

THEOREM 2. For the system (1), the iterative process is convergent to a proper solution, for any initial approximation $M_0(x_0, y_0)$ in the interior of the F convex figure.

Proof: Let

$$(5) \quad \begin{aligned} y_i &= f(x_{i-1}) \\ x_i &= g(y_{i-1}) \end{aligned}$$

the iteration of order i . We note

$$(6) \quad \begin{aligned} a_i &= \inf_{c_{i-1} \leq y \leq d_{i-1}} g(y), & b_i &= \sup_{c_{i-1} \leq y \leq d_{i-1}} g(y) \\ c_i &= \inf_{a_{i-1} \leq x \leq b_{i-1}} f(x), & d_i &= \sup_{a_{i-1} \leq x \leq b_{i-1}} f(x) \end{aligned}$$

Evidently

$$(7) \quad \begin{aligned} a_{i-1} &\leq a_i \leq x_i \leq b_i \leq b_{i-1} \\ c_{i-1} &\leq c_i \leq y_i \leq d_i \leq d_{i-1} \end{aligned} \quad i = 1, 2, \dots$$

In the iterative process, there are two possibilities: or $\lim_{i \rightarrow \infty} (b_i - a_i) = 0$ or $\lim_{i \rightarrow \infty} (b_i - a_i) \neq 0$. In the first case, the proceeding converges. We shall analyse the second case. Let $a^* = \lim_{i \rightarrow \infty} a_i$ and $b^* = \lim_{i \rightarrow \infty} b_i$. The possibility that $\lim_{i \rightarrow \infty} (d_i - c_i) = 0$ is excluded, because in this case $g(y)$ would take an infinity of values in a point, contrary to the definition. We note that $c^* = \lim_{i \rightarrow \infty} c_i$ and $d^* = \lim_{i \rightarrow \infty} d_i$. Evidently, there are the x_2 and x_3 values in the $[a^*, b^*]$ interval, and y_2 and y_3 in the $[c^*, d^*]$ interval, so that $f(x_2) = d^*$, $f(x_3) = c^*$, $g(y_2) = b^*$ and $g(y_3) = a^*$. The preceding lemma, specifies that here $x_2 = a^*$, $x_3 = b^*$, $y_2 = c^*$, $y_3 = d^*$. According to the Theorem 1, any point of the $O_1(a^*, d^*)$ $O_2(b^*, c^*)$ segment is a proper solution of the system 1. The convergence of the iterative process (5) is also provided in this case, fact that demonstrates the theorem.

Received 19. XI. 1963

UN ALGORITHME POUR RÉSOUDRE CERTAINS PROBLÈMES DE PROGRAMMATION MATHÉMATIQUE

par
F. RADÓ

à Cluj

1. Dans le livre [5] se trouve énoncé le problème suivant de programmation linéaire à contraintes conditionnelles:

On considère l'ensemble des vecteurs $x = (x_1, x_2, \dots, x_n)$ à composantes réelles, qui vérifient les conditions

$$(1) \quad x_j \geq 0, \quad j = 1, 2, \dots, n$$

$$(2) \quad D_i(x) = \sum_{j=1}^n d_{ij} x_j - d_i \geq 0, \quad i = 1, 2, \dots, m$$

$$(3) \quad A_k(x) = \sum_{j=1}^n a_{kj} x_j - a_k < 0 \Rightarrow B_k(x) = \sum_{j=1}^n b_{kj} x_j - b_k \geq 0, \quad k = 1, \dots, p.$$

Déterminer le vecteur x , pour lequel la fonction

$$(4) \quad f(x) = \sum_{j=1}^n c_j x_j$$

prend sa valeur minimale (les nombres a_{kj} , b_{kj} , d_{ij} , a_k , b_k , d_i , c_j sont donnés).

Divers problèmes d'économie conduisent à ce modèle mathématique, qui généralise la programmation linéaire. Dans le travail [1], un problème de minimum relatif au temps de fabrication d'un produit qui passe par une série de machines, dans un ordre déterminé, a été posé sous la forme énoncée. Les problèmes de planification en temps de la production (sequencing prob-

lem, machine-job scheduling problem) peuvent être mises également sous cette forme (voir sur ce sujet [2]).

