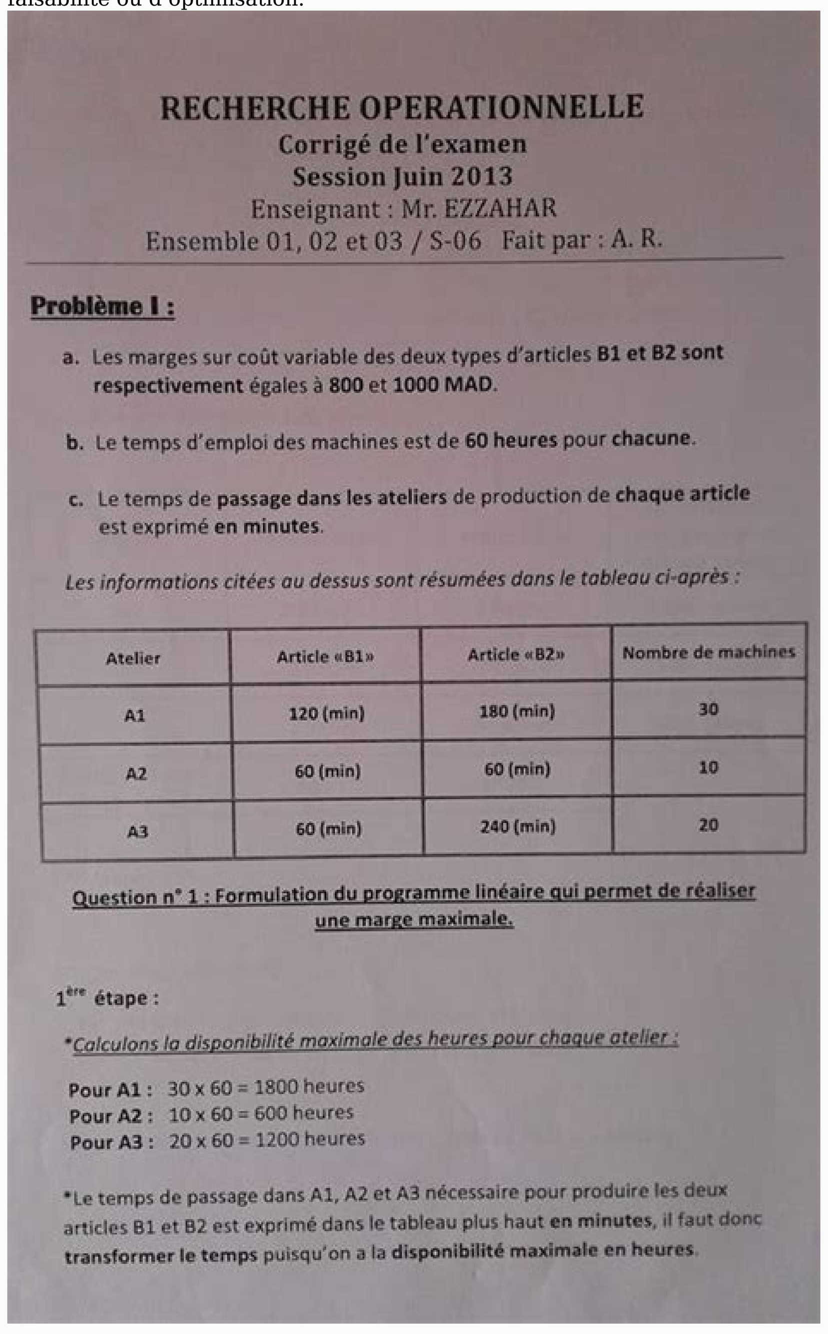
Recherche opérationnelle cours s5 pdf

Télécharger gratuitement TD, QCM, exercices et examens corrigés de Recherche opérationnelle PDF S5. Bachelor / Licence Informatique SMI (3ème année L3). Pour les cours, résumé, livres… vous trouverez les liens au bout de cette page. Tout en PDF/PPT, Tout est gratuit.Exercices & Examens de Recherche opérationnelleNOTE: N’oubliez pas de voir le cours de Recherche opérationnelle. Liens dans la section ci-dessous.Pour télécharger le cours complet de Recherche opérationnelle, Cliquez sur le/les liens ci-dessous.Cours complet de Recherche opérationnelleNOTE: N’oubliez pas de voir les autres Unités d’enseignements (matières/modules) d’Informatique. Liens dans la section ci-dessous.Tourner à la page principale d’Informatique pour voir la totalité des modules (cours, résumés, formation, exercices, td, examens, qcm, livres).Ou visiter directement les exercices des autres modules de la filière informatique à partir de ces liens ci-dessous: RECHERCHE OPERATIONNELLE 0. Introduction. Ce cours a été enseigné jusqu’en 2002, en année de licence, à la MIAGE de NANCY. L’objectif principal de ce cours est d’acquérir une connaissance approfondie de certaines techniques considérées à l’heure acutelle comme des méthodes de base en Recherche Opérationnelle. Celles-ci se retrouvent en effet, sous des formes plus complexes, dans les analyses professionnelles de faisabilité ou d’optimisation.



RECHERCHE OPERATIONNELLE
Corrigé de l’examen
Session Juin 2013
Enseignant : Mr. EZZAHAR
Ensemble 01, 02 et 03 / S-06 Fait par : A. R.

Problème I :

a. Les marges sur coût variable des deux types d’articles **B1 et B2 sont respectivement** égales à **800** et **1000 MAD**.

b. Le temps d’emploi des machines est de **60 heures** pour **chacune**.

c. Le temps de **passage dans les ateliers** de production de **chaque article** est exprimé **en minutes**.

Les informations citées au dessus sont résumées dans le tableau ci-après :

Atelier	Article «B1»	Article «B2»	Nombre de machines
A1	120 (min)	180 (min)	30
A2	60 (min)	60 (min)	10
A3	60 (min)	240 (min)	20

Question n° 1 : Formulation du programme linéaire qui permet de réaliser une marge maximale.

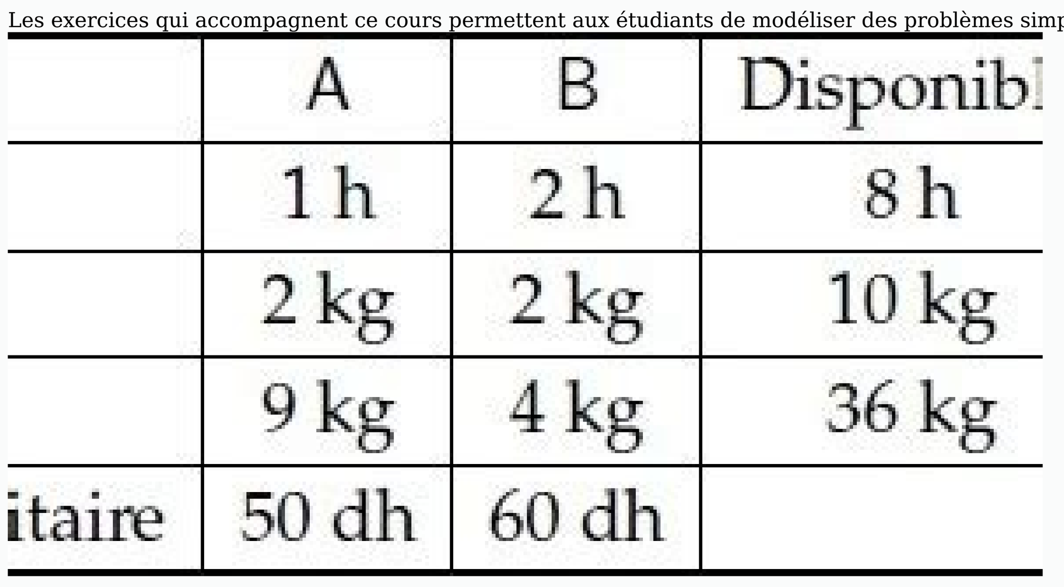
$1^{ère}$ étape :

Calculons la disponibilité maximale des heures pour chaque atelier :

Pour A1 : 30 x 60 = 1800 heures
Pour A2 : 10 x 60 = 600 heures
Pour A3 : 20 x 60 = 1200 heures

*Le temps de passage dans A1, A2 et A3 nécessaire pour produire les deux articles B1 et B2 est exprimé dans le tableau plus haut **en minutes**, il faut donc **transformer le temps** puisqu’on a la **disponibilité maximale en heures**.

