points P, Q et C sont des optima entiers du modèle; de plus, *z*\* et *z*0 prennent ici la même valeur 100.

En résumé, deux situations peuvent se présenter, selon que la droite de niveau  *z*0 contient ou non un point de la région admissible dont toutes les coordonnées sont entières. Et, dans les deux cas, *z*\* ≤  *z*0.

**1.5 Méthode SÉP : séparation.**

Nous cherchons maintenant à résoudre le modèle (P) en nombres entiers traité précédemment, mais en utilisant une méthode, dite SÉP, qui ne nécessite pas de graphique et s’adapte à des modèles comportant plus de deux variables de décision. Cependant, afin de mieux expliquer et de bien motiver cette approche, nous allons illustrer géométriquement ce que nous allons faire.

La méthode SÉP – un sigle pour séparation et évaluation progressive – repose sur deux opérations, qu’elle répète itérativement aussi souvent que nécessaire. La première, dite séparation, consiste à diviser la région admissible de la relaxation continue (P0) en diverses sous-régions, qui seront analysées séparément.

Dans notre exemple, nous partageons d’abord le polygone OABC en 3 parties, selon que la première coordonnée *x*1 est inférieure ou égale à 2, entre 2 et 3, ou encore supérieure ou égale à 3. Puisqu’il n’existe aucun nombre entier entre 2 et 3, nous ne retrouverons aucune solution admis­sible de (P) dans la portion centrale. Par conséquent, un optimum du modèle (P) appartient néces­sairement à l’une ou l’autre des sous-régions extrêmes.

**1.6 Méthode SÉP : évaluation et élimination.**

Nous avons mentionné que, pour déterminer une solution optimale de notre modèle (P) en nombres entiers, il nous suffira de chercher dans les sous-régions gauche et droite du graphique présentement affiché. Par laquelle commencerons-nous?

Si nous ne disposions pas du graphique, nous ne saurions pas dans laquelle des sous-régions extrêmes se trouve la solution optimale du modèle (P). Et choisir au hasard l’ordre de traitement des sous-régions serait inefficient, comme nous le verrons plus loin. Il nous faut donc un critère pour décider de cet ordre de traitement. Notre approche – et c’est ce qui explique la 2e lettre du sigle SÉP – consiste à d’abord **évaluer** la valeur optimale de la fonction-objectif dans chaque sous-région. Il serait possible de déterminer ces valeurs géométriquement. Mais, nous préférons les calculer à l’aide d’un logiciel. Nous obtenons ainsi que

* la valeur optimale de la fonction-objectif *z* dans la sous-région de gauche est 177;
* dans celle de droite, la valeur optimale est 204 et est atteinte au point dont les coordonnées sont 3 et4,667.

Il sera utile de présenter cette information sous la forme visuelle d’un *arbre d’énumération* (arbre 1). La boîte du haut correspond au polygone OABC; les deux boîtes de la ligne suivante, aux deux sous-régions extrêmes. Dans chaque boîte, la section supérieure donne la valeur maximale de la fonction-objectif dans la sous-région associée et la section inférieure, les coordonnées d’une solution optimale.

Puisque *z*2 est supérieur à *z*1, il est raisonnable de croire qu’une solution maximale du modèle a plus de chance de se retrouver dans la sous-région de droite que dans celle de gauche. Par consé­quent, nous nous intéresserons en premier lieu à cette sous-région de droite. Observant que la solution optimale de la boîte (P2) n’est pas entière, nous séparons la sous-région associée selon que la 2e coordonnée est inférieure ou égale à 4, entre 4 et 5, ou encore supérieure ou égale à 5. Puisqu’il n’existe aucun nombre entier entre 4 et 5, nous ne retrouverons aucune solution admissible de (P) dans la portion centrale. Par conséquent, il nous suffira à nouveau de considérer les deux cas extrêmes. Ceux-ci seront analysés séparément, à l’aide d’un logiciel approprié. Nous obtenons ainsi que (arbre 1) :

* la sous-région correspondant à « *x*2 ≥ 5» n’admet aucune solution admissible;
* dans l’autre, la valeur optimale 192 est atteinte au point (4; 4) dont les deux coordonnées sont entières.

L’existence d’une solution entière de valeur 192 garantit que la valeur maximale *z*\* est d’**au moins** 192. Nous allons montrer que *z*\* est également **au plus** 192, d’où nous conclurons que *z*\* est **égal** à 192 et que le point (4; 4) est en fait une solution optimale du modèle (P).

Donc, nous voulons montrer que 192 est une borne supérieure de *z*\*. Le polygone OABC, qui est la région admissible de la relaxation continue (P0), a été séparé en 5 sous-régions disjointes. Les deux sous-régions hachurées sur le graphique présentement affiché ne contiennent aucun point dont les deux coordonnées sont entières et peuvent donc être ignorées dans la recherche d’une solution optimale du modèle (P) en nombres entiers. De même, la sous-région associée à (P4) peut être ignorée, car elle est vide en fait. Enfin, la valeur maximale de la fonction-objectif dans la sous-région associée à (P1) est égale à 177, ce qui inférieur à la valeur 192. Conclusion : nulle part dans la région admissible, il ne peut exister de solution entière pour laquelle la fonction-objectif prend une valeur supérieure à 192 et, par conséquent, *z*\* ne peut dépasser 192.

* 1. **Arbre d’énumération de l’exemple et vocabulaire.**

Reprenons l’arbre d’énumération de notre exemple (arbre 1). Dorénavant, nous parlerons de *noeuds* au lieu de boîtes. Nous dirons que (P1) et (P2) ont été obtenus en séparant (P0) selon la variable *x*1; que (P0) est le *nœud-père*, (P1) et (P2), les *nœuds-fils*; que *x*1 est la *variable de séparation*.

