PROGRAMAREA LA MAȘINILE ELECTRONICE A UNEI PROBLEME DE OPTIMIZARE

L. NEGRESCU, I. PĂVĂLOIU și T. RUS

Problema alegerii optime a planurilor economice este, în general, o problemă matematică, care se poate include în schema generală a căutării optimului unei funcții cu legături date. Spre deosebire însă de modelul matematic citat mai sus, modelul real al unei probleme de optimizare în problemele economice nu privește un fenomen continuu, ci are un caracter discret. Acest caracter face ca metodele clasice ale matematicii, de găsire a procesului optim, să nu fie aplicabile, deoarece aceste metode presupun în afară de continuitate și derivabilitatea funcțiilor cu care se operează.

Din această cauză și datorită importanței planificării optime a proceselor economice, se caută metode matematice noi, care să permită rezolvarea problemei de mai sus, fără a face apel la proprietăți de continuitate și derivabilitate.

Vom examina așa-numita problemă de programare liniară precum și problema de găsire a extremelor unei funcții de mai multe variabile supuse la anumite legături, unde funcțiile și variabilele iau valori discrete. De fapt modelul programării liniare se include în cel de-al doilea model, avînd particularitatea că atît funcția de optimizat cît și legăturile la care sînt supuse variabilele sînt liniare, ceea ce și explică puternica dezvoltare a metodelor de rezolvare a problemelor de programare liniară.

Indiferent însă de modelul matematic ales pentru rezolvarea problemelor economice de această natură, găsirea soluției efective a acestei probleme este îngreunată de faptul că calculele necesare sînt uneori atît de laborioase, încît cu metode obișnuite nu se pot efectua în timp util.

Apariția mașinilor electronice de calcul a însemnat, din acest punct de vedere, o adevărată revoluție în tehnica calculului. Acest fenomen a fost determinat de necesitățile imperioase ale vieții, de sarcinile avîntului economiei planificate.

În prezent, Institutul de calcul al Filialei Cluj a Academiei R.P.R. efectuează lucrări interesante pe baza cărora s-au obținut rezultate importante și

s-a acumulat o experienţă pozitivă în domeniul folosirii metodelor matematice şi a maşinilor electronice de calcul în diferite probleme economice şi de planificare, probleme cerute de diferite întreprinderi şi instituţii din ţară.

În cele ce urmează vom expune rezolvarea unei asemenea probleme la o maşină electronică de calcul, şi anume problema determinării duratei optime a campaniei de prelucrare a zahărului în condiţii specifice ţării noastre, problemă al cărei model matematic corespunde unei probleme de programare dinamică de tipul problemelor de teoria stocurilor.

Acest model permite determinarea influenţei pe care o exercită asupra rezultatelor economice factorii agronomici, tehnologici şi financiari. Algoritmul de rezolvare a fost adaptat la maşina de calcul MECIPT-1, el încadrîndu-se într-o schemă mult mai generală, pe care o prezentăm mai jos.

Se dau variabilele x, y, u, v care iau valori discrete în intervalele

$$X_0(u, v) \leqq x \leqq X_1(u, v),\quad Y_0(x, u, v) \leqq y \leqq Y_1(x, u, v),\quad U_0 \leqq u \leqq U_1,$$
$$V_0 \leqq v \leqq V_1,$$

unde $X_i(u, v)$, $Y_i(x, u, v)$, $i = 0, 1$, sînt funcţii date definite pentru u, v şi x din intervalele respective, satisfăcînd la condiţiile $X_0(u, v) \leqq X_1(u, v)$ şi $Y_0(x, u, v) \leqq Y_1(x, u, v)$, iar U_0, U_1, V_0, V_1 sînt constante.

Se cere să se minimizeze (maximizeze) funcţia

$$c = \Phi(x, y, u, v, p, q, r, s),$$

ştiind că x, y, u, v, q, p, r, s satisfac relaţiile

$$\begin{aligned}
&\varphi_1(x, y, u, v, p) = 0, && a_0 \leqq p \leqq a_1,\\
&\varphi_2(x, y, u, v, p, q) = 0, && b_0 \leqq q \leqq b_1,\\
&\varphi_3(x, y, u, v, p, q, r) = 0, && c_0 \leqq r, \leqq c_1,\\
&\varphi_4(x, y, u, v, p, q, r, s) = 0, && d_0 \leqq s \leqq d_1.
\end{aligned} \tag{1}$$

Funcţiile φ_i, $i = 1, 2, 3, 4$, sînt astfel încît determină, pentru fiecare sistem de valori date variabilelor x, y, u, v, în mod unic necunoscutele p, q, r, s.