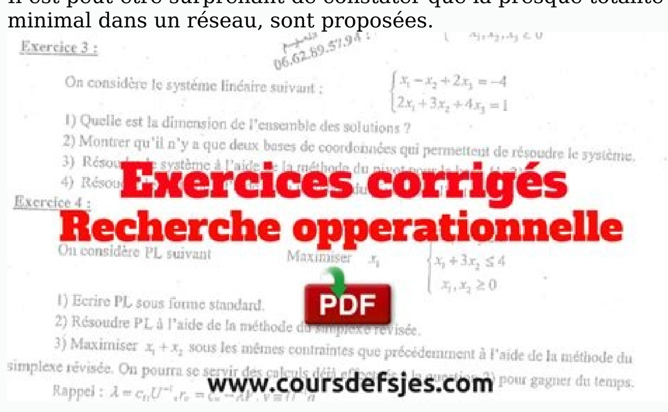
Les exercices qui accompagnent ce cours permettent aux étudiants de modéliser des problèmes simples en utilisant les techniques de la Recherche Opérationnelle.



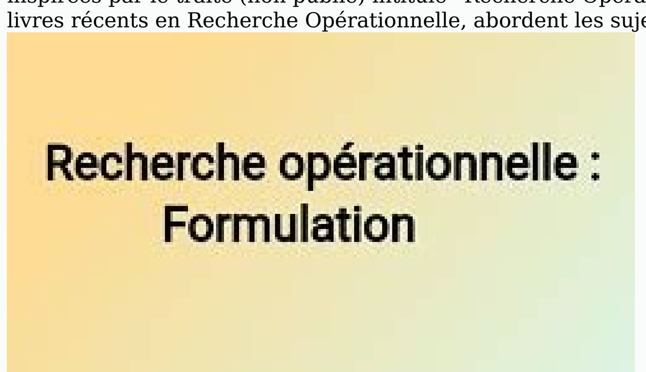
	A	B	Disponibl
	1 h	2 h	8 h
	2 kg	2 kg	10 kg
	9 kg	4 kg	36 kg
itaire	50 dh	60 dh	

Ces exercices ne tiennent évidemment pas compte de tous les paramètres d’une véritable analyse professionnelle.

Ils sont simplifiés volontairement et sont choisis suivant l’orientation des étudiants, en majorité dans le domaine de la gestion; mais les techniques utilisées s’appliquent également à la modélisation en ingéniérie et en sciences. Certaines méthodes de la Recherche Opérationnelle se démontrent - au niveau mathématique - assez facilement. L’algorithme du simplexe, par exemple, repose sur des arguments élémentaires de l’algèbre linéaire. D’autres méthodes présentées dans ce cours, par exemple celles de la programmation dynamique, sont des cas particuliers de développements analytiques et stochastiques plus avancés, qui, a` un niveau général, ne sont plus a` la portée d’un étudiant en licence (même en mathématiques). Une justification élémentaire de certains cas particuliers est cependant faisable et donne lieu a` quelques applications intéressantes. D’autres méthodes encore sont purement heuristiques, comme la méthode par séparation et évaluation (branch and bound) en programmation linéaire à valeurs entières. Il est peut-être surprenant de constater que la presque totalité des problèmes d’optimisationsénoncés dans ce cours (hormis ceux relevant de la programmation dynamique) peuvent être résolus à l’aide d’un algorithme de programmation linéaire. Cependant, des solutions spécifiques, par exemple au problème du flot maximal ou du flot de couˆt total minimal dans un réseau, sont proposées.



Elles sont souvent développées à partir d’un programme linéaire adapté au problème spécifique et font partie des méthodes standards dans la littérature récente. Ainsi la question du flot de couˆt total minimal permet de présenter quelques éléments du simplexe des réseaux. Le cours est divisé en cinq chapitres. Les deux premiers traitent de la programmation linéaire. Ils contiennent : - une introduction a` la technique du simplexe, une méthode standard pour la résolution d’un programme linéaire; - la dualité; - la méthode des coupes et la méthode par séparation et évaluation pour la résolution de programmes linéaires qui imposent a` certaines variables d’être à valeurs entières ou même binaires; - l’analyse post-optimale, qui détermine la variation de la solution en fonction d’un changement des valeurs des paramètres du programme. Le troisième chapitre traite de la théorie des jeux à deux personnes et a` somme zéro et s’appuie fortement sur les deux chapitres précédents. Au quatrième chapitre nous exposons deux problèmes d’optimisation de flots dans des réseaux: le flot maximal (avec sa coupe minimale) et le flot à couˆt total minimal. Le dernier chapitre est une introduction aux méthodes élémentaires de la programmation dyamique. Ce domaine est très vaste et de nombreux aspects ne sont pas étudiés, en particulier l’aspect stochastique en cas d’incertitudes sur les paramètres. Les algorithmes d’optimisation liés aux graphes (chemin de longueur maximale ou minimale par exemple) ne figurent pas dans ce cours parce qu’ils sont traités dans les cours d’outils conceptuels et d’algorithmique également enseignés à la Miage de Nancy. Les versions initiales de ce cours, ont été largement inspirées par le traité (non publié) intitulé ”Recherche Opérationnelle” de Francis Conrad, Professeur a` l’Université Henri Poincaré Nancy 1. Nous le remercions d’avoir mis ce texte à notre disposition. Nous n’avons pas connaissance d’un livre ou d’une monographie dont la présentation correspond a` ses notes, mais nous avons pu constater que les livres récents en Recherche Opérationnelle, abordent les sujets traités dans ce cours. Une liste, certainement incomplète, d’ouvrages récents est donnée ci-dessous. Bibliographie : C. Guéret, C. Prins, M. Sevaux, Programmation linéaire, Eyrolles, 2000. R. Favre, B.