À chaque nœud est associé un modèle linéaire **continu** (P*h*). Dans le cas où *h* = 0, il s’agit de la relaxation continue. Les modèles (P*h*), où *h* > 0, sont obtenus de (P0) en imposant des bornes à l’une ou l’autre des coordonnées. Par exemple,

(P2) est le modèle (P0) auquel on adjoint la contrainte « *x*1 ≥ 3»

(P3) est le modèle (P2) auquel on adjoint la contrainte « *x*2 ≤ 4».

On notera que l’ajout des bornes est cumulatif. Par exemple, (P3) est obtenu de (P0) par l’ajout de deux contraintes de bornes, soit « *x*1 ≥ 3» **et** « *x*2 ≤ 4».

* 1. **La gestion des bornes et de la liste.**

L’exemple traité ici est très simple et il a été facile de construire l’arbre d’énumération. Dans la plupart des cas cependant, il sera utile de disposer en tout temps des deux informations suivantes : des bornes inférieure et supérieure pour la valeur optimale *z*\* et une liste de «nœuds en attente». Ces données devront être mises à jour après chaque séparation.

Nous illustrons maintenant ces notions à l’aide de l’exemple résolu précédemment. Au départ, l’arbre d’énumération se réduit au nœud (P0). À gauche et à droite de l’arbre partiel (voir le tutoriel), sont affichés un intervalle et une liste :

* L’intervalle indique que, comme nous l’avons vu en section 1.4, la valeur optimale *z*\* est inférieure ou égale à la valeur maximale 210 du nœud (P0).
* Le nœud (P0) est dit *en attente* d’être séparé. La liste se réduit ici à ce seul nœud. Dans d’autres situations, cette liste pourra être vide, ou encore contenir plusieurs éléments.

Nous procédons maintenant à la 1re séparation et calculons à l’aide d’un logiciel approprié les solutions optimales des nœuds (P1) et (P2). Il faut ensuite mettre à jour l’intervalle et la liste :

* Comme nous l’avons mentionné en section 1.5, le point optimal (*x*1;*x*2) du modèle (P) appar­tient à l’une ou l’autre des sous-régions extrêmes, selon que *x*1est inférieure ou égale à 2, ou bien supérieure ou égale à 3. Dans le premier cas, *z*\*, qui est la valeur de la fonction-objectif au point optimal, ne peut excéder 177; dans l’autre, *z*\* ne peut excéder 204. Même si nous ne savons pas encore dans laquelle des sous-régions extrêmes se trouve le point optimal, nous pouvons conclure à coup sûr que 204 est une borne supérieure de *z*\*.
* Le nœud (P0) a déjà été séparé et n’est donc plus en attente; la liste se compose maintenant des nœuds (P1) et (P2).

Pour la prochaine séparation, nous devons d’abord choisir entre les nœuds (P1) et (P2). Tel que mentionné en section 1.6, nous retenons le nœud (P2) parce que sa valeur optimale 204 est plus élevée que celle de (P1). Ensuite, nous calculons à l’aide d’un logiciel approprié le contenu des nœuds (P3) et (P4). Il faut enfin mettre à jour l’intervalle et la liste :

* Le point optimal du modèle (P) appar­tient à l’une ou l’autre des sous-régions associées aux nœuds (P1), (P3) ou (P4) : par conséquent, sa valeur *z*\* ne peut, en aucun cas, excéder 192.
* La solution optimale du modèle (P3) est entière et sa valeur 192 constitue une borne inférieure de *z*\*, qui est la valeur maximale parmi les solutions entières du modèle (P).
* Le nœud (P2) vient d’être séparé et n’est donc plus en attente. La région admissible de (P4) est vide et il serait inutile d’y rechercher une solution entière : on peut donc éliminer ce nœud de la liste des nœuds en attente. La solution optimale du modèle (P3) est entière et il est impossible de trouver une meilleure solution dans cette sous-région : il serait donc inutile de la diviser, et nous éliminons le nœud (P3) de la liste des nœuds en attente. Enfin, le nœud (P1) également est éliminé, car la sous-région associée, dont la valeur maximale 177 est moins intéressante que la valeur 192 de la solution entière (4;4) déjà connue, ne peut contenir de solution optimale.

La liste des nœuds en attente est maintenant vide. La méthode SÉP s’arrête. La meilleure solution entière apparaissant dans l’arbre est une solution optimale du modèle (P). Dans le présent exemple, l’arbre contient une seule solution entière, qui automatiquement est optimale : il s’agit de la solution (4;4) du nœud (P3). Enfin, la valeur optimale *z*\* est égale au *z* de ce nœud (P3), soit 192.

Avant de quitter cet exemple, revenons sur les différents motifs qui nous ont permis de décider qu’un nœud donné ne devrait plus être considéré comme candidat à la séparation : un noeud (P*h*) est retiré de la liste des nœuds en attente :

* quand la sous-région associée à ce nœud ne contient aucune solution admissible (dans notre exemple, (P4) est éliminé sur la base de ce critère);
* ou encore, quand les contraintes d’intégrité du modèle (P) sont satisfaites par la solution optimale reportée dans ce nœud (c’est le cas de (P3) dans notre exemple);
* enfin, quand la valeur *zh* de (P*h*) n’est pas plus intéressante que la valeur de *z* pour une solution déjà connue qui satisfait à toutes les contraintes d’intégrité du modèle (P) (ici, (P1) a été éliminé parce que sa valeur optimale 177 est inférieure à la valeur 192 de la solution (4 :4) dont les deux coordonnées sont entières).
* **Leçon 2. Utilisation du didacticiel.**

**2.1 Démarrage du didacticiel : choix du problème à traiter.**

Le didacticiel, une fois lancé, affiche un menu proposant une liste d’exemples pré-programmés (voir le didacticiel…). Le plus souvent, on choisira un des problèmes pré-programmés de la liste affichée. Mais, l’usager ou son professeur peuvent construire des exemples additionnels à l’aide d’un gabarit en format Excel qui est disponible sur le site Internet de l’auteur; l’usager accède à ces problèmes en cliquant sur l’icône au bas de la liste.