Pour résoudre ce problème, M. SIMONNARD le transforme dans un problème de programmation linéaire partiellement discret, c'est-à-dire quelques des inconnues sont soumises à la condition d'être entières. Or les algorithmes connus pour la programmation discrète sont très difficiles. Donc cette voie est peu efficace.

Le problème particulier de [1] a été résolu par des méthodes spéciales, basées sur les particularités du problème.

On peut mettre les conditions (3) sous la forme de disjonctions

$$(3') \quad A_k(x) \geq 0 \vee B_k(x) \geq 0, \quad k = 1, 2, \dots, p.$$

Les conditions (3') ont été nommées par M. SIMONNARD „contraintes mutuellement exclusives”.

Considérons les 2^p problèmes de programmation linéaire, qui sont constituées par les conditions (1), (2) et l'une des conditions $A_k(x) \geq 0$, $B_k(x) \geq 0$ pour chaque $k = 1, 2, \dots, p$ et par la fonction (4). Une méthode pour résoudre le problème proposé est manifestement la suivante : on résout tous les 2^p problèmes de programmation linéaire mentionnés et l'on choisit la solution pour laquelle f a la plus petite valeur.

Nous avons donné dans [4] la description d'une méthode plus pratique, qui est également fondée sur la résolution successive des problèmes de programmation linéaire, mais leur nombre est essentiellement moins et présente encore l'avantage que chaque problème de programmation linéaire ne diffère de l'un des précédents que par une seule condition, que l'on ajoute. En employant la méthode du simplexe duale, le tableau initial correspondant à un problème de programmation linéaire (qui est le tableau final d'un problème précédent), se transformera dans le tableau final par un nombre réduit de pas, souvent par une seule transformation.

Les problèmes de planification en temps, posés sous une forme complexe, conduisent à des conditions qui s'appellent dans [5] „ensembles mutuellement exclusifs de contraintes” ; une telle condition s'écrit

$$(5) \quad (A_1(x) \geq 0 \wedge \dots \wedge A_i(x) \geq 0) \vee (A'_1(x) \geq 0 \wedge \dots \wedge A'_i(x) \geq 0),$$

où les $A_i(x)$ et les $A'_i(x)$ sont des fonctions linéaires en $x_1, \dots, x_n$.

Nous allons développer ci-dessous la méthode qui vient d'être mentionnée sous une forme généralisée, qui s'applique aussi dans le cas quand parmi les contraintes de la problème figurent des conditions de la forme (5). De plus, ni les fonctions qui entrent dans les conditions, ni la fonction (4) ne doivent être linéaires. C'est une chose facile de généraliser la méthode pour des disjonctions à plusieurs termes, ce que nous ferons ultérieurement.

2. Les problèmes de programmation mathématique ont la forme suivante :

Soit X un ensemble quelconque, $M_1, M_2, \dots, M_m$ des sous-ensembles de X et f une fonctionnelle définie sur X à valeurs réelles ; l'on cherche $x \in X$, assujetti aux conditions

$$(6) \quad x \in M_i, \quad i = 1, 2, \dots, m.$$

de façon que

$$(7) \quad f(x) \text{ soit minimum.}$$

Nous employerons les notations suivantes :

$\Sigma = M_1 \cap M_2 \cap \dots \cap M_m$ = l'ensemble des solutions admissibles

$S = \{x | x \in \Sigma, f(x) \text{ minimum}\}$ = l'ensemble des solutions.

Le problème peut être :

1) *propre*, si $S \neq \emptyset$ (l'ensemble vide). Alors $m = \min_{x \in \Sigma} f(x)$ s'appelle le *minimum propre* du problème. Un problème propre est dit *déterminé*, lorsque S contient un seul élément ;

2) *impropre*, si $\Sigma \neq \emptyset$, $S = \emptyset$. Posons $m = \inf_{x \in \Sigma} f(x)$ et disons que m est le *minimum impropre* du problème ; une suite $x_v \in \Sigma$, jouissant de la propriété $\lim f(x_v) = m$, s'appelle une *solution impropre* ;

3) *incompatible*, si $\Sigma = \emptyset$. Alors posons par définition $m = \infty$.