Lemaire, C. Picouleau, Précis de recherche opérationnelle, 5ème éd., Dunod, 2000. Y. Nobert, R. Ouellet, R. Parent, La recherche opérationnelle, Ga¨etan Morin, 1995. J. F. Phélizon, Méthodes et modèles de la recherche opérationnelle, Economica, 1998. J. F. Maurras, Programmation linéaire, compléxité, Springer, 2002. D. Alevra, M. Padberg, Linear optimization and extensions : problems and solutions, Springer, 2001. V.K. Balakrishnan, Network optimization, Chapman and Hall, 1995. G.B. Dantzig, M.N. Thapa, Linear programming, Springer, 1997. H.A. Eiselt, C.L. Sandblom, Integer programming and network models, Springer 2000. B. Korte, J. Vygen, Combinatorial optimization, 2nd ed., Springer, 2002. G. Sierksma, Linear and integer programming, Marcel Dekker, 2001. R.J. Vanderbei, Linear programming foundations and extensions, Kluwer, 2001. W. Domschke, A. Drexl, Einfu¨hrung in Operations Research, 3. Auflage, Springer, 1995. W. Domschke, A. Drexl, B. Schildt, A. Scholl, S.
Voss, Ubungsbuch Opera-¨ tions Research, Springer, 1995. H.J. Zimmermann, Operations Research Methoden und Modelle. 2. Auflage., Vieweg, 1992. I. Programmation linéaire : Algorithme du simplexe Nous commen¸cons par quelques exemples qui peuvent être résolus en termes de programmes linéaires. I.1 Exemples. a) Problème du mélange La table ci-dessous donne la composition et le couˆt de 9 alliages standards de plomb, zinc et étain. Alliage 1 2 3 4 5 6 7 8 9 plomb (%) 20 50 30 30 30 60 40 10 10 zinc (%) 30 40 20 40 30 30 50 30 10 étain (%) 50 10 50 30 40 10 10 60 80 couˆt unitaire 7.3 6.9 7.3 7.5 7.6 6.0 5.8 4.3 4.1 Le but est de trouver un mélange des 9 alliages qui permet de fabriquer a` coût minimal un alliage contenant .
30 % de plomb ; . 30 % de zinc ; . 40 % d’étain. Remarquons que l’alliage 5 a la bonne composition ; son couˆt unitaire est de 7.6. Le mélange, à parts égaux, des alliages 6, 7, 8 et 9 donne la composition souhaitée aussi : . plomb : (60 + 40 + 10 + 10) = 30 % . zinc : (30 + 50 + 30 + 10) = 30 % . étain : (10 + 10 + 60 + 80) = 40 % couˆt unitaire : Procédons de fac¸on plus systématique pour obtenir le mélange de couˆt minimal : xj partie de l’alliage j dans le mélange recherché (j=1,,9), (1) Le couˆt unitaire du mélange recherché est le minimum de la fonction z, définie par z(x1,x2, ,x9) = 7.3 x1 + 6.9 x2 + + 4.1 x9 sous les contraintes : 30% de plomb : 0.2 x1 + 0.5 x2 + + 0.1 x9 = 0.3 (2) (2) 30% de zinc : 0.3 x1 + 0.4 x2 + + 0.1 x9 = 0.3 (3) 40% d’étain : 0.5 x1 + 0.1 x2 + + 0.8 x9 = 0.4 (4) Le programme linéaire consiste à minimiser z(x1,x2, ,x9) sous les contraintes (1)-(4). La fonction z est souvent appelée la fonction - objectif (ou fonction objective).
Remarque : (1)-(4) est un système linéaire de quatre équations et 9 inconnus. Le problème est donc de trouver, dans l’espace affine des solutions, la solution qui minimise la valeur de z(x1, ,x9). b) Problème de transport Supposons que 250 (resp. 450) containers sont disponibles au dépôt D1 (resp. au dépôt D2) et que les magasins A,B et C ont commandés 200 containers chacun. Les couˆts de transport par containers sont les suivants : magasin A B C dépôt D1 3.4 2.2 2.9 dépôt D2 3.4 2.4 2.5 Le but est de minimiser le couˆt total de transport des containers des dépôts vers les magasins en respectant les disponibilités et les demandes. xj nombre de containers transportés du dépôt i vers le magasin j (i = 1,2, j = A,B,C) Le programme linéaire s’énonce de la fac¸on suivante : minimiser z(xij, i = 1,2, j = A,B,C) = 3.4 x1A + 2.2 x1B + 2.9 x1C + 3.4 x2A + 2.4 x2B + 2.5 x2C sous les contraintes - de disponibilité : x1A + x1B + x1C ≤ 250 x2A + x2B + x2C ≤ 450 - de demande : x1A + x2A = 200 x1B + B2B = 200 x1C + x2C = 200 xij ≥ 0, a` valeurs entières La forme générale du problème de transport s’énonce comme suite : p dépôts D1,D2, ,Dp q magasins M1,M2, ,Mq ai quantité disponible au dépôt Di i = 1, ,p bj quantité demandée par le magasin Mj j = 1, ,q fij couˆt unitaire de transport de Di vers Mj xij quantité transportée de Di vers Mj programme linéaire : minimiser sous les contraintes (dépôts) (magasins) xij ≥ 0 (éventuellement à valeurs entières) Remarque : 1) Le choix du signe d’égalité ou d’inégalité dans les contraintes concernant les dépôts et les magasins dépend du contexte concret. 2) (quantité disponible ≥ quantité demandée) I.2. Solution graphique pour programmes linéaires à deux variables.
Exemple : maximiser z(x1,x2) = 4 x1 + 3 x2 sous les contraintes 3 x1 + 4x2 ≤ 12 7 x1 + 2 x2 ≤ 14, x1,x2 ≥ 0 L’intersection des demiplans déterminés par les droites qui correspondent aux contraintes représente l’ensemble des solutions qui satisfont aux contraintes (domaine hachuré). La direction qui correspond a` la fonction-objectif est donnée par z(x1,x2) = 4 x1 + 3 x2 = constante solution graphique : (ligne coupée −−−). On obtient la solution optimale en effectuant une translation parallèle de cette direction du haut vers le bas jusqu’a` ce que le domaine hachuré est atteint. Le point x est un point extremal de ce domaine. La solution graphique n’est évidemment applicable qu’aux programmes avec trois variables au plus. I.3. La méthode du simplexe. I.3.1.Forme générale d’un programme linéaire. En forme générale un programme linéaire se lit de fa¸con suivante : maximiser sous les contraintes (P) En forme vectorielle (P) se lit : maximiser z(x) = cx sous les contraintes Ax ≤ b, x ≥ 0, ou` on a posé : x21
1 2 n b2 x b1 x =... , c = (c ,c , ,c), b = ... xn bm a a a1n a2111 a2212 a2n
A = (aij, i = 1, ,m, j = 1, ,m) = m1 m2 a a amn On dit que (P) est sous forme canonique. Autre forme : (P =) maximiser z(x) = cx sous les contraintes Ax = b, x ≥ 0. On dit que (P =) est sous forme standard. Proposition
3.1.Chaque programme linéaire en forme standard s’écrit en forme canonique et inversement. Preuve. a) Ax ≤ b, x ≥ 0 (forme canonique) ou` ei : variable d’écart. Soit la matrice d’identité d’ordre m, le vecteur d’écart. Alors 0 (forme standard). Posons m−fois b) Ax = b (forme standard) (forme canonique). Exercice : Ecrire le programme linéaire suivant sous forme canonique : maximiser z(x) = cx sous les contraintes Ax ≤ b, x2,x3, ,xn ≥ 0. Indication : . Remplacer x1 par 0. I.3.2. Solutions de base réalisables. Retournons a` l’exemple de I.2. Sous forme standard, il se lit : maximiser z(x1,x2,x3,x4) = 4x1 + 3x2 sous les contraintes 3x1 + 4x2 + x3 = 12 7x1 + 2x2 + x4 = 14, x1,x2,x3,x4 ≥ 0 On pose On voit facilement que x = x2 = 0 satisfait aux contraintes. On dit que x est x1 0 x3 12 x4 14 solution de base réalisable, la base étant J = {3,4} (troisième et quatrième composante de x), les variables bases étant x3 = 12 et x4 = 14. Remarquons que x n’est pas optimal : z(x) = cx = 0.