Convenons de cliquer sur le problème 7*b* de la liste (tableau T2). Le didacticiel résout alors la relaxation continue (P0) du modèle et en affiche une solution optimale (arbre A2). Pour alléger, il omet les variables dont la valeur est nulle dans la solution optimale du nœud. Ainsi, la solution optimale de (P0) est la suivante : *x*1 prend la valeur 2,4 et *x*3 prend la valeur 0,2; enfin, les variables non affichées, soit *x*2 et *x*4, sont nulles.

**2.2 Le critère de séparation.**

Cet exemple présente dès le départ une difficulté que nous n’avions pas rencontrée dans l’exemple traité en leçon 1 : plusieurs variables prennent dans la solution optimale du nœud (P0) une valeur qui n’est pas entière. Vaut-il mieux effectuer la prochaine séparation selon *x*1 ou bien selon *x*3? Divers critères peuvent présider au choix de la *variable de séparation*. Le didacticiel en accepte deux :

* Le *critère du meilleur cj* : dans le cas d’un modèle de maximisation, on sépare selon une variable non entière dont le coefficient dans la fonction-objectif est aussi grand que possible. Ici, nous choisirions *x*1 car son coefficient 10 est supérieur à celui de *x*3.
* Le *critère de la variable la plus distante* : on sépare selon une variable non entière qui est le plus éloignée possible de l’entier le plus près. Dans le cas du nœud (P0) affiché, ce critère recommande lui aussi la variable *x*1.

C’est ce dernier critère qui est utilisé dans le problème que nous avons choisi.**2.3 Structure de l’écran type du didacticiel.**

L’écran du didacticiel est divisé en 3 parties (voir le didacticiel) : une ligne de titre ; une section centrale ; et un pied de page où est affiché un message décrivant l’opération à effectuer par l’usager. La section centrale contient l’arbre d’énumération en construction et, au-dessus, une «barre d’information» qui résume les données requises pour gérer la construction de l’arbre :

* au centre de la barre, sont précisés le type d’optimisation (maximisation ou minimisation), ainsi que le critère retenu pour sélectionner la variable de séparation ;
* à gauche, se trouve une zone où sont inscrites les bornes inférieure *z* et supérieure  d’un intervalle dans lequel doit se trouver la valeur optimale *z*\* de la fonction-objectif ;
* enfin, dans la zone de droite, sont énumérés les nœuds en attente.

Les bornes, de même que la liste des nœuds en attente, doivent être mises à jour par l’usager lors de chaque itération.

Un bouton « ***Énoncé*** » au bas de la barre d’information, à gauche, permet à l’usager de visualiser l’énoncé du problème en cours de traitement. L’énoncé est affiché dans une zone de texte assez vaste qui occupe une portion considérable de la section centrale de l’écran et cache parfois en grande partie l’arbre d’énumération. Pour diminuer cette zone, il suffit de cliquer sur le bouton  et alors seule la fonction-objectif est donnée. On peut également fermer cette zone en utilisant le bouton  .

**2.4 Les itérations de la méthode SÉP.**

Après avoir cliqué sur un des exemples proposés, l’usager est mis en présence d’un arbre d’énumération partiel formé du seul noeud initial (P0). Commencent alors les itérations de la méthode SÉP. L’usager doit successivement :

* choisir le prochain nœud à séparer;
* cliquer sur la variable selon laquelle s’effectuera la séparation ;
* déterminer les bornes inférieure et supérieure d’un intervalle dans lequel doit se trouver la valeur optimale *z*\* du modèle (P) ;
* mettre à jour la liste des nœuds en attente.

Après la 2e étape, soit le choix de la variable, le didacticiel affiche les deux nœuds-fils résultant de la séparation ; ensuite, il amène l’usager au début de la prochaine étape, lui demandant d’entrer les valeurs des bornes inférieure, puis supérieure. Lors de la 1re itération, la première étape est court-circuitée : en effet, dans tous les problèmes, la liste se réduit alors au seul noeud (P0), qui automatiquement devra être choisi comme le prochain nœud à séparer.

Le didacticiel teste chacune des décisions de l’usager et, en cas d’erreur, l’avise sans délai par un message approprié.

**2.5 Résolution de l’exemple.**

Revenons à notre exemple (tableau T2). Initialement, le didacticiel demande de cliquer sur la variable selon laquelle s’effectuera la prochaine séparation. Tel qu’indiqué dans la barre d’information, le critère utilisé ici est celui de la variable la plus distante: nous cliquons donc sur la variable *x*1 du nœud (P0). Le didacticiel résout alors les modèles (P1) et (P2) et en affiche une solution optimale. Il nous faut maintenant entrer les nouvelles bornes dans la section gauche de la barre d’information. Puisque l’arbre ne contient aucune solution entière pour l’instant, nous n’avons pas de valeur à fournir pour la borne inférieure; par conséquent, conformément au message affiché au bas de l’écran, nous appuyons sur la touche Retour. C’est maintenant le tour de la borne supérieure : la valeur optimale *z*\* du modèle (P) ne peut excéder la valeur 24,71 du nœud (P1); nous entrons donc cette valeur 24,71, puis appuyons sur la touche Retour. La prochaine étape consiste à mettre à jour la liste des nœuds en attente. Le nœud (P0), qui vient d’être séparé, doit être enlevé de la liste; nous cliquons donc sur le bouton (P0) apparaissant dans la barre d’information, à l’extrême-droite. De même, le nœud (P2) doit être éliminé de la liste, car la région admissible associée est vide; nous cliquons donc sur le bouton (P2) de la barre. Enfin, nous cliquons sur le bouton Fin à l’extrême-droite de la barre, pour indiquer que cette étape est terminée.