Nous attachons de la sorte à chaque problème de notre type un *minimum* m . Si $m = \infty$, le problème est incompatible.

Pour les problèmes concrets de programmation mathématique l'ensemble X est, le plus souvent, un espace vectoriel V_n à n dimensions. C'est le cas de la programmation linéaire, discrète, convexe, à contraintes conditionnelles, etc. Les ensembles M_i , qui correspondent à toutes ces contraintes, sont des ensembles fermés dans V_n ; si de plus f est continue, il ne peut se présenter autre minimum impropre que $m = -\infty$.

3. Soit $\mathcal{H}$ un ensemble de parties de l'ensemble X et f une fonctionnelle réelle définie sur X de manière que l'on possède un algorithme qui résout le problème énoncé au paragraphe 2, quel que soit le système fini d'éléments $M_1, M_2, \dots, M_m$ de $\mathcal{H}$. Ce sont les *problèmes fondamentaux*.

Nous nous proposons de résoudre le suivant problème P :

On se donne les ensembles $A_1, \dots, A_p, B_1, \dots, B_p, C_1, \dots, C_q$ de $\mathcal{H}$; déterminer $x \in X$, soumis aux conditions

$$(8) \quad x \in C_j, \quad j = 1, \dots, q$$

$$(9) \quad x \in A_i \cup B_i, \quad i = 1, \dots, p.$$

de façon que

$$(10) \quad f(x) \text{ soit minimum.}$$

Le problème P est du type de celui énoncé au paragraphe 2, donc les définitions données là sont valables pour ce problème ; mais ceci n'est pas un problème fondamental, parceque les ensembles $A_i \cup B_i$ ne soit pas supposés d'être dans $\mathcal{H}$.

Nous établirons un algorithme qui indique la nature du problème P et conduit à une de ses solutions, si l n'est pas incompatible. Pour cela nous supposons qu'on possède un algorithme, applicable aux problèmes fondamentaux, jouissant de la même propriété. Nous supposons que les solutions impropres, données par cet algorithme, soient des suites d'éléments de X d'une structure particulière, c'est-à-dire qu'elles appartiennent à un certain ensemble $\mathcal{X}$ de suites dans X (si X est un espace vectoriel, on peut prendre en général pour $\mathcal{X}$ l'ensemble des suites de la forme $\{a + \lambda_n b\}$, où $a \in X$, $b \in X$ et $\{\lambda_n\}$ est une suite numérique). Nous supposons ensuite que nous ayons un algorithme par lequel l'on puisse décider si un élément $x \in X$ satisfait ou non à une condition de la forme (9) et si l existe ou non une infinité de termes d'une suite donnée de $\mathcal{X}$, qui vérifient une condition (9).

Toutes ces hypothèses sont satisfaites, si les problèmes fondamentaux sont des problèmes de programmation linéaire, discrète, concave à contraintes linéaires. Dans ces cas l'algorithme de vérification pour les suites de $\mathcal{X}$ est très simple. Si les contraintes sont dérivables, en employant la formule de Taylor, on obtient de même l'algorithme respectif de vérification.

Nous compléterons ensuite l'algorithme établi pour déterminer une solution du problème P , de sorte qu'il conduise à toutes les solutions du problème P . Pour cela il faudra supposer que l'algorithme de résolution des problèmes fondamentaux donne toutes les solutions de ces problèmes et encore que nous puissions construire l'intersection de certains sous-ensembles de X avec les ensembles de $\mathcal{H}$.