La procédure d’optimisation se fait par une suite de tableaux, appelés tableaux du simplexe. Le premier résume les données et les variables de base. La dernière ligne est égale à −c. Premier tableau du simplexe c 4 3 0 0 cj variables de base x1 x2 x3 x4 0 12 3 4 1 0 0 14 7 2 0 1 z(x) 0 -4 -3 0 0 Retour au cas général : (P =) maximiser z(x) = cx sous Ax = b, x ≥ 0 ou` une matrice m × n. On peut supposer sans restriction de généralité que rang A = m < n. Rappel : Rang A = nombre (maximal) de lignes linéairement indépendantes de A = nombre (maximal) de colonnes linéairement indépendantes de A. En fait, si Rang A < m il y a des contraintes redondantes qui peuvent être supprimées exemple : 5x1 + 5x2 ≤ 10, 10x1 + 10x2 ≤ 20, Rang . Si m = n, la solution de Ax = b est unique et donnée par x = A−1b. Il n’y a donc rien a` maximiser. Définition. Posons A = (a1,a2, ,an) (aj est la j−ème colonne de A), et, pour J AJ = {aj, j ∈ J}. et si Rang AJ = Rang . Soit J (j /∈ J). On note xJ = (xj, j ∈ J). tel que xj = 0 pour j /∈ J et tel que Ax = AJxJ = b (c’est-à-dire que xJ satisfait aux contraintes). Exemple (suite) : Posons Alors Rang A = 2 = m, n = 4 , Rang AJ = 2 . variable de base : x3 = 12, x4 = 14 . variable hors base : x1 = x2 = 0 . solution de base réalisable : b. Remarque : L’avantage de passer de la forme canonique a` la forme standard est qu’on obtient immédiatement une solution de base réalisable qui sert de solution de départ pour l’algorithme du simplexe. Les variables de base sont les variables d’écart.
I.3.3. Changement de base réalisable. Soit x une solution de base réalisable de (P =).
On a alors z(x) = cx = cJxJ ou` cJ = (cj, j ∈ J). Le but est de trouver une autre base réalisable, notée J, et une solution de base réalisable associée, notée x, tel que z(x) > z(x) (x est meilleur que x). La méthode du simplexe consiste à remplacer une variable de base, notée xr, par une variable hors base, notée xk. On dit que xr est variable sortante de la base = 0, xk est variable entrante dans la base J : xk = 0 −→ xk > 0.
On aura donc. Question : Comment choisir r et k ? Réponse : .
Choisir r tel que 0 (c’est-à-dire choisir r tel que toutes les contraintes sont satisfaites par x). . Choisir k tel que z(x) > z(x) (augmentation de la valeur de z). Il faut maintenant des formules pour le choix de r et de k. Ces formules sont développées dans la suite. 1) Choix de r. Notons bi (i ∈ J) les colonnes de AJ. Dans l’exemple on a Comme Rang AJ = m, toutes les colonnes de A s’écrivent comme combinaison linéaire de b1,b2, ,bm ; en particulier (1) Il faut supposer. Remarquons que, graˆce à la forme particulière des b1 et b2 dans l’exemple, les coefficients aik ci-dessus (i = 1,2 ; k ∈ {1,2,3,4}) sont des éléments de A.
(2) Comme x est solution de base réalisable, on a . On pose = 0 pour i /∈ J. x est alors solution de base réalisable de (P =) par rapport a` 0. Il s’ensuit de (2) que Il faut donc choisir r tel que
. (3) 2) Choix de . Soit x une solution de base réalisable par rapport a` la base J. Alors Notons.
Il faut donc choisir k tel que zk − ck < 0 ou encore et il existe i ∈ J tel que. (4) Retour a` l’exemple : J = {3,4}, c3 = c4 = 0, et donc zj = 0 pour j = 1,2,3,4. Donc 4. Donc k = 1. Choix de = 2. Donc r = 2. x4 sort de la base, et x1 entre en base. Nouvelle base J = {3,1}. D’après (2) on a = 2. Donc 2 = 8.
Le passage de J = {3,4} a` J = {3,1} a donc permis d’augmenter la valeur de z à 8, mais ce n’est pas encore la valeur maximale. 3) Calcul du nouveau tableau. Reprenons la transformation obtenue en 1) de x −→ x qui a permis d’augmenter la valeur de z. Puisque la valeur n’est pas nécessairement maximale, il est nécessaire, en général, de répéter les démarches 1) et 2) plusieurs fois pour arriver a` une solution de base réalisable qui est un point maximal de z. Il nous faut donc un deuxième tableau ou xJ est remplacé par xJ et qui admet la même interprétation que le premier. Il faut appliquer la même transformation linéaire aux colonnes de A qui a permis de passer devoir (2) 1, ,m,j = 1, ,n) est remplacé par A = (aij, i = 1, ,m,j = 1, ,n), suivant les formules : On aura alors , (6) ou`, sont les nouvelles variables de base. La dernière ligne du nouveau tableau se calcule de la même fa¸con :
. (7) On a en fait parce que . En appliquant ceci a` notre exemple nous obtenons le deuxième tableau du simplexe par rapport a` la base. Deuxième tableau du simplexe c 4 3 0 0 cj variables de base x1 x2 x3 x4 0 6 0 22/7 1 -3/7 4 2 1 2/7 0 1/7 z(x) 8 0 -13/7 0 4/7 Comme on a déjà vu e partie 2) la valeur de z est passée de zéro à 8, mais il y a encore des valeurs négatives en dernière ligne du tableau. Un autre changement de base doit être effectué en appliquant les formules (3) et (4) remplacé par J : , r = 1, x3 sort de base. est la nouvelle base. Remarquons que, d’après (6), le deuxième tableau s’interprète de même fa¸con que le premier : avec solution de base réalisable Notons aussi que la structure des colonnes qu correspondent aux variables de base (x1 et x3) est la même que celle des colonnes qui correspondent aux variables de base du tableau de départ. Ceci permet de procéder de même fa¸con pour passer au troisième tableau, en appliquant les formules (2) a` (7).
La valeur de z passe alors de 8 a` 127/11, et cette valeur est maximale parce que toutes les valeurs de la dernière ligne sont non négatives. La solution optimale est donc ˜ , ce qui confirme le résultat obtenu en I.2. Elle est unique parce qu’aucun changement de base n’est plus possible. c 4 3 0 0 cj variables de base x1 x2 x3 x4 3 4 21/11 16/11 0 1 1 0 7/22 -1/11 -3/22 2/11 z(x˜) 127/11 0 0 13/22 7/22 Troisième tableau du simplexe I.4. Existence et unicité d’une solution optimale. Dans cette section nous résumons ce qui a été dit en I.3, et nous étudions les divers cas qui peuvent se produire en résolvant un programme linéaire à l’aide du simplexe. Après avoir écrit le premier tableau du simplexe et ajouté zj = −cj en dernière ligne, trois cas sont possibles : (i) zj −cj ≥ 0 pour j = 1, ,n.
La solution de base réalisable x est déjà optimale.
Elle est solution optimale unique ou non unique (voir les exemples ci-dessous).
(ii) Il existe j ∈ {1, ,n} tel que zj − cj < 0 et aij ≤ 0 pour tous i ∈ J.
Le domaine des solutions réalisables est alors non-borné et le problème a été mal posé, parce que max z(x) = +∞.
Exemple : maximiser z(x1,x2) = x1 + x2 sous les contraintes −x1 + x2 ≤ 1, x1,x2 ≥ 0. (iii) Le cas habituel : il existe j ∈ {1, ,n} tel que zj −cj < 0, et il existe i ∈ J tel que aij > 0. Le changement de base tel qu’il a été décrit en partie I.3 ci-dessus est alors possible et doit être effectué, éventuellement plusieurs fois, jusqu’a` ce qu’on arrive au cas (i) ci-