Commence maintenant la 2e itération. Le prochain nœud à séparer est nécessairement (P1), puisqu’il s’agit du seul élément de la liste des nœuds en attente. La variable de séparation sera *x*3. Après cette réponse, le didacticiel résout les modèles (P3) et (P4), puis affiche les nœuds correspondants. À nouveau, nous laissons vide la case associée à la borne inférieure; la borne supérieure est maintenant 23. Enfin, (P1) est enlevé de la liste des nœuds en attente; aucun autre nœud n’est éliminé.

Nous en sommes au début de la 3e itération, et le message demande de cliquer dans le nœud qui devra être séparé. Ici, le hasard a voulu que la valeur optimale de *z* soit la même, soit 23, dans les deux nœuds en attente. En principe, nous pourrions choisir (P3) ou (P4), indifféremment ; cependant, le didacticiel accepte seulement le premier. Si nous cliquons dans le nœud (P4), il affiche le message suivant :

« *Ce choix est correct ; mais nous avons choisi plutôt de faire la séparation à partir du nœud P3. Cliquer dans le nœud P3 pour continuer.* »

Cliquons donc sur le bouton Ok pour indiquer que nous avons pris connaissance du message, puis dans le nœud (P3). La variable de séparation est évidemment *x*2, car il s’agit de la seule dont la valeur dans le nœud (P3) n’est pas entière. Après cette réponse, le didacticiel affiche les nœuds-fils (P5) et (P6). Les bornes restent inchangées lors de cette itération. De plus, nous enlevons de la liste le nœud (P3) qui vient d’être séparé.

Une autre itération débute : le nœud (P4) sera séparé, selon la variable *x*2. Le didacticiel calcule et affiche ensuite une solution optimale des modèles (P7) et (P8). Une première : la solution optimale de ce dernier est entière; par conséquent, la valeur 19 de ce nœud servira de borne inférieure. La nouvelle borne supérieure 21,5 provient de (P5). L’itération se termine par la mise à jour de la liste : (P4), qui vient d’être séparé, est enlevé ; (P8), dont la solution est entière, est éliminé; enfin, (P6) est éliminé, car sa valeur 17 est moins élevée que celle de (P8).

On recommence : le nœud (P5) sera séparé, selon la variable *x*4. Une nouvelle solution entière est alors obtenue, dont la valeur 20 est meilleure que celle de la première. Nous révisons donc la borne inférieure. Quant à la borne supérieure, elle est maintenant égale à 20,33. Enfin, il faut enlever de la liste le nœud (P5), puis éliminer (P9) et (P10).

Pour paraphraser le vers de Boileau, «vingt fois sur le métier, remettons notre ouvrage»! Cette fois, c’est le nœud (P7) qui sera séparé; et la variable de séparation sera *x*1. La borne inférieure ne change pas, car les nœuds-fils n’apportent pas de nouvelle solution entière ; la borne supérieure est maintenant égale à 20. Enfin, le nœud (P7) est enlevé de la liste, puis (P11) et (P12) sont éliminés.

La liste des nœuds en attente est maintenant vide. Ce qui indique la fin de la construction de l’arbre (arbre A2). D’ailleurs, le didacticiel demande de «cliquer dans le nœud associé à une solution optimale». Il s’agit ici de (P9). Noter que les nœuds dont la solution est entière sont affichés en bleu, ce qui facilite la recherche dans l’arbre.

* **Leçon 3. Quelques cas particuliers.**

**3.1 Autre critère de séparation : le critère du meilleur *cj*.**

Dans l’exemple traité en leçon 2, nous avons utilisé le critère de la variable la plus distante pour sélectionner la variable de séparation. Certains des exemples pré-programmés utilisent plutôt le critère du meilleur *cj* qui recommande de séparer le nœud selon une variable non entière dont le coefficient dans la fonction-objectif est

* aussi grand que possible dans le cas d’un modèle de maximisation;
* aussi petit que possible dans le cas d’un modèle de minimisation.

Pour illustrer ce critère, nous reprenons le modèle (P) utilisé en leçon 2. Supposons que soit affiché le menu initial Sélection d’un problème. (Ce menu, c’est celui qui apparaît immé­dia­tement après le lancement du programme; on peut également y revenir en toute occasion, ou presque, en cliquant sur le bouton Retour au menu initial placé à droite de la barre des messages.) Donc, nous sommes devant le menu initial. Nous cliquons sur le problème 6*b*. On pourra être surpris que le problème choisi ici diffère de celui retenu en leçon 2, alors que nous avons annoncé que nous nous intéressons au même modèle. C’est que la donnée d’un problème comprend à la fois un modèle et un critère de séparation. Ainsi, dans la liste des exemple pré-programmés, le problème porte le numéro 6*b* quand le modèle (P) considéré ici est résolu selon le critère du meilleur *cj*, et le numéro 7*b* quand (P) est résolu selon le critère de la variable la plus distante.