4. Nous allons décrire l'algorithme qui conduit à une solution du problème P .

Soit P_0 le problème fondamental constitué par les conditions (8), (10) et soit m_0 son minimum. Soit $k_1, k_2, \dots, k_r$ un sous-ensemble des nombres $1, 2, \dots, p$ et $\alpha_1, \alpha_2, \dots, \alpha_r$ des nombres égaux à 0 ou 1 ; notons par

$$(11) \quad P_{k_1, k_2, \dots, k_r}^{\alpha_1, \alpha_2, \dots, \alpha_r}$$

le problème fondamental comprenant les conditions (8), (10) et les r conditions suivantes :

$$x \in A_{k_v}, \text{ lorsque } \alpha_v = 0$$

$$x \in B_{k_v}, \text{ lorsque } \alpha_v = 1, v = 1, 2, \dots, r.$$

On note le minimum du problème (11) par

$$(12) \quad m_{k_1, k_2, \dots, k_r}^{\alpha_1, \alpha_2, \dots, \alpha_r}$$

Nous construirons par un procédé itératif une suite (*), constituée par des problèmes fondamentaux du type (11), et en même temps avec la formation de chaque problème de la suite (*) nous inscrirons dans une deuxième suite (**) le minimum du problème respectif et dans une troisième suite (***) l'une quelconque de ses solutions, pourvu qu'il existent telles solutions ; sinon, on posera ∞ aux positions respectives. Nous marquerons certains termes, en soulignant les termes correspondantes dans les trois suites, lorsque la solution qui se trouve inscrite dans (***) satisfait à toutes les conditions (9). On doit préciser : une solution impropre étant une suite $\{x_v\}$, nous disons que cette suite satisfait à une condition de la forme (9), lorsqu'une infinité des termes x_v satisfont à cette condition.

Posons dans la première place de la suite (*) le problème P_0 et complétons les premières places des autres suites comme il a été dit ; voyons s'il est nécessaire de souligner ces termes.

Imaginons maintenant que l'on ait déjà construit les trois suites jusqu'à un certain rang et l'on ait effectué les soulignations nécessaires. Considérons les termes de la suite (**), qui ont la plus petite valeur dans (**), et distinguons les trois cas suivants :

a) Aucun de ces minima n'est souligné et au moins l'un est différent de ∞ . Choisissons un tel terme, soit (12) ; le terme correspondant dans (*) et alors (11), et celui dans (***) soit noté par x' . En tenant compte que x' n'est pas souligné, on peut choisir un indice k_{r+1} de la sorte que

$$x' \notin A_{k_{r+1}} \cup B_{k_{r+1}}.$$

Supprimons le terme (11) et ajoutons aux termes de la suite (*) les deux suivants :

$$(13) \quad P_{k_1, \dots, k_r, k_{r+1}}^{\alpha_1, \dots, \alpha_r, 0} \text{ et } P_{k_1, \dots, k_r, k_{r+1}}^{\alpha_1, \dots, \alpha_r, 1}$$

Supprimons les termes qui correspondent à (11) dans les suites (**) et (***) et ajoutons là ceux qui correspondent aux termes (13). Vérifions si les termes nouveaux dans (***) satisfont à toutes les conditions (9), et soulignons dans le cas affirmatif les termes respectifs.

b) Si au moins l'un des plus petits termes de la suite (**) est souligné, la construction des suites est terminée. Lorsque parmi les problèmes (*) qui correspondent à ces termes soulignés se trouvent des problèmes fondamentaux propres, alors on choisit l'un d'eux, soit $\tilde{P}$; si ces-ci étaient tous impropres, on noterait par $\tilde{P}$ l'un quelconque de ces problèmes. Soit $\tilde{m}$ et $\tilde{x}$ les termes des suites (**) et (***) qui correspondent à $\tilde{P}$.

Le problème P est propre ou impropre selon qu'il est le problème fondamental $\bar{P}$; $\tilde{m}$ est le minimum du problème P et $\tilde{x}$ l'une de ses solutions.

c) Si tous les termes de la suite (**) sont ∞ , la construction est aussi terminée. Dans ce cas le problème P est incompatible.

Le procédé itératif s'achève en un nombre fini de pas.

Démonstration. Si $r = p$, le terme (12) est manifestement ou bien souligné, ou bien ∞ ; donc si l'on avait pour chaque terme $r = p$, nous serions dans les cas b) ou c) et le procédé serait terminé. Mais, si le procédé ne s'achève pas plus vite, on aboutit, en un nombre fini de pas, au cas $r = p$ pour tous les termes. Donc le procédé s'achève.