dessus. Est-ce possible que le simplexe n'aboutit pas, c'est-à-dire que, dans le cas (iii), on arrive dans un loop ?
La réponse est oui, et de tels exemples ont été construits. Mais en pratique ils sont très rares. Exemples ou` la solution optimale n'est pas unique : 1) maximiser z(x1,x2) = 2x1 − x2 sous les contraintes 2x1 −x2 ≤ 4, −x1 +x2 ≤ 1, −x1 +2x2 ≤ 4, 2x1 +x2 ≤ 12, x1,x2 ≥ 0. La droite qui correspond a` la première contrainte donne exactement la direction indiquée par la fonction a` maximiser. La partie de cette droite qui appartient au bord de l'ensemble des solutions réalisables est donc l'ensemble des solutions réalisables optimales. Remarquons que le simplexe nous donne seulement les points extrémaux du segment des solutions optimales. Ce sont précisément les solutions de base réalisable optimales. Les autres solutions réalisables optimales sont les combinaisons convexes de ces solutions de base optimales.
Il est toujours vrai que les solutions de base optimales sont des points extrémaux de l'ensemble convexe des solutions optimales (voir Théorème 5.2). 2) maximiser z(x1,x2,x3) = x1 − 2x2 + x3, sous les contraintes x1 − 2x2 + x3 ≤ 10, 2x1 − 3x2 − x3 ≤ 6, x1,x2,x3 ≥ 0. Premier tableau c 1 -2 1 0 0 c variables de base x1 x2 x3 x4 x5 0 10 1 -2 1 1 0 0 6 2 -3 -1 0 1 z(x) 0 -1 2 -1 0 0 Deuxième tableau c 1 -2 1 0 0 c variables de base x1 x2 x3 x4 x5 0 7 0 -1/2 3/2 1 -1/2 1 3 1 -3/2 -1/2 0 1/2 z(x) 3 0 1/2 -3/2 0 1/2 Troisième tableau c 1 -2 1 0 0 c variables de base x1 x2 x3 x4 x5 1 14/3 0 -1/3 1 2/3 -1/3 1 16/3 1 -5/3 0 1/3 1/3 z(x) 10 0 0 0 1 0 Le troisième tableau nous fournit la solution de base optimale. En effectuant une étape supplémentaire du simplexe (k = 5, r = 2) on arrive a` une autre solution de base optimale x = t(0, 0, 10) : Quatrième tableau c 1 -2 1 0 0 c variables de base x1 x2 x3 x4 x5 1 10 1 -2 1 1 0 0 16 3 -5 0 1 1 z(x) 10 0 0 0 1 0 Toutes les combinaisons convexes de ces deux solutions optimales sont donc des solutions optimales aussi.
Indépendant du simplexe on peut vérifier directement que les vecteurs suivants sont des solutions optimales eux aussi : pour tous ϕ > 0 On a évidemment z(x) = z(x) = z(xϕ) = 10 pour tous ϕ > 0. Terminons ce paragraphe par des preuves mathématiques de quelques énoncés concernant le simplexe. Retournons a` la forme standard : (P =) maximiser z(x) = cx sous les contraintes Ax = b, x ≥ 0 Théorème 4.1.Soit une solution de base réalisable de (P =) par rapport à une base = rang A. Soit . Si zj − cj ≥ 0 pour tous j = 1, ,n, x est solution de base réalisable optimale. Preuve.
Soit A = (a1,a2, ,an), AJ = (aj, j ∈ J) = (bi,i = 1, ,m). Soit x0 une autre solution réalisable de (P =) par rapport a` la base J. A démontrer : Donc Théorème 4.2.Soit une solution de base réalisable de (P =) par rapport à une base = rang A. Soit. Supposons que aij ≤ 0 pour tous i ∈ J et pour tous j ∈ {1, ,n} tel que zj −cj < 0. Alors l'ensemble est solution réalisableest un ensemble non borné. Preuve. On reprend les notations du Théorème 4.1. Comme aik ≤ 0 par hypothèse,.
On pose Donc xϕ définie par comme ci-dessus et par = 0 pour i /∈ J ∪{k} est une solution réalisable de (P =), mais pas nécessairement une solution de base réalisable. puisque zk − ck < 0. En faisant tendre on voit que z(xϕ) n'est pas borné. I.5. Structure géométrique des solutions réalisables. Pour les exemples qu'on a résolu graphiquement, la ou les solutions optimales se trouvent toujours sur le bord de l'ensemble convexe des solutions réalisables. Si la solution optimale était unique il s'agissait d'un point extrémal. Ceci n'est pas un hasard ; c'est en fait toujours le cas, comme le montrent les énoncés qui suivent. Commen¸cons par quelques définitions : . Un ensemble E ⊂ Rn est convexe si x1,x2 ∈ E, 0 < λ < 1 =⇒ λx1 + (1 − λ)x2 ∈ E. .
x est point extrémal de l'ensemble convexe E si l'énoncé suivant vaut il existe x1,x2 ∈ E et λ ∈ [0,1] tel que x = λx1 + (1 − λ)x2 =⇒ x1 = x2 = x Rappelons la forme canonique d'un programme linéaire : maximiser z(x) = cx (P) sous les contraintes Ax ≤ b, x ≥ 0, ou` on suppose que la matrice A est une matrice m × n de rang égal à m(< n). La forme standard est alors donnée par maximiser z(x) = cx (P =) sous les contraintes (Lemme 5.1.Soit Γ (resp.
Σ) l'ensemble des solutions réalisables de resp. de (P =)) : Γ et Σ sont des ensembles convexes et fermés. Esquisse de la preuve. A démontrer a) x1,x2 ∈ Γ, 0 < λ < 1 =⇒ xdef= λx1 + (1 − λ)x2 ≥ 0 et Ax ≤ b b) et pareil pour Σ. Rappelons que la méthode du simplexe se sert uniquement de solutions de base de (P =). Le théorème suivant nous dit que ce sont en fait les points extrémaux de M. Théorème 5.2 est solution de base réalisable de (P =) si et seulement si est point extrémal de Σ. Remarque. Il s'ensuit que chaque point extrémal de Γ est solution de base réalisable de (P=).
Les exemples ou` la solution n'est pas unique montrent que les solutions optimales se trouvent sur le bord. Le théorème suivant confirme ce fait. Théorème 5.3.Chaque solution optimale de (P) (resp. de (P =)) fait partie du bord de Γ (resp. de Σ).
Preuve. Soit u un point intérieur de Σ, tel que z(u) = cu = max{z(v), v ∈ Σ}. Il t existe alors ε > 0 tel que {v ∈ Rn+m ; |u−v| < ε} ⊂ Σ, en particulier ˜ . Mais), ce qui est une contradiction a` l'hypothèse que u est un point maximal de z. Est-ce possible qu'il existe des solutions optimales, mais qu'il n'existe pas de solution de base réalisable optimale ? Le simplexe, ne traitant que de solutions de base ne permettrait alors pas de trouver une solution optimale. Le théorème suivant dit que ceci n'est pas le cas. Théorème 5.4.Si (P =) possède une solution optimale, alors (P =) possède une solution de base réalisable optimale. Conclusion. S'il existe des solutions optimales, elles se trouvent sur le bord de Σ par le théorème 5.3., et il existe toujours une solution de base réalisable optimale, qui est un point extrémal de Σ d'après le Théorème 5.2. et qu'on trouve a` l'aide du simplexe. I.6. Initialisation de la méthode du simplexe. Jusqu'a` présent on a supposé que le programme linéaire de départ est en forme canonique. Dans ce cas une solution réalisable initiale est facile a` obtenir, et l'algorithme du simplexe s'applique directement a` la forme standard associée. Si le programme de départ est déjà en forme standard, une solution de base réalisable qui sert de solution initiale n'est pas toujours immédiate comme le montre l'exemple suivant : Maximiser z(x) = −x1 + 2x2 − 2x3 sous les contraintes x1 + x2 − x3 = 3, −x1 + 3x2 = −4 et x1,x2,x3 ≥ 0. Nous présentons dans cette section deux techniques qui permettent de trouver une solution de base réalisable qui sert de solution initiale pour le simplexe. I.6.1. La MéthodeM. Supposons que le programme linéaire à résoudre est donné par maximiser z(x) = cx, (P =) sous les contraintes Ax = b, x ≥ 0 (A matrice m × n, Rang A = m < n). En ajoutant des variables d'écarts (de signe positif ou négatif) il est toujours possible de mettre un programme linéaire en la forme (P =) ci-dessus avec. S'il n'est pas évident de trouver une solution de base réalisable qui peut servir