En réponse à notre clic sur le problème 6*b*, le didacticiel résout la relaxation continue (P0) et en affiche une solution optimale (arbre A3). La barre d’information indique qu’il s’agit d’un problème de maximisation et que le critère retenu est celui du meilleur *cj*. Le message au bas de l’écran demande de «cliquer sur la variable *xj* selon laquelle s’effectuera la séparation». Comme le nœud (P0) contient plus d’une variable non entière, il nous faudra connaître les coefficients de la fonction-objectif pour choisir entre les variables candidates. À cette fin, nous cliquons sur le bouton « ***Énoncé*** » de la barre d’information, puis sur le bouton  pour réduire la zone de texte; ensuite, nous déplaçons la zone au bas de l’écran.

Dans le nœud (P0), les variables *x*1 et *x*3 prennent des valeurs non entières. De plus, le coefficient de *x*1 dans la fonction-objectif est 10, tandis que celui de *x*3 est 7. Par conséquent, la 1re sépa­ration s’effectuera selon *x*1. Noter que le critère de la variable la plus distante recommanderait ici le même choix. Nous cliquons donc sur la variable *x*1 dans le nœud (P0), puis nous complétons l’itération comme en leçon 2.

Le prochain nœud à séparer sera évidemment (P1), dans lequel les variables *x*2 et *x*3 prennent des valeurs non entières. Or, les coefficients de *x*2 et de *x*3 dans la fonction-objectif sont 12 et 7 respectivement. Nous cliquons donc sur la variable *x*2. Cette fois, le critère de la variable la plus distante favoriserait une variable différente.

Nous poursuivons ainsi la construction de l’arbre d’énumération. Le voici une fois parachevé (arbre A3). La solution optimale correspond au nœud (P7) : la fonction-objectif y prend la valeur 20; et la seule variable non nulle de la solution optimale est *x*1, dont la valeur est égale à 2. Noter que la solution optimale du nœud (P7) coïncide avec celle obtenue en leçon 2 en appliquant le critère de la variable la plus distante. Il arrive parfois que les deux critères donnent des solutions optimales différentes. Mais, évidemment, la valeur optimale *z*\* de la fonction-objectif est nécessairement la même.

**3.2 Les cas d’égalité.**

Nous indiquons dans cette section comment le didacticiel traite les cas d’égalité qui surviennent parfois lors de l’application de la méthode SÉP.

Pour illustrer le premier cas, reprenons l’arbre partiel suivant rencontré en section 2.5 (arbre A2, avec les nœuds (P0) à (P4) seulement) et plaçons-nous au moment où il s’agit de cliquer dans le prochain nœud à séparer. En principe, nous devons sélectionner un nœud dont la valeur est la plus grande possible. Ici, deux nœuds, soit (P3) et (P4), présentent la valeur maximale 23 et sont donc à égalité selon le critère utilisé. Mathématiquement, on pourrait choisir l’un ou l’autre des nœuds-candidats. Cependant, le didacticiel, en cas d’égalité, départage les candidats en lice en fonction de leur numéro : il retient en effet le nœud dont l’indice est le plus faible, et refuse les autres. Dans notre exemple, il considère que le prochain nœud à séparer est nécessairement (P3). Et, si l’usager clique dans le nœud (P4), il rejette ce choix et affiche le message suivant :

« *Ce choix est correct ; mais nous avons choisi plutôt de faire la séparation à partir du nœud P3. Cliquer dans le nœud P3 pour continuer.* »

L’usager doit alors cliquer sur le bouton Ok, puis dans le nœud (P3), conformément au message.

Il arrive également que plusieurs variables soient à égalité lors du choix de la variable de séparation. Considérons, parmi les exemples pré-programmés, le problème PRO6-09*b* (tableau T3) et plaçons-nous au moment où il faut choisir la variable selon laquelle le nœud (P3) sera séparé. Ici, les variables *x*3 et *x*4 sont à égalité selon le critère choisi, qui est celui de la variable la plus distante. On tente d’abord de les départager à l’aide de l’autre critère, soit celui du meilleur *cj* : comme le coefficient de *x*4 dans la fonction-objectif *z* est supérieur à celui de *x*3 et qu’il s’agit dans ce problème de maximiser *z*, le choix doit se porter sur *x*4. Cependant, si l’usager clique plutôt sur *x*3, le didacticiel ne parle pas de choix invalide, mais se fait nuancé :

« *Choix sous-optimal : il est généralement recommandé, lorsque plusieurs variables candidates sont à égalité selon le critère choisi, de départager ces variables à l’aide de l’autre critère.* »

Si les deux variables candidates *x*3 et *x*4 étaient encore à égalité selon le 2e critère, le didacticiel retiendrait, plus ou moins arbitrairement, celle dont l’indice est le plus faible, soit *x*3. L’usager, dont le choix se porterait plutôt sur *x*4, ne commettrait pas d’erreur, mais s’écarterait du déroulement prévu ; le didacticiel afficherait alors le message suivant :

« *Ce choix est correct : cependant, nous avons choisi de séparer selon la variable de plus petit indice parmi celles qui optimisent les deux critères.* »

**Tutoriel - La méthode SÉP - Tableaux**

|  |
| --- |
| **T1. Le modèle (P) de l’exemple à deux variables** |
|  Max *z* = 12*x*1 + 36*x*2  sous les contraintes : 2*x*1 + 3*x*2 ≤ 20 (1) -6*x*1 + 4*x*2 ≤ 5 (2) *x*1 , *x*2 ≥ 0 (3) *x*1 , *x*2 entiers. (4) |

Exemple tiré de *Méthodes d'optimisation pour la gestion*; par Y. Nobert, R. Ouellet et R. Parent; 2008, Gaëtan Morin, éditeur; p. 230.

