MAGYAR TUDOMÁNYOS AKADÉMIA
Számítástechnikai Központja

TÁJÉKOZTATÓ

6

BUDAPEST, 1961
JÚNIUS

MAGYAR TUDOMÁNYOS AKADÉMIA
Számítástechnikai Központja

TÁJÉKOZTATÓ

6

BUDAPEST, 1961
JÚNIUS

E szám munkatársai:

Bóka András, az MTA Számitástechnikai Központ Számológépkutatási osztály vezetője, mb. igazgatóhelyettes

Dancs István, az MTA Sz.K. tudományos munkatársa

Dénes József matematikus, Budapest

Frey Tamás kandidátus, az MTA Sz.K. elméleti osztályának vezetője

Kiss Imre, az MTA Sz.K. tudományos munkatársa

Kornai János kandidátus, az MTA Sz.K. külső munkatársa

Ladányi József, az MTA Sz.K. munkatársa

Németh Pál, az MTA Sz.K. tudományos segédmunkatársa

Podhradszky Sándor, az MTA Sz.K. tudományos munkatársa

Révész György, az MTA Sz.K. tudományos munkatársa

Szentiványi Tibor, az MTA Sz.K. tudományos munkatársa

Vasvári György okl. vill.mérnök, Budapest

Veidinger László, az MTA Sz.K. tudományos munkatársa

Szerkesztette:

Dancs István

Lektorálta:

Pataky Ernő

Sajtó alá rendezte:

Kanics Lászlóné

Felelős szerkesztő és kiadó:

Aczél István dr.

Kiadvány száma: S-534
Alak: A/4 Ivszám: 16.75
Megjelent 1961. julius hóban, 55o példányban

Készült az ÉTI Rotaprint üzemében
F.v. Hordós István

Előző Tájékoztatónkat teljesen az M-3 elektronikus számológép hasznositásával és üzemeltetésével kapcsolatos tapasztalatok ismertetésének szenteltük. A közölt cikkekkel elsősorban az elektronikus számológépeknek a tudományos, müszaki és gazdasági számitások területén való jelentőségére kivántunk rámutatni. A jelen számban ugyancsak elsődleges szerepet játszik az M-3 elektronikus számológép, de most már csak az érdekesebb, eredetibb ötletet tartalmazó feladatok megoldását közöljük. A szinvonal emelése érdekében olyan cikkeket is közlünk, amelyek munkatársaink önálló eredményeit tartalmazzák a numerikus matematika, numerikus analizis területén. Nagyobb teret adtunk a számológépkutatással kapcsolatos munkáink leirásának, ezen belül első alkalommal számolunk be az M-3 gépen végzett müszaki jellegü fejlesztéseink egy részéről. Végül pedig foglalkozunk külföldi eredmények, gépek ismertetésével is; a jobb tájékoztatás érdekében az utóbbi rész terjedelmét a jövőben még növelni szándékozunk.

SZERKESZTŐ BIZOTTSÁG

TARTALOMJEGYZÉK

A PAMUTSZÖVŐ IPARÁG OPTIMÁLIS BERUHÁZÁSI TERVÉNEK MEGHATÁROZÁSA LINEÁRIS PROGRAMOZÁSSAL

Kornai János - Frey Tamás

A pamutszövő iparág optimális beruházási tervének meghatározására készült lineáris programozási modell a rekonstrukció, a részleges korszerüsités és az uj géptipusok fajtáinak és mennyiségének vonatkozásában ad a széles körben vitatott kérdésekre választ. A feladat M-3 gépen való matematikai megoldása a jelenleg rendelkezésre álló relative kisméretü memóriával a szimplex módszer egy általunk módositott variánsa alapján történt, fixpontos módszerrel.

A Textilipari Kutató Intézet a Könnyüipari Minisztérium megbizásából vizsgálatot végzett a pamutszövő iparág II. ötéves beruházási tervének meghatározására.