de solution initiale pour le simplexe, on ajoute aux contraintes des variables artificielles yi ≥ 0 : Ax = b est remplacé par. Les nouvelles contraintes ne sont bien sur pas équivalentes aux contraintes initiales. On pénalise les yi > 0 en remplac¸ant la fonction-objectif z(x) par est une valeur positive très élevée. On choisira les yi = bi, i = 1, ,m, comme solution de base réalisable qui sert de solution initiale. Diminuant considérablement la valeur de z, elles vont dispara^ıtre de la base au cours des étapes du simplexe. Etant finalement des variables hors base, leur valeur sera égale à zéro et le programme linéaire résolu sera quand même le programme (P =). De fac¸on plus précise, on part du programme linéaire suivant : (PM) maximiser sous les contraintes. S'il y a des composantes de b qui sont négatives, on multiplie par (−1) les contraintes initiales concernées. La solution de base réalisable initiale est y = b. On exprime zen termes de variables hors base en substituant les yi : 1er cas. Le tableau final du simplexe appliqué à (PM) comporte un ou plusieurs yi > 0. Le programme inital (P =) n'a alors pas de solutions de base réalisables et donc pas de solutions réalisables du tout. En effet : Si (P =) possède une solution de base réalisable (c'est-à-dire tel que Ax = b), le M >> 0 aurait fait sortir tous les yi de la base en les posant égal à zéro. 2nd cas. Le tableau final ne comporte pas de yi en variables de base. La solution optimale de (PM) est alors solution de (P =), parce que yi = 0 (i = 1,2, ,m). Il reste alors à se persuader qu'il s'agit de la (ou d'une) solution de base réalisable optimale de (P) (voir Théorème 4.1.). Exemple (suite) : Les nouvelles contraintes se lisentx1 + 3x2 +y2 = −4, x1,x2,x3,y1,y2 ≥ 0. Après avoir substitué y1 dans la fonctionobjectif, on obtient. Premier tableau c -1+2M 2-2M c variables de base x1 x2 0 y1 3 1 1 0 y2 4 1 -3 -7M -7M 1-2M -2+2M -1 0 Deuxième tableau c -1+2M 2-2M cj variables de base x1 x2 -1+2M x1 3 1 1 0 y2 1 0 -4 -3+4 -7M -3-M 0 -1 1 Troisième tableau c -1+2M 2-2M c variables de base x1 x2 -1+2M x1 4 1 -3 -2-M x3 1 0 -4 -7M -6 0 9 0 1 0 La solution optimale unique est donnée par x1 = 4, x2 = 0, x3 = 1 et z(x1,x2,x3) = −6. I.6.2. Le programme auxiliaire. Supposons que le programme linéaire à résoudre soit donné par maximiser z(x) = cx, (P =) sous les contraintes Ax = b, x ≥ 0. La méthode du programme auxiliaire consiste a` faire tendre M ↑ +∞ dans (PM) et résoudre le programme qu'on obtient après le passage à la limite. Soit . Maximiser) revient a` maximiser z″(x,y) ou encore à maximiser lim z″(x,y) = m M→∞ . On résoud donc le programme auxilaire (PA), donné par =1 maximiser (PA) sous les contraintes. On applique le simplexe a` (PA) en partant de la solution de base réalisable y = b. 1er cas. max z″(y) < 0. Il existe yi > 0,i ∈ {1,2, ,m}. (P =) n'a donc pas de solutions réalisables. . Exemple. maximiser z(x1,x2,x3,x4) = x1 + x2 + x3 + x4 sous x1 + 2x2 + x3 = 2, x1 + x2 + 5x3 = 12, x1 + 2x2 + 6x3 + x4 = 13, x1,x2,x3,x4 ≥ 0 maximiser z″(x,y) = −y1 − y2 = −14 + 2x1 + 3x2 + 6x3 sous x1 + 2x2 + x3 + y1 = 2, x1 + x2 + 5x3 + y2 = 12, x1 + 2x2 + 6x3 + x4 = 13, x1,x2,x3,x4,y1,y2 ≥ 0. Premier tableau (x4 tient lieu de variable artificielle) c 2 3 6 0 0 0 c variables de base x1 x2 x3 x4 y1 y2 0 2 1 2 1∗ 0 1 0 0 12 1 1 5 0 0 1 0 13 1 2 6 1 0 0 -14 z″(x) -14 2 -3 -6 0 0 0 − Deuxième tableau c 2 3 6 0 0 0 c variables de base x1 x2 x3 x4 y1 y2 6 2 1 2 1 0 1 0 0 2 -4 -9 0 0 -5 1 0 1 -5 -10 0 1 -6 0 -14 z″(x) -2 4 9 0 0 6 0 La solution optimale est : x1 = 0, x2 = 0, x3 = 2, x4 = 1, y1 = 0 y2 = 0. La valeur maximale de z″(x,y) étant négative, le programme linéaire en question n'a pas de solution réalisable. 2nd cas. max z″(y) = 0. La solution optimale de (PA) peut servir de solution de base réalisable initiale pour l'algorithme de simplexe appliqué à (P =). Il faut alors exprimer la fonction-objectif de (P =) en termes des variables hors base et appliquer le simplexe à (P =). Exemple : maximiser z(x1,x2,x3,x4) = x1 + x2 + x3 + x4 sous 2x2 + x3 = 2 x1 + x2 + 5x3 = 12 x1 + 2x2 + 6x3 + x4 = 13, x1,x2,x3,x4 ≥ 0. maximiser sous (PA) z″(x,y) = −y1 − y2 = −14 + x1 + 3x2 + 6x3 2x2 + x3 + y1 = 2 x1 + x2 + 5x2 + y2 = 12 x1 + 2x2 + 6x3 + x4 = 13, x1,x2,x3,x4,y1,y2 ≥ 0 Premier tableau (x4 tient lieu de variable artificielle) c 1 3 6 0 0 0 c variables de base x1 x2 x3 x4 y1 y2 0 2 0 2 1∗ 0 1 0 0 12 1 1 5 0 0 1 0 13 1 2 6 1 0 0 -14 z″(x) -14 -1 -3 -6 0 0 0 Deuxième tableau c 1 3 6 0 0 0 c variables de base x1 x2 x3 0 1 0 6 2 0 2 1 0 2 1 -9 0 0 -5 1 0 1 1∗ -10 0 1 0 -6 6 0 0 -14 z″(x) -2 -1 9 0 Troisième tableau c 1 3 6 0 0 0 c variables de base x1 x3 x4 y1 y2 6 2 0 2 1 0 1 0 0 1 0 1 0 -1 1 1 1 1 1 -10 0 1 -6 0 -14 z″(x) -1 0 -1 0 1 0 0 Quatrième tableau c 1 3 6 0 0 0 c variables de base x1 x2 x3 2 -1 -2 6 = x3 0 0 0 1 3 = x2 1 0 1 0 -1 1 1 1 = x1 11 1 0 0 -9 0 4 1 10 1 -14 z″(x) 0 0 0 0 La solution optimale est : x1 = 11, x2 = 1, x3 = x4 = y1 = y2 = 0, z″(x,y) = 0. Cette solution sert de solution initiale pour le simplexe appliqué au programme de départ. Afin d'exprimer z(x1,x2,x3,x4) = x1 +x2 +x3 +x4 en termes de variables hors base, on lit le quatrième tableau de la fac¸on suivante : 0 = x3 + 2x4 1 = x2 − x4 11 = x1 − 9x4 On en déduit z(x1,x2,x3,x4) = x1 +x2 +x3 +x4 = 9x4 +12 (x4 étant variable hors base). Cinquième tableau c 0 0 0 9 c variables de base x1 x2 x3 x4 0 = x3 0 0 0 1 2∗ 0 = x2 1 0 1 0 -1 0 = x1 11 1 0 0 -9 12 z(x) 0 0 0 0 -9 Sixième tableau c 0 0 0 9 c variables de base x1 x2 x3 x4 9 = x4 0 0 0 0.5 1 0 = x2 1 0 1 0.5 0 0 = x1 11 1 0 4.5 0 12 z(x) 0 0 0 4.5 0 La solution optimale du programme de départ est alors x1 = 11, x2 = 1, x3 = x4 = 0. z(x1,x2,x3,x4) = 12. II. Programmation linéaire : Dualité, programma-tion en nombres entiers, analyse postoptimale. II.1. Le programme linéaire dual. Commen¸cons par une interprétation économique du programme dual. Exemple. Problème de la production. Notons : xj nombre d'unités du produit Pj fabriquées en entreprise I (j = 1,2, ,n) aij nombre d'unités de la matière première Mi utilisées pour la fabrication d'une unité de Pj cj bénéfice de l'entreprise I en vendant une unité de Pj. Le programme linéaire pour déterminer le plan de production qui permet de maximiser le bénéfice de l'entreprise I s'énonce comme suite : maximiser z(x) = cx sous les contraintes de disponibilité . En forme matricielle les contraintes se lisent : Ax ≤ b, x ≥ 0 ou` A = (aij ; i = 1,2, ,m, j = 1,2, ,n), b = t(b1,b2, ,bm). Supposons que l'entreprise II essaie de s'emparer du marché. Sous l'hypothèse d'un comportement économique de l'entreprise I, celle-ci est prête à céder les matières premières à un prix qui est au moins aussi élévé que le bénéfice qu'elle fera en vendant ses produits. Soit yi le prix que l'entreprise II devra payer pour une unité de Mi (i = 1,2, ,m). Les contraintes sont les suivantes : . En forme matricielle yA ≥ c, y ≥ 0 ou` y = (y1,y2, ,ym). L'entreprise II essaiera de minimiser le cou^t d'achat des matières premières : elle essaiera de minimiser . def Le programme linéaire pour l'entreprise II est donc de minimiser w(y) = yb sous les contraintes yA ≥ c, y ≥ 0.