Fie

$$\begin{aligned}
&x_1 < x_2 < x_3 < \ldots < x_{n-1} < x_n,\quad y_1 < y_2 < y_3 < \ldots < y_{m-1} < y_m,\\
&U_0 \leqq u_1 < u_2 < u_3 < \ldots < u_{T-1} < u_T \leqq U_1,\\
&V_0 \leqq v_1 < v_2 < v_3 < \ldots < v_{k-1} < v_k \leqq V_1
\end{aligned} \tag{2}$$

mulţimea valorilor discrete pe care le iau respectiv variabilele x, y, u, v.

Se observă că problema propusă are caracterul unei probleme de extremum legat, care însă nu se poate rezolva aplicînd metode clasice, dat fiind faptul că varabilele x, y, u, v iau valori discrete, iar asupra funcţiilor $X_i(u, v)$, $Y_i(x, u, v)$, $i = 0, 1$, şi φ_i, $i = 1, 2, 3, 4$, nu au fost impuse ipoteze de derivabilitate şi continuitate.

Pentru a rezolva problema procedăm astfel:

Pentru fiecare pereche (u_i, v_j) din mulţimea (2) se calculează valorile funcţiilor X_0 şi X_1, determinîndu-se astfel mulţimea valorilor lui x corespunzătoare perechii (u_i, v_j). După aceea se calculează valorile funcţiilor Y_0 şi Y_1, determinîndu-se astfel mulţimea valorilor lui y corespunzătoare tripletului (x_k, u_i, v_j).

Un sistem (x, y, u, v) astfel determinat îl vom numi sistem admisibil.

Fiecărui sistem admisibil (x, y, u, v) îi corespunde, conform relaţiilor (1), cel mult un sistem de valori (p, q, r, s).

Cu ajutorul relaţiilor (1) determinăm toate aceste valori corespunzătoare tuturor sistemelor admisibile de forma (x, y, u, v). Obţinem astfel o mulţime finită de sisteme

$$M\{x, y, u, v, p, q, r, s\}.$$

Pentru fiecare element al mulţimii M se calculează valoarea funcţiei $\Phi(x, y, u, v, p, q, r, s)$ şi notăm cu N mulţimea valorilor obţinute.

Soluţia problemei este dată de $l = \min N$ ($L = \max N$).

Dăm acum schema logică generală a programului de rezolvare a problemei după metoda descrisă mai sus.

Introducem următoarele notaţii:

$\Pi_0^{(i,j)}$ – programul de calcul al funcţiei $X_0(u, v)$ pentru $u = u_i$ şi $v = v_j$;
$\Pi_1^{(i,j)}$ – programul de calcul al funcţiei $X_1(u, v)$ pentru $u = u_i$ şi $v = v_j$;
P_1 – operatorul logic care verifică inegalitatea $X_0(u_i, v_j) \leqq X \leqq X_1(u_i, v_j)$;
$\Pi_2^{(i,j,k)}$ – programul de calcul al funcţiei $Y_0(x, u, v)$ pentru $x = x_k$, $u = u_i$ şi $v = v_j$;
$\Pi_3^{(i,j,k)}$ – programul de calcul al funcţiei $Y_1(x, u, v)$ pentru $x = x_k$, $u = u_i$ şi $v = v_j$;
P_2 – operatorul logic care verifică inegalitatea