A számitás céljaira egy lineáris programozási modellt állitottunk fel. Tudomásunk szerint elsőizben történt, hogy matematikai programozás segitségével vizsgálták a beruházások gazdaságosságát. Sőt, ezen tulmenően: ez volt az első nagyobbméretü, elektronikus számológépet igénylő általános lineáris programozási probléma, amelynek numerikus számitása és közgazdasági értékelése befejeződött. /Szállitási problémák megoldására már korábban is sor került./

A m o d e l l

1/ A programozás köre. A számitás kiterjedt a Pamutipari Igazgatóság összes szövődéire. Ezen belül foglalkoztunk az un. sima és vetüléktarkázó gépekkel, amelyek együttesen a szövőgéppark 97 %-át teszik ki, továbbá a szövőelőkészités legfontosabb müveleteivel, a kereszt- és vetülékcsévéléssel, az irezéssel és felvetéssel. Az állóeszköz-állomány néhány kisebb jelentőségü részét számitáson kivül hagytuk.

2/ A döntési problémák. Modellünket ugy szerkesztettük meg, hogy az a következő, széles körben vitatott kérdésekre adjon választ:

- A meglévő gyárakban hajtsunk-e végre rekonstrukciót, vagy épitsünk uj gyárakat?

- A jelenlegi régi gépeket megtartsuk-e, vagy kisebb beruházással hajtsunk végre rajtuk részleges korszerüsitést, vagy pedig szereljük le a régi gépeket?

- Milyen uj géptipusokat szerezzünk be, s egy-egy tipusból mennyit.

3/ A változók. A modellben 43 olyan változó van, amely valamilyen gazdasági tevékenységet reprezentál. /Pl. régi irezőgép továbbmüködtetése; régi irezőgép leszerelése; uj keresztcsévélőgép beállitása régi épületbe; uj keresztcsévélőgép beállitása uj épületbe, s ezzel együtt a szükséges épülettér megteremtése, stb/

Ezenkivül 7 segédváltozóra volt szükség, a közgazdasági tartalmuknál fogva egyenlőtlenség formájában megadott feltételek egyenlőséggé alakitásához.

A számitás eredményeképpen kapott program meghatározza a pamutszövő iparág 1965. évi állóeszköz-állományát. Megszabja, milyen legyen a géppark összetétele 1965-ben; továbbá: mennyi uj épületteret létesitsünk a régi épülettér kiegészitésére.

4/ A feltételek. Ezek a következő főbb csoportokba sorolhatók:

a/ A pamutszövő iparág extern termelési kötelezettségei.

b/ A pamutszövő iparág rendelkezésére álló beruházási erőforrások korlátai.

c/ A pamutszövő iparágon belüli vertikális arányosságát biztositó anyagmérlegek.

d/ Olyan feltételek, amelyek a korábbi időszaktól örökölt adottságokat fejezik ki, /A rendelkezésre álló, a régi üzemekben lévő épülettér. A régi géppark egyes gépcsoportjainak állománya./ A modellben összesen 24 feltétel szerepelt.

5/ A célfüggvény. Azt a programot tekintjük optimálisnak, amely az előirt feltételeket minimális költséggel teljesiti. A költségek közé csupán azokat a ráforditásokat soroltuk, amelyek adott termelési volumen /a megfelelő feltételekben megszabott extern termelési kötelezettség teljesitése/ mellett a géppark összetételétől, s általában a beruházási programtól függnek.

A különböző időpontokban felmerülő költségek összegezésére kétféle formulát alkalmaztunk;

a/ Az Országos Tervhivatal által hivatalosan előirt formulát: az évi üzemeltetési költségekhez hozzáadjuk az egyszeri beruházási költség 2o %-át. /"Egyszerü kamat"./

b/ A diszkontálás módszerét. Ez esetben egy 25 éves elszámolási időszak összes beruházási és üzemeltetési költségeit összegezzük, felmerülésük éve szerint diszkontálva.

6/ <u>A számitás változatai</u>. A számitást 8 változatban végeztük el. /Ezenkivül további változatokat számoltunk egy "kicsinyitett" modell segitségével, amelyhez már nem kellett elektronikus számológépet igénybevenni./

A számitási változatok modelljei a következőkben térnek el egymástól:

- Egyes változatokban alacsonyabb, másokban magasabb termelési kötelezettség teljesitését irtuk elő.

- Egyes változatokban az egyszerü kamatozásu formulát, másokban a diszkontálási formulát alkalmaztuk.

- Egyes változatokban a hivatalosan előirt, másokban közgazdasági megfontolások alapján korrigált devizaárfolyammal számoltunk.

<u>A numerikus számitásnál alkalmazott algoritmusról és gépi programról</u>.