Toute solution admissible x^* du problème P est une solution admissible pour l'un au moins des problèmes de la suite (*). En effet x^* satisfait à la condition (8), donc il est une solution admissible du problème P_0 . Considérons la suite (*) à une étape de sa construction, soit à l'itération d'ordre s ; supposons que x^* soit une solution admissible d'un certain problème (disons P') de cette suite et démontrons que cette propriété reste valable à l'itération suivante. En passant à l'itération $s+1$, un problème (*) est remplacé par deux autres; lorsque ce problème n'est pas P' , la propriété rest manifestement valable; si ceci est justement P' , on observe que x^* sera admissible pour l'un au moins des deux problèmes, qui viennent d'occuper la place de P' .

On a pour le minimum m du problème P

$$(14) \quad m \geq \min (**).$$

En effet, si P est un problème propre et $\tilde{x}$ l'une de ses solutions, $m = f(\tilde{x})$. Nous venons de voir que $\tilde{x}$ est une solution admissible d'un certain problème de la suite (*), donc $f(\tilde{x}) \geq$ que le minimum de ce problème, et a fortiori $f(\tilde{x}) \geq \min (**)$. Si P est un problème impropre et $\tilde{x} = \{\tilde{x}_v\}$, $v = 1, 2, \dots$ est l'une de ses solutions impropres, on a $m + \varepsilon > f(\tilde{x}_v)$, lorsque $v > N(\varepsilon)$, d'où la relation (14) résulte. Si le problème P est incompatible, la relation (14) est évidente.

Supposons que nous nous trouvions dans le cas b); $\min (**) = \tilde{m}$. Si $\bar{P}$ est un problème fondamental propre, on a $\tilde{m} = f(\tilde{x})$. $\tilde{x}$ étant souligné, il est une solution admissible du problème P , donc $m \leq f(\tilde{x})$; compte tenu de (14), il vient

$$(15) \quad m = \min (**) = \tilde{m} = f(\tilde{x});$$

ainsi le problème P est propre, dont $\tilde{x}$ est une solution. Soit maintenant $\bar{P}$ un problème impropre, $\tilde{x} = \{\tilde{x}_v\}$, $v = 1, 2, \dots$; $\tilde{m} = \lim f(\tilde{x}_v)$. Pour une infinité d'indice v , soit v_μ , $\tilde{x}_{v_\mu}$ est une solution admissible du problème $\bar{P}$, donc

$$m \leq f(\tilde{x}_{v_\mu}).$$

On en déduit $m \leq \tilde{m} = \min (**)$; donc $\tilde{m} = \min (**) = m$. Le problème P doit être impropre; car si l'existait une solution admissible $\bar{x}$ du problème P , jouissant de la propriété $f(\bar{x}) = m$, $\bar{x}$ serait une solution admissible d'un certain problème $\bar{P}$ de la suite (*) et $\bar{x}$ serait une solution propre de $\bar{P}$; ce qui contredit l'hypothèse que tous les problèmes de la suite (*) à minimum égal à m , sont impropres. $\tilde{x}_{v_\mu}$, $\mu = 1, 2, \dots$, étant des solutions admissibles du problème P et $\lim f(\tilde{x}_{v_\mu}) = m$, la suite $\{\tilde{x}_{v_\mu}\}$ représente bien une solution (impropre) du problème P . Et les affirmations du cas b) se trouvent ainsi établies.

On doit encore établir, que le problème P est incompatible dans le cas c). En effet, s'il avait une solution admissible, celle-ci devrait être également une solution admissible d'un certain problème (*); ce que est absurde.

5. Nous désirons de décider dans le cas du problème propre, si la solution est unique ou non. Notre algorithme conduit au sous-cas b) quand P est propre. Considérons l'ensemble $\mathcal{M}$ des problèmes propres de la suite (*) qui ont leur minimum égal à $\tilde{m}$, et le sous-ensemble $\mathcal{M}'$ des problèmes soulignés de $\mathcal{M}$.

1) Si $\mathcal{M}' = \mathcal{M}$, $\mathcal{M}'$ est réduit à un seul problème fondamental et ceci a une solution unique, alors le problème P admet aussi une seule solution.

2) Si ou bien $\mathcal{M}'$ contient au moins deux problèmes ou bien l'unique problème de $\mathcal{M}'$ a plusieurs solutions, alors le problème P admet plusieurs solutions.