Voir aussi *Problèmes résolus de recherche opérationnelle*; par Y. Nobert, R. Ouellet et R. Parent; 1999, Gaëtan Morin, éditeur; page 97.

|  |
| --- |
| **T2. Modèle de l’exemple de la leçon 2** |
|  Max *z* = 10*x*1 + 12*x*2 + 7*x*3 + 9*x*4  sous les contraintes : 3*x*1 + 2*x*2 + 4*x*3 + 5*x*4 ≤ 8 2*x*1 + 4*x*2 + 1*x*3 + 6*x*4 ≤ 5 *xj* ≥ 0 pour *j* = 1, 2, 3, 4 *xj* entier pour *j* = 1, 2, 3, 4. |

Exemple tiré de : *La recherche opérationnelle, 3e édition*;

par Y. Nobert, R. Ouellet et R. Parent; 2001, Gaëtan Morin, éditeur; page 312.

|  |
| --- |
| **T3. Modèle de l’exemple de la section 3.2** |
|  Max *z* = *x*1 + 12*x*2 − 3*x*3 + 4*x*4 + 12*x*5  sous les contraintes : *x*1 + 4*x*2 + 4*x*3 + 4*x*4 ≤ 35 9*x*2  + 3*x*5 ≥ 18 11*x*1 + 2*x*2 + 7*x*4 + 2*x*5 ≤ 34 2*x*2 + 3*x*3 + 4*x*4 + 7*x*5 ≤ 80 *xj* ≥ 0 pour *j* = 1, 2, 3, 4, 5 *xj* entier pour *j* = 1, 2, 3, 4, 5. |

Exemple tiré de *Méthodes d'optimisation pour la gestion*; par Y. Nobert, R. Ouellet et R. Parent; 2008, Gaëtan Morin, éditeur; p. 242.

Voir aussi *Problèmes résolus de recherche opérationnelle*; par Y. Nobert, R. Ouellet et R. Parent; 1999, Gaëtan Morin, éditeur; 101.

**Tutoriel - La méthode SÉP - Arbres**

**A1. Arbre d’énumération de l’exemple à deux variables**

**A2. Arbre d’énumération de l’exemple de la leçon 2**

*x*2 5

*x*2 4

*x*1 

*x*1 

P1: *z*1 = 177

*x*1 = 2

*x*2 = 4,25

P3: *z*3 = 192

*x*1 = 4

*x*2 = 4

P0: *z*0 = 210

*x*1 = 2,50

*x*2 = 5

P4

Aucune solution admissible

P2: *z*2 = 204

*x*1 = 3

*x*2 = 4,667

P3: *z*3 = 23

*x*1 = 2

*x*2 = 0,25

*x*1 

*x*1 

*x*3 = 0

*x*3 

*x*2 = 0

*x*2 

*x*2 = 0

*x*2 

*x*4 = 0

*x*4 

*x*1 

*x*1 

P1: *z*1 = 24,71

*x*1 = 2

*x*2 = 0,14

*x*3 = 0,43

P4: *z*4 = 23

*x*1 = 1

*x*2 = 0,5

*x*3 = 1

P12

Aucune solution admissible

P9: *z*9 = 20

*x*1 = 2

P11: *z*11 = 18,8

*x*1 = 1

*x*3 = 1

*x*4 = 0,2

P2

Aucune solution admissible

P0: *z*0 = 25,4

*x*1 = 2,4

*x*3 = 0,2

P8: *z*8 = 19

*x*2 = 1

*x*3 = 1

P6: *z*6 = 17

*x*1 = 0,5

*x*2 = 1

P10

Aucune solution admissible

P7: *z*7 = 20,33

*x*1 = 1,33

*x*3 = 1

P5: *z*5 = 21,5

*x*1 = 2

*x*4 = 0,17

**A3. Arbre d’énumération de l’exemple de la section 3.1**

*x*3 = 0

P6

Aucune solution admissible

*x*3 

*x*4 = 0

*x*4 

*x*2 = 0

*x*2 

*x*1 

*x*1 

*x*1  1

P3: *z*3 = 23,53

*x*1 = 2

*x*3 = 0,37

*x*4 = 0,11

P1: *z*1 = 24,71

*x*1 = 2

*x*2 = 0,14

*x*3 = 0,43

P7: *z*7 = 20

*x*1 = 2

P0: *z*0 = 25,4

*x*1 = 2,4

*x*3 = 0,2

P8: *z*8 = 20,33

*x*1 = 1,33

*x*3 = 1

P5: *z*5 = 23,5

*x*1 = 2

*x*3 = 0,5

P4: *z*4 = 19

*x*2 = 1

*x*3 = 1

*x*1 

P9: *z*9 = 18,75

*x*1 = 1

*x*3 = 1,25

P10

Aucune solution admissible

P2

Aucune solution admissible

**A4. Arbre d’énumération de l’exemple de la section 3.2**

*x*5 

*x*5 

*x*2 

*x*2 

P0: *z*0 = 615,27

*x*2 = 18,5455

*x*3 = 2,1818

*x*5 = 3,2727

P2

Aucune solution admissible

P1: *z*1 = 606,86

*x*2 = 18

*x*3 = 2

*x*5 = 3,4286

P4: *z*4 = 576

*x*2 = 16

*x*3 = 1,333

*x*5 = 4

P3: *z*3 = 595,5

*x*2 = 18

*x*3 = 2,375

*x*4 = 0,375

*x*5 = 3

**Tutoriel - La méthode SÉP - Figures**

**F1. La région admissible de (P0) et quelques droites de niveaux de *z***



**F2. Région admissible du modèle (P) du tableau 1**

****

**Annexe 1. Le vocabulaire des modèles linéaires**

Un *modèle d’optimisation* est constitué de la donnée

* d’un ensemble fini de variables, notées *x*1, *x*2, …, *xn*, qui a priori prennent leurs valeurs dans l’ensemble des nombres réels;
* d’une fonction-objectif *z* = *f*(*x*1, …, *xn*) que l’on cherche à optimiser (maximiser ou minimiser);
* d’un ensemble fini de contraintes.