$$Y_0(x_k, u_i, v_j) \leqq y \leqq Y_1(x_k, u_i, v_j);$$

Π_4 – programul care rezolvă ecuaţia $\varphi_1(x_k, y_s, u_i, v_j, p) = 0$;
P_7 – operatorul logic care verifică condiţia $i > K$;
P_8 – operatorul logic care verifică condiţia $j > T$;
P_3 – operatorul logic care verifică inegalitatea $a_0 \leqq p \leqq a_1$;
Π_5 – programul care rezolvă ecuaţia $\varphi_2(x_k, y_s, u_i, v_j, p_\lambda, q) = 0$, unde p_λ este soluţia ecuaţiei $\varphi_1(x_k, y_s, u_i, v_j, p) = 0$ şi $a_1 \leqq p_\lambda \leqq a_1$;
P_4 – operatorul logic care verifică inegalitatea $b_0 \leqq q \leqq b_1$;
Π_6 – programul care rezolvă ecuaţia $\varphi_3(x_k, y_s, u_i, v_j, p_\lambda, q_\lambda, r) = 0$, unde q_λ este soluţia ecuaţiei $\varphi_2(x_k, y_s, u_i, v_j, p_\lambda, q) = 0$ şi $b_0 \leqq q_\lambda \leqq b_1$;
P_5 – operatorul logic care verifică inegalitatea $c_0 \leqq r \leqq c_1$;
Π_7 – programul care rezolvă ecuaţia $\varphi_4(x_k, y_s, u_i, v_j, p_\lambda q_\lambda, r_\lambda, s) = 0$, unde r_λ este soluţia ecuaţiei $\varphi_3(x_k, y_s, u_i, v_j\ v_\lambda, q_\lambda, r) = 0$ şi $c_0 \leqq r \leqq c_1$;

P_6 – operatorul logic care verifică inegalitatea $d_0 \leqq s \leqq d_1$;
Π_8 – programul care calculează valoarea funcției $\Phi(x_k, y_s, u_i, v_j, p_\lambda, q_\lambda, r_\lambda, s_\lambda)$ pentru fiecare element al mulțimii M;
$F_1(s)$ – operatorul de readresare al indicelui s al variabilei y;
$F_2(k)$ – „ „ „ „ „ „ k „ „ x;
$F_3(j)$ – „ „ „ „ „ „ j „ „ v;
$F_4(i)$ – „ „ „ „ „ „ i „ „ u;
$O_1(s)$ – operatorul de restabilire al indicelui s al variabilei y;
$O_2(k)$ – „ „ „ „ „ „ k „ „ x;
$O_3(j)$ – „ „ „ „ „ „ j „ „ v;
$O_4(i)$ – „ „ „ „ „ „ i „ „ u;
Π_9 – programul de alegere al soluției optime și tipărire.

Cu aceste notații schema logică generală a programului are următoarea formă:

$$\overset{5}{\downarrow}\Pi_0^{(i,j)}\,\Pi_1^{(i,j)}\,\overset{4}{\downarrow}P_1(X_0 \leqq x_k \leqq X_1)\overset{1}{\uparrow}\Pi_2^{(i,j,k)}\,\Pi_3^{(i,j,k)}\overset{3}{\downarrow}P_2(Y_0 \leqq y_s \leqq Y_1) \times$$

$$\times\overset{2}{\uparrow}\Pi_4 P_3(a_0 \leqq p_\lambda \leqq a_1)\overset{2}{\uparrow}\Pi_5 P_4(b_0 \leqq q_\lambda \leqq b_1)\overset{2}{\uparrow}\Pi_6 P_5(c_0 \leqq r_\lambda \leqq c_1) \times \quad (3)$$

$$\times\overset{2}{\uparrow}\Pi_7 P_6(d_0 \leqq s_\lambda \leqq d_1)\overset{2}{\uparrow}\Pi_8\Pi_9\overset{2}{\downarrow}F_1(s)\,P_9(s > m)\overset{3}{\uparrow}\overset{1}{\downarrow}F_2(k)\,O_1(s) \times$$

$$\times P_{10}(k > n)\overset{4}{\uparrow}F_3(j)\,O_1(s)\,O_2(k)\,P_7(j > T)\overset{5}{\uparrow}F_4(i)\,O_1(s)\,O_2(k)\,O_3(j) \times P_8(i > k)\overset{5}{\uparrow}O_p,$$

unde P_9 și P_{10} sînt operatori logici care verifică condiția $s > n$, respectiv $k > n$. Parametrii i, j, k, s inițial iau valoarea 1.

Programele Π_i, $i = 0, 1, \ldots, 8$, după cum s-a specificat mai sus, sînt programe de calcul al valorilor unor funcții și, prin urmare, forma și dimensiunile lor depind de forma funcțiilor ale căror valori le calculează.

Parametrii i, j, k, s sînt localizații în programele Π_i în sensul că valorile pe care le iau sînt specifice fiecărui program în parte.

Asupra schemei (3) se pot pune diferite probleme fie legate de ordinea în care trebuie luate programele Π_i în schemă, în așa fel ca să obținem economie maximă de timp și de celule ocupate în memorie, fie de reducerea numărului de programe Π_i la un număr minim, în așa fel ca să putem rezolva problema într-un caz destul de general.