A numerikus számitásokat a Számitástechnikai Központ végezte el, az M-3-as gépen.

A lineáris programozási feladatok M-3 gépen történő lefuttatása több sulyos problémát vetett fel, amelyek nagyrészét sikerült megoldanunk. Első helyen állt ezek között az operativ memória relative kicsiny voltának kérdése. A szóbajövő matematikai módszerek közül a szimplex-eljárás egy általunk módositott változata látszik legelőnyösebbnek a rendelkezésre álló operativ memóriakapacitás optimális kihasználása szempontjából. E módszer lényege: a lineáris programozási feladat egyenlőtlenség-alaku mellékfeltételeit segédváltozók bevezetésével egyenletekké alakitjuk, majd az igy nyert mellékfeltétel-rendszer átrendezése, ismeretlenjeinek átszámozása, sorainak és oszlopainak[1/] alkalmas faktorokkal történő megszorzása utján elérjük, hogy az

$$\underline{\underline{A}}\,\underline{x} = \underline{b} \text{ azaz } (\underline{\underline{B}} + \underline{\underline{C}})\underline{x} = \underline{b}$$

alakban particionált $\underline{\underline{A}}$ matrix baloldali, $\underline{\underline{B}}$ jelölt kvadratikus minorja numerikusan is jól invertálható legyen /azaz det $\underline{\underline{B}}$ legalább olyan nagyságrendü legyen mint $\underline{\underline{B}}$ maximális eleme/. $\underline{\underline{B}}^{-1}$-gyel megszorozva az egyenletet

$$(\underline{\underline{E}} + \underline{\underline{B}}^{-1}\underline{\underline{C}})\underline{x} = \underline{\underline{B}}^{-1}\underline{b} \text{ azaz } (\underline{\underline{E}} + \tilde{\underline{\underline{B}}})\underline{x} = \tilde{\underline{b}}$$

adódik. Ez az alak általában még nem generál megengedett megoldást, minthogy $\tilde{\underline{b}} \geq \underline{0}$ általában még nem érvényes. Ezért első lépésekként az oszlopvektorokra particionált $\underline{\underline{E}} + \tilde{\underline{\underline{B}}}$ mátrixban olyan oszlopcseréket hajtunk végre - a szokásos szimplex-eljáráshoz hasonlóan - hogy a $\tilde{\underline{b}} \geq \underline{0}$ feltétel kielégitéséhez mennél közelebb juthassunk. E célból sorfolytonosan tároljuk $\tilde{\underline{\underline{B}}}$ elemeit /a mátrixot mindig ugy rendezzük át, hogy a triviális $\underline{\underline{E}}$ matrix az összeg baloldalán szerepeljen főminorként, igy azt ne kelljen sohase tárolni/ - pontosabban szólva egy kiemelt, egynél nagyobb szorzófaktor reciprokát, $\frac{1}{\beta}$ -t és $\frac{1}{\beta}\tilde{\underline{\underline{B}}}$ egynél kisebb elemeit.

Tároljuk ezenkivül a preferenciafüggvény alkalmasan normált együtthatóit, egy-egy identifikálásukra alkalmas sorszámmal közös rekeszben. Utóbbira azért van szükség, mert a végrehajtott oszlopvektorok cseréjével együtt - programozási okokból - célszerü a preferenciaegyütthatókat is cserélgetni, és igy szükséges egyszerü azonositásuk, amely igy egyszersmint a bázisba bevont vektorok azonositására is alkalmas.

A program maga több részből áll; az egyes programrészek bonyolult voltát elsősorban az indokolja, hogy fixpontos módon számolva igen gondosan kell ügyelni arra, hogy a kerekitési hibák

[1/] Az oszlopok valamelyikének, pl. a j-edik oszlopnak az α_j faktorral történő szorzása következményekkel jár: a preferencia-függvényben az x_j ismeretlen c_j együtthatóját és az optimum megtalálása után a bázisban esetleg szereplő x_j értékét is meg kell szorozzuk α_j -vel.