Nous considérons ici deux types de contraintes.

* Contraintes fixant le type d’une variable. Nous nous limitons ici aux deux types suivants :

 *xj* ≥ 0 (contrainte de non-négativité)

 *xj* est un nombre entier (contrainte d’intégrité)

Une contrainte qui exigerait que *xj* soit une variable binaire est remplacée ici par «*xj* ≥ 0 et *xj* ≤ 1 et *xj* est un nombre entier».

* *Contraintes technologiques* dans lesquelles un membre gauche est comparé à une constante. Par exemple,

 *x*1 + 3*x*2 = 7

 *x*1 *x*2 < 0

 cos (*x*1 *x*2) + sin (*x*7) ≠ 0

 

Une *fonction* des variables *x*1, …, *xn* est dite *linéaire* lorsqu’elle s’exprime comme la somme de termes du 1er degré. Ainsi, les fonctions dont les valeurs sont données par l’une ou l’autre des expressions suivantes

 *x*1 + *x*2

 3*x*1 – 7*x*4 + 8*x*10

 *x*1 + *x*2 + … + *xn*

sont linéaires, tandis que celles dont les valeurs sont données par les expressions

 *x*1 *x*2

 

 sin (*x*1) + *x*2

 

ne sont pas linéaires. Un *modèle* d’optimisation est dit *linéaire* quand les conditions suivantes sont satisfaites :

* La fonction-objectif est linéaire.
* Chaque contrainte technologique s’écrit comme une équation, ou encore comme une inéquation de signe ≤ ou ≥.
* Dans toute contrainte technologique, le membre gauche est linéaire et le membre droit se réduit à une constante. Une telle contrainte s’écrit toujours sous la forme

*a*1*x*1 + *a*2*x*2 + … + *anxn*   *b*.

* Toute variable *xj* est soumise à une contrainte de non-négativité.

Un modèle linéaire PL s’exprime toujours sous la forme suivante, où *E* est inclus dans {1, 2, …, *n*} et dénote l’ensemble des indices *j* des variables *xj* qui sont soumises à une contrainte d’intégrité.

 Max (Min) *z* = *c*1*x*1 + *c*2*x*2 + … + *cnxn*

 sous les contraintes :

 *a*11*x*1 + *a*12*x*2 + … + *a*1*nxn*  *b*1

 

PL

 *am*1*x*1 + *am*2*x*2 + … + *amnxn*  *bm*

 *xj* ≥ 0 *j* = 1, 2, …, *n*

 *xj* est entier *j* *E*

Un *modèle* linéaire est dit

* *continu* quand *E* = ∅ (aucune variable n’est soumise à une contrainte d’intégrité);
* *en nombres entiers* quand *E* ≠ ∅ (au moins une variable est soumise à une contrainte d’intégrité).
* *totalement en nombres entiers* quand *E* = {1, 2, …, *n*} (toutes les variables du modèle sont soumises à une contrainte d’intégrité);
* *mixte* quand *E* ≠ ∅ et *E* ≠ {1, 2, …, *n*} (certaines variables sont soumises à une contrainte d’intégrité, d’autres non).

**Annexe 2. Méthode géométrique de résolution d’un modèle continu à 2 variables**

**A2.1 Description de l’exemple**

Les modèles linéaires continus à deux variables de décision se résolvent facilement en analysant leur repré­sentation géométrique. Nous illustrons la procédure utilisée à l’aide de la relaxation continue (P0) de l’exemple de base traité en leçon 1. Ce modèle est donné au tableau 1 ci-dessous.

|  |
| --- |
| **Tableau 1. Le modèle (P0)** |
|  Max *z* = 12*x*1 + 36*x*2  sous les contraintes : 2*x*1 + 3*x*2 ≤ 20 (1) -6*x*1 + 4*x*2 ≤ 5 (2) *x*1 , *x*2 ≥ 0 (3) |

Nous nous intéressons en premier lieu à la région admissible de ce modèle (P0), c’est-à-dire à l’ensemble des points (*x*1 ; *x*2) qui satisfont à toutes les contraintes de (P0). Notons d’abord qu’en vertu des contraintes (3) de non-négativité, on peut se restreindre au premier quadrant et aux demi-axes qui le bornent. Nous traitons maintenant les contraintes technologiques une à une.

**A2.2 Les points qui satisfont à la contrainte (1)**

Nous indiquons ici comment déterminer la région des points qui satisfont à une inéquationlorsque les deux coefficients de son membre gauche sont positifs. Considérons donc l’inéquation (4) suivante :

 2*x*1 + 3*x*2 ≤ 24. (4)

(Nous avons choisi comme membre droit un multiple des coefficients 2 et 3 afin d’éviter les compli­cations inutiles dues aux fractions.) Nous nous intéressons d’abord à la *droite associée*, dont l’équation est :

 2*x*1 + 3*x*2 = 24. (5)

Pour tracer la droite (5), il suffit d’en connaître deux points. Les plus simples à trouver, ce sont ceux dont une coordonnée est nulle, c’est-à-dire les points de la forme (0; *x*2) et (*x*1; 0). Le premier s’obtient en remplaçant *x*1 par 0 dans (5), puis en résolvant :

 (2 × 0) + 3*x*2 = 24

 *x*2 = 24 / 3 = 8.