Prima problemă nu o vom aborda în această lucrare, ea fiind obiectul unei comunicări ulterioare. Problemei reducerii numărului programelor Π_i îi corespunde, în formularea matematică a problemei în general, particularizarea funcțiilor φ_i, $i = 1, 2, 3, 4$, în așa fel încît cu un număr minim de programe Π_i să se poată rezolva un număr maxim de ecuații $\varphi_i = 0$.

O problemă dictată de practică și anume problema determinării perioadei optime de recoltare și fabricare a zahărului din sfecla de zahăr, care se încadrează în schema generală propusă de noi mai sus, a condus la următorul model particular al problemei date:

Să se minimizeze

$$C = \frac{C_1(x, y, u, v, p, q, r, s)}{C_2(x, y, u, v, p, q, r, s)}, \quad (4)$$

unde $C_i = \int_{p_0^{(1)}}^{p} \Phi_1^{(i)}(t, x, y, u, v, p, q, r, s)\,dt + \int_{p_0^{(2)}}^{q} \Phi_2^{(i)}(t, x, y, u, v, p, q, r, s)\,dt +$

$$+ \int_{p_0^{(3)}}^{r} \Phi_3^{(i)}(t, x, y, u, v, p, q, r, s)\,dt + \int_{p_0^{(4)}}^{s} \Phi_4^{(i)}(t, x, y, u, v, p, q, r, s)\,dt +$$

$$+ \Phi_5^{(i)}(x, y, u, v, p, q, r, s), \quad (i = 1, 2), \quad (5)$$

iar x, y, u, v, p, q, r, s satisfac următoarele condiții:

$$X_0(u, v) \leqq x \leqq X_1(u, v),\ Y_0(x, u, v) \leqq y \leqq Y_1(x, u, v),\ U_0 \leqq u \leqq U_1,$$
$$V_0 \leqq v \leqq V_1,\ a_0 \leqq p \leqq a_1,\ b_0 \leqq q \leqq b_1,\ c_0 \leqq r \leqq c_1,\ d_0 \leqq s \leqq d_1 \quad (6)$$

și ecuațiile (1) sînt de forma

$$\int_{p_0}^{p} \psi_0(t, x, y, u, v)\,dt = \omega_0(x, y, u, v),$$

$$\int_{q_0}^{q} \psi_1(t, x, y, u, v, p)\,dt = \int_{p_0}^{p} \omega_1(t, x, y, u, v, p)\,dt + \lambda_0(x, y, u, v, p), \quad (7)$$

$$\int_{r_0}^{r} \psi_2(t, x, y, u, v, p, q)\,dt = \int_{q_0}^{q} \omega_2(t, x, y, u, v, p, q)\,dt +$$

$$+ \int_{p_0}^{p} \lambda_1(t, x, y, u, v, p, q)\,dt + \mu_0(x, y, u, v, p, q),$$

$$\int_{s_0}^{s} \psi_3(t, x, y, u, v, p, q, r)\,dt = \int_{r_0}^{r} \omega_3(t, x, y, u, v, p, q, r)\,dt +$$

$$+ \int_{q_0}^{q} \lambda_2(t, x, y, u, v, p, q, r)\,dt + \int_{p^0}^{p} u_1(t, x, y, v, p, q, r)\,dt + \nu_0(x, y, u, v, p, q, r),$$

unde toate funcțiile ce intervin sînt pozitive. Pentru orice sistem admisibil (x_k, y_s, u_i, v_j) presupunem că funcțiile $\psi_0, \omega_1, \psi_1, \lambda_1, \omega_2, \psi_2, \mu_1, \lambda_2, \omega_3, \psi_3$ sînt definite pe noduri echidistante în raport cu variabila t și de pașii constanți h_i, $i = 1, 2, \ldots, 10$, începînd cu nodurile inițiale date p_0, q_0, r_0, s_0, care corespund limitei inferioare a integralelor respective. De asemenea, pentru orice element al mulțimii M presupunem că funcțiile $\Phi_j^{(i)}$, $(j = 1, 2, 3, 4)$, $i = 1, 2)$, sînt definite pe nodurile echidistante în raport cu t începînd cu nodurile inițiale p_0^i și de pașii constanți $h_{10+j} (j = 1, 2, 3, 4)$ pentru $i = 1$ și $h_{14} + j$ pentru $i = 2$.