és számjegyveszteségek végső és összesitett hatása ellenére elegendően pontos végeredmények álljanak rendelkezésünkre. A program első része a pillanatnyilag tárolt $\tilde{\tilde{\underline{\underline{B}}}}$ mátrix normálását hajtja végre olymódon, hogy kikeresi ennek abszolut értékre maximális elemét, azt 2^{-30}-cal növeli és ezzel $\tilde{\tilde{\underline{\underline{B}}}}$-t is, a kiemelt $\frac{1}{\beta}$ faktort is elosztja. A második hasonló értelemben felnormálja a tárolt mindenkori $\tilde{\tilde{\underline{\underline{b}}}}$ vektort. A program harmadik része függ attól, hogy már egy megengedett bázisból indulunk-e ki, azaz $\tilde{\tilde{\underline{b}}} \geq \underline{0}$ érvényes-e már, avagy sem. Utóbbi esetben, azaz ameddig nem jutottunk el egy megengedett bázisig a cserék folyamán, a harmadik rész végigtapogatja a $\tilde{\tilde{\underline{\underline{B}}}}$ matrix valamennyi $\tilde{\tilde{b}}_{ij}$ elemét és kiválasztja közülük azt - pontosabban szólva elraktározza annak két indexét - amelynek alapján cserélve /a két index a két cserélendő oszlopvektor helyét szolgáltatja/ a legnagyobb mértékben csökken a csere után adódó $\tilde{\tilde{\underline{b}}}$ vektor negativ elemeinek száma. Az első alkalommal, amikor már megengedett bázisból indulunk ki, e programrész helyébe egy másikat viszünk, amely először kiszámitja az adott $\tilde{\tilde{\underline{\underline{B}}}}$ matrix elemei és a preferenciaegyütthatók segitségével azt, hogy mely oszlopvektorok bázisba vonása csökkenti a preferenciafüggvény értékét, és ezután megkeresi lexikografikus sorrendben azt a legelső $\tilde{\tilde{b}}_{ij}$ elemet, amely a kijelölt oszlopvektorok valamelyikéhez tartozik és ujból megengedett bázishoz vezet. E harmadik programrészekkel kapcsolatosan megemlitjük elsősorban, hogy legalább olyan pontosságu számolásra van szükség itt, mint a tényleges vektorcserénél, mert egyébként a kerekitési pontatlanságok megengedett bázisról ismét meg nem engedett bázisra vihetnének át, ami a számitási időt is, a programot is és a végső pontatlanságot is nagyban növelné. Ezt a pontosságot a részletszámitásoknál szereplő oszlopvektorrészek ideiglenes felnormálásával lehet elérni. Meg kell emliteni végül, hogy a lexikografikus sorrend a tapasztalatok szerint jobbnak bizonyult a csere indexeinek kiválasztására /összesen átlagosan kevesebb cserével lehetett az optimumig eljutni/, mint az az eljárás, amikor az oszlopvektorok kiválasztása a szimplex eljárásra jellemző és az oszlopelemek és preferenciaegyütthatók skalársorozatából adódó jellemzők nagyságrendi sorrendje alapján történt és csak az oszlopon belül használtunk lexikografikus eljárást. Optimálisnak a teljesen véletlenszerü kiválasztási módszer látszik, de ezt memóriakapacitási okokból nem tudtuk kipróbálni. E harmadik programrészben kell vizsgálni végül azt, hogy van-e egyáltalán korlátos megoldása a feladatnak, ill., hogy az nem elfajuló-e?

Végül a program negyedik része a cseréket végzi el. $\tilde{b}_{ij}$-vel jelölve azt az elemet, amelynek alapján cserélünk, az uj $\underline{\underline{B}}$ általános eleme ilyen alaku

$$\tilde{b}_{rs} = \frac{b_{rs}\,\tilde{b}_{ij} - \tilde{b}_{is}\,b_{rj}}{\tilde{b}_{ij}}$$