Comme on verra il y a un lien mathématique étroit entre les deux programmes linéaires. Définition. Programme primal Programme dual maximiserminimiser w(y) = yb ( ) soussous Remarque. On a liens suivants : m = nombre de contraintes de (P) = nombre de variables de (D), n = nombre de variables de (P) = nombre de contraintes de (D). Si (P) contient deux contraintes, (D) contient deux variables et peut être résolu graphiquement quelque soit le nombre de variables de (P). Quelle est la forme de (D) si (P) n'est pas en forme canonique ? On détermine (D) en ramenant (P) a` la forme canonique, comme le montre l'exemple siuvant : Exemple :
maximiser z(x) = cx (P) . Par la définition ci-dessus le dual de (P) est donné par (D) minimiser sous (u,v) . En posant y = u − v, on obtient (D) minimiser w(y) = yb sous yA ≥ c, y sans restriction de signe . Le résultat général est le suivant. Pour comprendre précisement son message, il convient d'étudier sa preuve. Théorème 1.1.Les liens entre le programme primal et son dual sont les suivants : Primal Dual maximisation minimisation coefficient de z second membre des contraintes second membre des contraintes coefficient de w = sans contrainte de signe contrainte≤≥ variable ≥≤ 00 sans contrainte de signe = variablecontrainte≥ ≤ Preuve. On se donne (P) maximiser z(x,y) = cx + dy sous Ax + By ≤ a, Cx + Dy = b, x ≥ 0, (y) matrice r × s, B matrice r × (n − s) matrice (m − r) × s, D matrice (. On démontre que le dual de (P) est donné par : (D) minimiser w(u,v) = ua + vb sous uA + vC ≥ c, uB + vD = d, u ≥ 0, (v) maximiser sous! !!!!!!!sous CCA maxDDB z(DDx) = (c,d,yy12 −d)yy12 !!!!!!!!!!! ! x ⇐⇒ !!
−B x a − ≤ b !− !− −b Passons au dual : ! ⇐⇒ (avec !! !! u ≥ 0, (v) ! et b est lié à v dans (D) qui est une variable sans contrainte de signe. La variable y est sans contrainte de signe dans (P). Elle est facteur de B et D qui forment la contrainte égalité dans (D). Théorème 1.2.Le dual du dual est le primal. Preuve. Pour le dual de la définition ci-dessus on a ! . Le dual est donc donné par . La démonstration dans le cas plus général du Théorème 1.2 est analogue.! ! ! II.2. Le théorème de dualité.
Après avoir défini le dual d'un programme linéaire, nous étudions dans cette section les liens entre les solutions de programmes en dualité. Proposition 2.1. Soit (x,y) une solution réalisable de (P) et (u,v) une solution réalisable de (D). Alors : 1) z(x,y) ≤ w(u,v) 2) z(x,y) = w(u,v) =⇒ (x,y) et (u,v) sont des solutions optimales de (P) et (D). Preuve. 1) Dans cette chaine d'inégalités il est important de noter que x ≥ 0 et u ≥ 0.
2) ua+vb étant une borne supérieure pour z(x,y), l'égalité z(x,y) = ua+vb signifie que (x,y) est un point maximal de z. Raisonnement analogue pour w(u,v). Rappelons les trois cas qui peuvent se produire en appliquant le simplexe a` (P) : (i) il existe une (ou plusieurs) solutions optimales finies (ii) l'ensemble des solutions réalisables est non borné et max z(x,y) = +∞ (iii) (P) ne possède pas de solutions réalisables. Le même raisonnement s'applique a` (D). Le théorème suivant dit que trois cas seulement se produisent parmi les 9 cas possibles (voir tableau ci-dessous). (P) (D) (i) (ii) (iii) (i) (ii) (iii) a) × × × × b) b) c) × Théorème 2.2. (Théorème de dualité). Seuls les trois cas suivants peuvent se produire : a) (P) et (D) possèdent des solutions optimales et max z(x,y) = min w(u,v). b) (P) ou (D possède une solution réalisable, mais pas les deux. c) Ni (P) ni (D) possèdent des solutions réalisables. Preuve. Essentiellement à l'aide de la proposition 2.1. Pour la partie c) il suffit de donner un exemple ou` ni (P) ni (D) ont des solutions réalisables : (P) maximiser z(x1,x2) = x1 + x2 sous x1 − x2 = 1, x1 − x2 = −1, x1x2 ≥ 0 (D) minimiser w(y1,y2) = y1 − y2 sous y1 + y2 ≥ 1, y1 + y2 ≤ −1, (y1),(y2) Supposons maintenant que le cas a) du théorème précédent s'est réalisé. Quelles sont les liens entre les solutions optimales de (P) et de (D) ? Comment calculer l'une a` partir de l'autre ? La réponse se déduit facilement du théorème suivant. Théorème 2.3.Soient (x,y) resp. (u,v) des solutions réalisables de (P) resp. de (D). (x,y) et (u,v) sont alors des solutions optimales de (P) et de (D) si et seulement si les énoncés suivants valent : . Si une contrainte est satisfaite en tant qu'inégalité dans (P) resp. (D), alors la variable correspondante de (D) resp. de (P) est nulle.
. Si la valeur d'une variable restreinte (c'est-a`-dire une variable ≥ 0 ou une variable ≤ 0) dans l'un des programmes , alors la contrainte correspondante de l'autre programme est une égalité. Preuve. Soit ξ ≥ 0 tel que Ax+By +ξ = a, et soit η ≥ 0 tel que uA+vC −η = c (ξ,η sont des variables d'écart). Alors Donc z(x,y) − w(u,v) = −ηx − uξ ≤ 0 par construction. D'après le théorème 2.2 (x,y) et (u,v) sont optimales si et seulement si ηx + ξu = 0. Donc si et seulement si uiξi = 0 pour tout i = 1,2, ,r et ηjxj = 0 pour tout j = 1,2, ,s. L'assertion s'ensuit : contrainte = dans (variable = 0 dans (variable = 0 dans (contrainte = dans (contrainte = dans (variable = 0 dans (variable = 0 dans (contrainte = dans (Exemple.! maximiser z(x) = 4x1+ 5x2 ≤ ≥ sous 3x1 + x2 ≤ 1, x1 + 4x2 1, x1, x2 0 ! minimiser w(y) = y1 + y2 sous 3y1 + y2 ≥ 4, y1 + 4y2 ≥ 5, y1, y2 ≥ 0 solution optimale de (Comment en déduire la solution optimale de (D) ? D'après le Théorème 2.3 on a Donc y1 = y2 = 1 est solution optimale de (D). En effet Le lien entre les solutions du primal et le dual sont facilement visibles si on les résoud par le simplexe. En fait, les tableaux finaux se recouvrent partiellement comme le montre le schéma ci-après. Les variables d'écart ξ de (P) sont notées ξ1,ξ2, et les variables d'écart η de (D) sont notées η1,η2. (P) → → x1 x2 ξ1 ξ2 solution ↓ (D) ↓ ↓ ξ1 y1 → η1 η2 y1 y2 optimale de (D) -4/11 -1/11 1 0 1 ξ2 y2 x1 η1 -1/11 3/11 0 1 1 1 0 0 x2 η2 0 1 0 solution optimale de (P) 3/11 2/11 0 0 Les colonnes de x1 et x2 et la dernière colonne représentent le tableau final du simplexe appliqué au primal. La solution optimale du primal appara^ıt en dernière ligne : x1 = 3/11, x2 = 2/11 (le tableau est transposé par rapport a` la représentation habituelle : les lignes du tableau habituel sont les colonnes ici). Les lignes de y1 et y2 et la dernière ligne forment le tableau optimal du simplexe appliqué au dual. La solution optimale du dual appara^ıt en dernière colonne : y1 = y2 = 1. Il n'est donc pas nécessaire de résoudre (P) et (D). Il suffit de résoudre (P) ou (D), la solution optimale de l'autre se trouve dans le tableau optimal du premier, en dernière ligne. Il suffit alors d'identifier les variables : (i = 1,2) II.3. Programmation en nombres entiers Dans beaucoup de problèmes d'optimisation une solution a` valeurs entières est exigée. Par exemple dans le problème de la production, ou` l'on cherche le nombre optimal de pièces à fabriquer et ou` ce nombre est, pour des raisons pratiques, souvent entier. L'exemple suivant montre alors que ce nombre optimal entier n'est pas toujours l'entier le plus proche de la solution optimale si celle-ci est fractionnaire. Exemple.