Pentru calculul integralelor din formulele (5) și (7) utilizăm formula de cvadratură a trapezului. Programele $\Pi_0^{(i,j)}$, $\Pi_1^{(i,j)}$, $\Pi_2^{(i,j,k)}$, $\Pi_3^{(i,j,k)}$ din schema (3) depind de forma funcțiilor X_0, X_1, Y_0 și Y_1, ele rămînînd aceleași și în schema de rezolvare a modelului particular.

Programul Π_4 realizează următorul algoritm. Se calculează valoarea funcției ω_0 pentru un sistem admisibil (x, y, u, v).

Se face $p = p_0 + ih_1$, unde $i = 1$; la început se calculează valoarea integralei $\int_{p_0+(i-1)h}^{p_0+ih} \psi_0(t, x, y, u, v)\,dt$, pentru același sistem admisibil (x, y, u, v).

După aceasta, dacă notăm cu γ valoarea funcției ω_0 și cu δ valoarea integralei, se verifică condiția logică $\delta > \gamma$. Dacă această condiție nu este înde-

plinită, atunci se mărește i cu o unitate și se trece la operatorul de calcul al integralei după care rezultatul se adună cu cel precedent. Acest ciclu se repetă pînă cînd condiția logică $\delta > \gamma$ este adevărată. Pentru a obține soluția cu precizia dată (făcînd abstracție de eroarea datorită formulei de cvadratură întrebuințată), procedăm prin înjumătățirea ultimului interval de integrare, astfel că dacă ε este precizia cerută, atunci numărul pașilor de înjumătățire este dat de formula $N = \left\lfloor \log_2 \frac{h}{\varepsilon} \right\rfloor + 1$. Dacă memoria mașinii permite păstrarea tuturor soluțiilor primei ecuații, atunci, considerînd șirul tuturor sistemelor admisibile (x, y, v, u), vom obține un șir corespunzător de soluții p.

Dacă notăm cu $A_1^{(x,y,u,v)}$, operatorul care calculează valoarea funcției ω_0, cu A_2^i operatorul care calculează valoarea integralei, cu A_3 operatorul care execută interpolarea prin înjumătățire, schema logică a programului are următoarea formă:

$$\left[\begin{matrix} 1 \Rightarrow i,\ j,\ k,\ r \\ (x_i,\ y_j,\ u_k,\ v_k) \Rightarrow (x,\ y,\ u,\ v) \end{matrix}\right] A_1^{(x,y,u,v)} \overset{s_1}{\downarrow} \overset{1}{\downarrow} A_2^i\ P\ (\gamma > \delta) \overset{2}{\uparrow} F\ (i)\ O \overset{1}{\uparrow} \times$$

$$\times \overset{3}{\downarrow} \overset{2}{\downarrow} [k + 1 \Rightarrow k]\ P\ (N > k) \overset{4}{\uparrow} A_3 O \overset{3}{\uparrow} \overset{4}{\downarrow} O_p, \qquad (S_1)$$

unde O_p este sau operatorul de oprire, dacă se calculează o singură soluție p, sau un grup de operatori care realizează redresarea și restabilirea tuturor parametrilor necesari pentru a se putea trece la calculul unei alte soluții p.

Să observăm acum că relația a doua din (7) diferă de prima relație numai prin faptul că apare în membrul al doilea o integrală a cărei limită superioară este soluția ecuației rezolvată de programul schemei (S_1) (bineînțeles abstracție făcînd de forma funcțiilor care apar).

Prin urmare, se vede clar că în acest caz vom avea de a face cu doi operatori în plus. Primul va fi operatorul care calculează valoarea integralei din membrul al doilea (a cărui structură este analogă cu a operatorului A_2^i din schema (S_1)), iar al doilea operator este operatorul de interpolare pentru care această integrală a cărui conținut îl precizăm mai jos. Deoarece noi nu cunoaștem valoarea funcției de sub semnul integralei în punctul p, operatorul de calcul al integralei calculează expresia

$$\sum_{i=1}^{m} \int_{p_0+(i-1)h_2}^{p_0+ih_2} \omega_1\ (t,\ x,\ y,\ u,\ v,\ p)\ dt, \qquad (8)$$

unde m este astfel ales ca $p_0 + (m - 1)\ h_2 < p < p_0 + mh_2$.