3) Si $\mathcal{M}'$ se réduit à un seul problème à solution unique, mais $\mathcal{M}' \neq \mathcal{M}$, alors on doit appliquer de nouveau un pas a) du notre algorithme, en prenant pour le problème (11) qui figure là l'un des problèmes appartenant à $\mathcal{M} \setminus \mathcal{M}'$ et continuer de la sorte jusqu'on arrive au cas 1) ou 2).

Lorsqu'on désire de connaître toutes les solutions du problème P , on continue les itérations de 3) jusqu'on a $\mathcal{M}' = \mathcal{M}$. Puis on établit pour chaque problème fondamental de $\mathcal{M}'$ l'ensemble des solutions et le sous-ensemble qui satisfait aux conditions (9). Ces sous-ensembles réunis donnent l'ensemble S des solutions du problème P .

6. Généralisons le problème P , en changeant les conditions (9) par les suivantes :

$$9') \quad x \in A_i^{(1)} \cup A_i^{(2)} \cup \dots \cup A_i^{(s_i)}, \quad i = 1, \dots, p,$$

où les ensembles $A_i^{(s)}$ appartiennent à $\mathcal{H}$.

On peut procéder de proche en proche, en ajoutant premier fois, à $\mathcal{H}$ les ensembles $A_i^{(j)} \cup A_i^{(l)}$, puis $A_i^{(j)} \cup A_i^{(l)} \cup A_i^{(m)}$ et ainsi de suite.

Mais il est beaucoup plus avantageux de modifier l'algorithme décrit au paragraphe 4: Les nombres α_v prennent maintenant plusieurs valeurs: $0, 1, \dots, s_{k_v} - 1$, et le même est le nombre des problèmes qui remplacent, pendant le pas a), le problème supprimé de la suite (*); nous aurons dans (13) s_{k_v} problèmes à la place des deux. Ce sont les seules modifications nécessaires.

7. Nous revenons maintenant au problème linéaire formulé au paragraphe 1 par les contraintes (1), (2), (3') et la fonction (4). Nous allons résoudre numériquement ce problème en employant une technique de calcul basée sur notre algorithme et sur la méthode du simplexe dual (Voir [5]; une justification directe de la méthode du simplexe dual se trouve dans [3]).

Nous supposons que

$$(16) \quad c_j \geq 0, j = 1, 2, \dots, n.$$

Si le problème est propre, on pourra toujours trouver une transformation linéaire de variables de manière que les conditions (16) soient remplies.

On résout chaque problème de programmation linéaire de la suite (*) par une succession de tableaux. On écrit les coefficients et le terme libre d'une condition (1) ou (2) dans une ligne du tableau initial et ceux d'une condition (3') dans deux lignes. Dans un problème (*) il n'interviennent qu'une partie des contraintes correspondant à ce lignes, qui seront marquées par le signe x . Voici le tableau initial :

		x_1 x_2 $\dots$ x_n	Terme libre
x		1	0
x			0
$\vdots$			$\vdots$
x			0
x			
$\vdots$			
x			
		d_{11} d_{12} $\dots$ d_{1n}	d_1
		$\vdots$	$\vdots$
		d_{m1} d_{m2} $\dots$ d_{mn}	d_m
	1)	a_{11} a_{12} $\dots$ a_{1n}	a_1
		b_{11} b_{12} $\dots$ b_{1n}	b_1
	$\vdots$		
	p)	a_{p1} a_{p2} $\dots$ a_{pn}	a_p
		b_{p1} b_{p2} $\dots$ b_{pn}	b_p
	f	c_1 c_2 $\dots$ c_n	0

Notons une ligne quelconque de ce tableau par

$$l_{v1}^0, l_{v2}^0, \dots, l_{vn}^0, l_v^0$$

et soit $L_v^0 = (l_{v1}^0, l_{v2}^0, \dots, l_{vn}^0)$. Soit un tableau quelconque (qui se rattache éventuellement à un autre problème de la suite (*)) :

	x_1 x_2 $\dots$ x_n	T.l.
n lignes {	M	X
$\vdots$		
	L_v	t_v
$\vdots$		

Nous calculerons ce tableau par le procédé de la méthode du simplexe révisée : les premières n lignes sont calculées du tableau antérieur par l'élimination usuelle, tandis que les autres lignes selon les formules

$$(17) \quad L_v = L_v^0 \times M$$

$$(18) \quad t_v = L_v^0 \times X + t_v^0.$$

De ces éléments on calcule seulement ceux de la colonne des termes libre et d'une certaine ligne.