maximiser 10x1 + 11x2, sous la contrainte 10x1 + 12x2 ≤ 59, x1,x2 ≥ 0 à valeurs entières. La solution optimale x est donnée par x = (x1,x2) = (5.9,0). En effet, x1 ≤ 5.9 − 1.2x2, et on obtient la solution optimale en posant x1 = 5.9, x2 = 0. Mais la = (6,0) n'est pas réalisable et la solution 0) n'est pas optimale parmi les solutions a` valeurs entières. La solution optimale a` valeurs entières est donnée par ˜x = (1,4). Les méthodes présentées ci-dessous pour trouver la solution optimale a` valeurs entières utilisent le simplexe à plusieures reprises. II.3.1. La méthode par séparation et évaluation (Branch and Bound). Exemple. maximiser z(x1,x2) = x1 + 4x2 (P)!!!!sous les contraintes 5x1 + 8x2 ≤ 40, −2x1 + 3x2 ≤ 9, x1, x2 ≥ 0 et à valeurs entières. Etapes : 1) Résoudre (P) a` l'aide du simplexe sans tenir compte de la contrainte x1,x2 à valeurs entières. Solution optimale : x = (x1,x2) = (1.55,4.03).z(x) = 17,67. Brancher par rapport a` x1 : x1 ≤ 1 ou x1 ≥ 2.
2) Ajouter la contrainte x1 ≤ 1 à (P) et résoudre ce programme à l'aide du simplexe, sans tenir compte de la contrainte x1,x2 à valeurs entières. Solution optimale x = (1,3.67), z(x) = 15.67. Brancher par rapport a` x2 : x2 ≤ 3 ou x2 ≥ 4. 3) Ajouter a` (P) les contraintes x1 ≤ 1 et x2 ≤ 3. La solution optimale de ce programme est à valeurs entières : x = (1,3), z(x) = 13. Borner a` 13 (c'est-à-dire ne pas poursuivre le branchement si on obtient des solutions optimales < 13). 4) Ajouter a` (P) les contraintes x1 ≤ 1 et x2 ≥ 4. Ce programme n'a pas de solutions réalisables. 5) Ajouter a` (P) la contrainte x1 ≥ 2. La solution optimale x = (2,3.75), z(x) = 17. Brancher par rapport a` x2 : x2 ≤ 3 ou x2 ≥ 4. 6) Ajouter a` (P) les contraintes x1 ≥ 2 et x2 ≤ 3. Solution optimale x = (3.2,3), z(x) = 15.2. Brancher par rapport a` x1 : x1 ≤ 3 ou x1 ≥ 4.
7) Ajouter a` (P) les contraintes 2 ≤ x1 ≤ 3 et x2 ≤ 3. Solution optimale x = (3,3), z(x) = 15. Borner a` 15. 8) Ajouter a` (P) les contraintes x1 ≥ 4 et x2 ≤ 3. Cas sans intérêt parce que z(x) = 14 < 15 pour la solution optimale x. 9) Ajouter a` (P) les contraintes x1 ≥ 2 et x2 ≥ 4. Pas de solutions réalisables. La solution optimale a` valeurs entières est donc celle trouvée en 7) : x1 = x2 = 3, z(x) = 15. Remarque : La méthode par séparation et évaluation n'intervient pas seulement en programmation linéaire, mais dans tous les problèmes d'optimisation, ou` on travaille avec des arbres de décision.
II.3.2. La méthode des coupes. Exemple : maximiser z(x1,x2) = 2x1 + x2 sous les contraintes x1 − 4x2 ≤ 0, 3x1 + 4x2 ≤ 15, x1,x2 ≥ 0 à valeurs entières. La méthode des coupes consiste à ajouter des contraintes supplémentaires qui permettent d'approcher les solutions réalisables à valeurs entières sans les écarter du domaine des solutions réalisables. Dans l'exemple ci-dessus une telle contrainte est donnée par x1 ≤ 3 (on coupe a` x1 = 3). Comment trouver de telles contraintes supplémentaires si le programme linéaire contient plus de deux variables ? Soit maximiser z(x) = cx sous les contraintes Ax = b, x 0, ou` A est une (P) ≥ matrice m × n. Posons, pour une base J (avec |J| = m) et une solution réalisable x, . Puisque AJ est régulier, xJ+(AJ)−1AJCxJC = (AJ)−1b. Posons : Adef= (AJ)−1AJC et ˜bdef= (AJ)−1b. On a donc . Soit [˜aij] la partie entière de ˜ la partie fractionnaire de ˜ Exemples : Il s'ensuit : (1) Soit x une solution réalisable de (P =) a` valeurs entières.
L'égalité (1) vaut pour x, le côté gauche étant a` valeurs entières. En plus, (2) Comme la partie gauche de (2) est à valeurs entières, . Comme (1) est valable pour toute solution réalisable, il s'ensuit que : Proposition 3.1.En ajoutant à (P =), la contrainte , (i ∈ J) on n'écarte pas de solutions réalisables a` valeurs entières de l'ensemble des solutions réalisables. Remarque. Comme l'exemple suivant montre, on arrive a` la solution optimale à valeurs entières en ajoutant, a` plusieurs reprises si nécessaire, des contraintes supplémentaires suivant la proposition ci-dessous. Exemple. (P) maximiser z(x1,x2) = 2x1 + xz sous les contraintes − x1 + x2 ≤ 0, 5x1 + 2x2 ≤ 18, x1,x2 ≥ 0 entiers Pour résoudre (P) procède par étapes : 1) Résolution avec le simplexe sans tenir compte de la condition "x1,x2 à valeurs entières". Solution optimale. Tableau initial c 2 1 0 0 cJ variables de base x1 x2 x3 x4 0 0 -1 1 1 0 0 18 5 2 0 1 z(x) 0 -2 -1 0 0 Tableau final c 2 1 0 0 cJ variables de base x1 x2 0 0 1 18 7 0 1 5 7 1 7 2 18 7 1 0 2 7 1 7 z(x) 54 7 0 0 1 7 3 7 Nouvelle contrainte suivant la Proposition 3.1 (choisir i = 1) : (S1) Pour exprimer (S1) en termes des variables initiales x1,x2, observer que (voir graphique plus bas) 2) Résolution de (P), (S1) inclus, par le simplexe. Exige le passage par un programme auxiliaire. Programme auxiliaire, tableau initial c 0 0 5 7 1 7 -1 0 cJ variables de base x1 x2 x3 x4 x5 y 0 18 7 0 1 5 7 1 7 0 0 0 0 xJ2 = x1 xJ3 = y 18 7 4 7 1 0 0 0 2 7 5∗ 7 1 7 1 7 0 -1 0 1 4 7 z″ 4 7 0 0 5 7 1 7 1 0 − − − − La fonction-objectif du programme auxiliaire est donné par : maximiser Programme auxiliaire, tableau final c 0 0 5 7 1 7 -1 cJ variables de base x1 x2 x3 x4 x5 0 = x2 2 0 1 0 0 1 0 = x1 14 5 1 0 0 1 5 2 5 5 7 = x3 4 5 0 0 1 1 5 7 5 4 7 z″ 0 0 0 0 0 0 − Comme max z″ = 0, le programme (P)+(S1) prossède une solution réalisable. Elle est donnée par Tableau final seconde phase c 0 0 0 2 5 1 5 cJ variables de base x1 x2 x3 x4 x5 0 = x2 2 0 1 0 0 1 0 = x1 14 5 1 0 0 1 5 2 5 0 = x3 4 5 0 0 1 1 5 7 5 38 5 z 38 5 0 0 0 2 5 1 5 La fonction-objectif, exprimée en terme des variables hors base, est donnée par La solution optimale est égale à. Nouvelle contrainte selon la Propositon 3.1 : (S2) Pour exprimer en termes de x1 et de x2 observer que Donc x1 + x2 ≤ 4 (voir graphique plus bas).
3) Résolution de (P), (S1) et (S2) inclus, exige a` nouveau le passage par un programme auxiliaire. Seconde phase du simplexe, tableau final c 0 0 0 1 3 0 1 3 cJ variables de base x1 x2 x3 x4 x5 x6 0 = x2 2 3 0 1 0 1 3 0 5 3 0 = x1 10 3 1 0 0 1 3 0 2 3 0 = x3 8 3 0 0 1 2 3 0 7 3 0 = x5 4 3 0 0 0 1 3 1 5 3 22 3 z(x) 22 3 0 0 0 1 3 0 1 3 La solution optimale est égale à . Nouvelle contrainte selon la Proposition 3.1 : (S3) x4 + x6 ≥ 1 Exprimé en termes de x1,x2 (S3) se lit : 2x1 + x2 ≤ 7 (voir graphique plus bas). 4) Résolution de (P), (S1),(S2) et (S3) inclus, exige a` nouveau le passage par un programme auxiliaire et mène, cette fois, à une solution optimale a` valeurs entières, qui est x = t(3,1,2,0,1,1,0), la solution optimale a` valeurs entières de (P) est donc donnée par x1 = 3,x2 = 1, z(x1,x2) = 7. tableau final de la seconde phase du simplexe c 0 0 0 0 0 0 1 2 cJ variables de base x1 x2 x3 x4 x5 x6 x7 0 = x2 1 0 1 0 0 0 2 1 2 0 = x1 3 1 0 0 0 0 -1 1 2 0 = x3 2 0 0 1 0 0 -3 1 0 = x5 1 0 0 0 0 1 -2 1 2 0 = x6 1 0 0 0 1 0 1 3 2 7 z(x) 7 0 0 0 0 0 0 1 2 Sur le graphique on voit que, en ajoutant successivement (S1),(S2) et (S3), on n'écarte pas de solutions à valeurs entières, et on coupe l'ensemble des solutions réalisables de fa¸con que la solution optimale a` valeurs entières appara^ıt comme point extrémal. Pr. Abdelghni LAKEHAL.