Pentru a calcula valoarea integralei $\int_{p_0}^{p} \omega_1\ (t,\ x,\ y,\ u,\ v,\ p)\ dt$ se observă că valoarea funcției $\omega_1\ (t,\ x,\ y,\ u,\ v,\ p)$ pentru $t = p$ poate fi calculată după următoarea formulă:

$$\omega_1(p,\ x,\ y,\ u,\ v,\ p) = \frac{(h_2 - p')\ [\omega_1(p_0 + mh_2,\ x,\ y,\ u,\ v,\ p - \omega_1(p_0 + (m-1)h_2,\ x,\ y,\ u,\ v,\ p)]}{h_2} +$$

$$+ \omega_1\ (p_0 + (m - 1)\ h_2,\ x,\ y,\ u,\ v,\ p), \qquad (9)$$

dacă funcția ω_1 este crescătoare în raport cu t în intervalul $[p_0 + (m - 1)\ h_2,\ p_0 + mh_2]$, sau după formula

$$\omega_1(p,\ x,\ y,\ x,\ v,\ p) = \frac{p'\ [\omega_1(p_0 + (m-1)h_2,\ x,\ y,\ u,\ v,\ p) - \omega_1(p_0 + mh_2,\ x,\ y,\ u,\ v,\ p)]}{h_2} +$$

$$+ \omega_1\ (p_0 + mh_2,\ x,\ y,\ u,\ v,\ p), \qquad (10)$$

dacă funcția ω_1 este descrescătoare în raport cu t în intervalul $[p_0 + (m - 1)h_2,\ p_0 + mh_2]$, unde $p' = p_0 + mh - p$.

Operatorul de interpolare va calcula valoarea funcției $\omega_1\ (p,\ x,\ y,\ u,\ v,\ p)$, după care va calcula valoarea integralei

$$\int_{p}^{p_0 + mh_2} \omega_1\ (t,\ x,\ y,\ u,\ v,\ p)\ dt \qquad (11)$$

și o va scădea din valoarea expresiei (8).

Pentru a obține soluția q din a doua relație (7) folosindu-ne de programul realizat de schema (S_1) înaintea programului operatorului A_2^i din schema (S_1), vom introduce programul schemei

$$\Phi_1 \left[\begin{matrix} 1 \Rightarrow (i,\ j,\ k,\ s,\ r) \\ (x_i,\ y_j,\ u_k\ v_s,\ p_r) \Rightarrow (x,\ y,\ u,\ v,\ p) \end{matrix}\right] A_1^{(x,y,u,v,p)} \overset{1}{\downarrow} A^{I_1} \times$$

$$\times\ P\ (m > i) \overset{2}{\uparrow} F\ (i)\ O \overset{1}{\uparrow} \overset{1}{\downarrow} \overset{2}{\downarrow} A^{I_2} O \overset{s_1}{\uparrow},$$

unde $A^{I_1} \equiv \int_{p_0+(i-1)h_2}^{p_0+ih_2} = \omega_1\ (t,\ x,\ y,\ u,\ v,\ p)\ dt,$

A^{I_2} este operatorul care calculează valoarea funcției în punctul p, conform relațiilor (9) sau (10), precum și integrala (11), pe care o scade apoi din valoarea expresiei (8), iar Φ_1 operatorul care execută următoarele înlocuiri în schema (S_1): $\psi_1 \Rightarrow \psi$, $\lambda_0 \Rightarrow \omega_0$, $h_3 \Rightarrow h$.

Celelalte ecuații din relațiile (7) se rezolvă cu ajutorul aceluiași program care se completează cu un număr de subprograme corespunzător numărului de operatori A^{I_1} și A^{I_2} egal cu numărul integralelor noi care apar în membrul doi al acestor relații. Valorile C_i se calculează după un program asemănător cu programul schemei S_2, în care lipsește ultima condiție falsă, rezultatele folosindu-se de data aceasta la calcularea expresiei (4). Calculele executate de noi la MECIPT-1 au urmat calea descrisă mai sus, în sensul că s-a rezolvat fiecare ecuație separat, pentru toate sistemele admisibile (x, y, u, v).

În încheiere subliniem că rezolvarea în ansamblu a problemei după un singur program ar necesita o memorie operativă a mașinii de cîteva mii de celule. Din această cauză, programele au fost întocmite pe blocuri, după schemele descrise mai sus, în așa fel ca rezultatele unui program intermediar să constituie date inițiale pentru programul următor.

În felul acesta, rezultatele ultimului program sînt și rezultatele problemei.