Un problème $P_{k_1, k_2, \dots, k_r}^{\alpha_1, \alpha_2, \dots, \alpha_r}$ se trouve résolu, lorsqu'on a des nombres non-négatifs à l'intersection de la colonne T.l. avec les lignes marquées. Sa solution est le vecteur dans la position de X et son minimum se trouve dans la ligne f et la colonne T.l.

Nous vérifions les conditions (3') restantes de même par la formule (18). Si ces conditions ne sont pas toutes vérifiées, nous choisissons l'indice k_{r+1} et complétons les deux lignes qui correspondent à cet indice. En considérant seulement la première, nous avons un tableau simplexe dual pour le problème $P_{k_1, \dots, k_r, k_{r+1}}^{\alpha_1, \dots, \alpha_r, 0}$ et en considérant le deuxième, un tableau pour $P_{k_1, \dots, k_r, k_{r+1}}^{\alpha_1, \dots, \alpha_r, 1}$.

En même temps avec les transformations des tableaux nous complétons les suites (*) et (**). On ne doit écrire la suite (***), car celle-ci est incluse dans les tableaux succésifs.

La solution est $x_1 = \frac{17}{3}$, $x_2 = \frac{2}{3}$, $x_3 = 0$; $f_{\min} = \frac{19}{3}$. C'est la seule solution, puisque le nombre $\frac{19}{3}$, qui représente le minimum de la suite (**), n'y figure qu'une seule fois.

BIBLIOGRAPHIE

- [1] N é m e t i L., R a d ó F., *Ein Wartezeitproblem in der Programmierung der Produktion*. Mathematica, 5 (28), 65–95 (1963).
- [2] N é m e t i L., *Das Reihenfolgeproblem in der Fertigungsprogrammierung und Linearplanung mit logischen Bedingungen*. Mathematica, 6 (29), 87–99 (1964).
- [3] R a d ó F., *Despre problema programării liniare*. Studii și cercet. de mat. Cluj, 11, vol. annexe: Probl. teor. și pract. calc., 167–177 (1960).
- [4] R a d ó F., *Programarea liniară cu condiții logice*. Comunic. Acad. R.P.R., 13, 12, 1039–1042 (1963).
- [5] S i m o n n a r d M., *Programmation linéaire*. Dunod, Paris, 1962.

Reçu le 19. XI. 1963

ERROR ESTIMATION BY MEANS OF DIFFERENTIAL INEQUALITIES.

by

E. SCHECHTER

Cluj

1. Many of the estimations of errors in the numerical solution of differential equations, known as yet, are in most cases not suited to actual numerical computations. The reason is, that they are either too complicated, or they require too much information about the equation involved. Therefore in practice one often resorts to less rigorous procedures.

The purpose of the paper is to achieve some progress in this direction. The method used is that of differential inequalities; especially Szarki's results [6] are fundamental to us.

To apply them we first need to continue adequately the discrete function, obtained by numerical integration, on the whole interval. Such a continuation is discussed in the second paragraph for ordinary differential equations (another possibility may be found in [7]). Certain special cases were studied for several variables too [3]. Next paragraph is devoted to ordinary differential equations. Some differential inequalities are presented here with an eye on the numerical purposes. Comparison is made between (3.10) and an estimation of Bieberbach. It follows that though the information required is less, (the Lipschitz constant only) the estimation by (3.10) is better in some cases.

Last paragraph deals with parabolic equations. The results do not essentially differ from those in the previous paragraph so that we haven't discussed them in detail.

2. Let

$$(2.1) \quad y' = f(t, y)$$

be a system of differential equations with the initial conditions

$$(2.2) \quad y(t_0) = y_0$$