Ainsi, la droite (5) passe par le point P = (0; 8). Pour calculer la première coordonnée du point de la forme (*x*1; 0), on remplace *x*2 par 0 dans (5), puis on résout : on obtient ainsi que la droite (5) passe par le point Q = (12; 0). La figure 1 représente cette droite (5), de même que les droites parallèles d’équations

 2*x*1 + 3*x*2 = 36

 2*x*1 + 3*x*2 = 48.

La figure 1 montre que la valeur du membre gauche « 2*x*1 + 3*x*2» augmente dans la direction indiquée par la flèche. Par conséquent, un point (*x*1 ; *x*2) de coordonnées non négatives satisfait à la contrainte (4) si et seulement s’il appartient au triangle OPQ.

**Figure 1. Droites de niveaux « 2*x*1 + 3*x*2 = *d*»**



De même, la droite associée à la contrainte (1) du modèle (P0), dont l’équation est «2*x*1 + 3*x*2 = 20», passe par les points C = (10; 0) et R = (0; 20/3). Enfin, l’ensemble des points de coordonnées non négatives qui satisfont à la contrainte (1) correspond au triangle ombré de la figure 2.

**Figure 2. La région des points de coordonnées non négatives qui satisfont à la contrainte (1)**



**A2.3 Les points qui satisfont à la contrainte (2)**

L’approche est légèrement différente lorsque les coefficients du membre gauche ne sont pas de même signe. Traçons donc la droite associée à (2), dont l’équation est :

 -6*x*1 + 4*x*2 = 5. (6)

À nouveau, il s’agit de déterminer deux points de la droite. Le premier, qui est de la forme (0; *x*2), s’obtient en remplaçant *x*1 par 0 dans (6), puis en résolvant :

 (-6 × 0) + 4*x*2 = 5

 *x*2 = 5 / 4 = 1,25.

Pour le second, on peut rechercher un point de la forme (*x*1; 0), mais alors la première coordonnée est négative. On peut également choisir un point du premier quadrant dont l’une des coordonnées est entière : par exemple, on peut fixer *x*1 plus ou moins arbitrairement à la valeur 3, puis calculer l’autre coordonnée ainsi :

 4*x*2 = 5 + 6*x*1 = 5 + (6 × 3) = 23

 *x*2 = 23 / 4 = 5,75.

Ainsi, la droite (6) passe par les points A = (0; 1,25) et S = (3; 5,75).

Comme dans le cas traité dans les figures 1 et 2, le membre gauche «-6*x*1 + 4*x*2 » est inférieur à la constante 5 d’un côté de la droite (6), et supérieur à 5 de l’autre. Pour déterminer lequel des deux côtés fait l’affaire, il suffit de prendre un point quelconque hors de la droite et de tester s’il vérifie ou non la contrainte. Prenons le point O = (0; 0), pour lequel le calcul du membre gauche est particulièrement simple. Évidemment, le point (0; 0) satisfait à la contrainte (2) : ainsi, la région des points de coordonnées non négatives qui satisfont à la contrainte (2) correspond à la portion ombrée de la figure 3.

**Figure 3. La région des points de coordonnées non négatives qui satisfont à la contrainte (2)**



**A2.4 La région admissible du modèle (P0)**

La région admissible du modèle (P0) est formée des points (*x*1 ; *x*2)

* dont les deux coordonnées sont non négatives;
* et qui satisfont aux contraintes (1) et (2).

Cette région admissible est donc l’intersection des portions ombrées des figures 2 et 3. Elle est représentée à la figure 4 comme le polygone OABC.

**Figure 4. Construction de la région admissible du modèle (P0)**



**A2.5 Les droites de niveaux de la fonction-objectif**

Une solution optimale du modèle (P0) est un point admissible qui maximise la fonction-objectif. Considérons donc les points (*x*1; *x*2) qui donnent à la fonction-objectif *z* une valeur particulière, disons 36. Le lieu de ces points est la droite d’équation :

 12*x*1 + 36*x*2 = 36. (7)

Pour tracer la droite (7), il suffit d’en connaître deux points. Les plus simples à trouver, ce sont ceux dont une coordonnée est nulle, c’est-à-dire les points de la forme (*x*1; 0) et (0; *x*2). On résout donc :

 12*x*1 + (36 × 0) = 36 d’où *x*1 = 36 / 12 = 3

 (12 × 0) + 36*x*2 = 36 d’où  *x*2 = 36 / 36 = 1.

Ainsi, la droite (7) passe par les points (3; 0) et (0; 1). La figure 5 représente cette droite (7), de même que les droites de niveaux «*z* = 72» et «*z* = 108».

**Figure 5. Quelques droites de niveaux de la fonction-objectif**



**A2.6 Une solution optimale du modèle (P0)**

La figure 5 indique que la valeur de la fonction-objectif *z* augmente dans la direction de la flèche. On a donc intérêt, pour maximiser *z*, à pousser la droite de niveau aussi loin que possible. On constate facilement que le niveau *z* le plus élevé qu’on peut atteindre tout en gardant contact avec la région admissible est celui qui correspond au point B. Ainsi, B = (2,5; 5) est l’unique optimum du modèle (P0); et la valeur maximale la fonction-objectif *z* est :

 *z* = 12*x*1 + 36*x*2 = (12 × 2,5) + (36 × 5) = 210. (8)

**Figure 6. Détermination de la solution optimale du modèle (P0)**