és hasonlóan adódik az uj $\underline{b}$ vektor elemeinek számitási törvénye is. Az i-edik sor és a j-edik oszlop elemeit hasonló, de a kiemelt szorzófaktorok által módositott formula szolgáltatja. Közös jellemvonása azonban e formuláknak, hogy $\tilde{b}_{ij}$ a nevezőjük. Ez azért használható igen előnyösen fel, mert igy az osztást nem kell elvégeznünk /és igy fixpontos gépen nem kell előzően ezt megvizsgálni, hogy kisebb lesz-e a hányados abszolut értéke 1-nél/, hanem ehelyett a $\tilde{\underline{\underline{B}}}$-ből kiemelt szorzófaktor reciprokát szorozzuk $\tilde{b}_{ij}$ -vel. Minthogy azonban igy a szorzatok különbsége szolgáltatja lényegében az uj elemeket, erősen fenyeget a nagymértékü jegyveszteség. Ezt a kiválasztott j-edik oszlopvektor, azaz a $\tilde{b}_{ij}$ és $\tilde{b}_{rj}$ elemek előzetes közös felnormálásával, aránylag kis müveleti és programtöbblettel csökkenteni lehet, sőt ez tapasztalataink szerint lényegében elegendő is a szokásos 1^{o}/oo-es végső hibakorlát biztositására. Utolsó lépésként a kicserélt oszlopvektoroknak megfelelő preferenciaegyütthatók memóriabeli helyét is megcseréljük, majd visszatérünk az első programrészhez.

A teljes program kb. 42o utasitásból áll, igy minden olyan feladat megoldható a jelenlegi gépkapacitás mellett, ahol a mellékfeltételek számának szorzata a változók száma és a mellékfeltételek száma különbségével nem lépi át a kb. 65o-es küszöböt. A ténylegesen megoldott feladat közel volt a küszöbhöz: 24 mellékfeltétel és 5o változó szerepelt ugyanis. Egy ilyen feladat teljes megoldása géphibamentes futást feltételezve kb. 9,5 óra, amiből kb. 3-at az előkészitő müveletek $\underline{B}$ inverziója, $\underline{B}^{-1}$ és $\underline{C}$ ill. $\underline{B}^{-1}$ és $\underline{b}$ szorzása, továbbá az adatok konverziója /1,5-et a megengedett bázisra való áttérés, 5-öt pedig az optimum megkeresése vesz igénybe.

A memória teljes kihasználtsága miatt a kinálkozó legegyszerübb kontroll-lehetőséget, amely szerint az eredeti $\underline{A}$ matrix megfelelő oszlopvektoraiból álló matrix /olyan sorrendben, mint a pillanatnyilag a $\underline{\tilde{B}}$ -ben nem szereplő oszlopvektorok/ és $\underline{\tilde{\tilde{B}}}$ szorzata eredményül $\underline{A}$ eddig figyelembe nem vett oszlopvektorai matrixát adja, nem tudtuk alkalmazni és igy ténylegesen átlagosan háromszor kellett egy-egy feladatot futtatnunk. A szalagmemória megépülte után lényegesen kedvezőbb lesz ilyen szempontból a helyzet. Azt a további, de nem elég megbizható kontrollt, hogy a $\underline{\tilde{b}}$ vektor mindig az eredeti mellékfeltétel-rendszer egy-egy megoldását generálja, alkalmaztuk, Ez azonban nem mutathatja ki a sokkal nagyobb valószinüséggel $\underline{\tilde{\tilde{B}}}$ számitásánál keletkező, csak a $\underline{\tilde{b}}$ -nél fellépő, vagy az előzőek miatt - több lépéssel később - $\underline{\tilde{\tilde{b}}}$ -re is átterjedő hiba jelenlétét.

A tapasztalatok szerint fixpontos gépeknél a fent leirt eljárást célszerü használni - lehetőleg az imént emlitett kontrollmódszerekkel együtt - lineáris programozási feladatok megoldásánál.

A számitás eredményeiből levonható gyakorlati közgazdasági következtetések

A számitás eredményeképpen kapott optimális programokból fontos gyakorlati következtetéseket szürhetünk le. Ezek közül néhányat emlitünk:

1/ A Minisztériumban a hagyományos tervezési módszerekkel készült eredeti program, amelynek kontroll-számitása volt ez a programozás, nagyarányu épitést irt elő. Ezzel szemben a matematikai programozás alapján azt ajánlottuk, hogy a meglévő üzemek rekonstrukcióját kell előtérbe helyezni, s az épitést a minimumra kell szoritani. Az épitési keretet szinte teljesen kihasználatlanul lehet hagyni, a az igy felszabadult összeg a termelékenységet emelő gépekre forditható.

2/ A régi gépek nagyobb részét kell leszerelni, s uj korszerü gépekkel pótolni, mint amennyit az eredeti program tervezett. Ilymódon felszabadul a leszerelendő gépek nagyjavitására szánt pénz, ami ismét uj gépek beszerzésére forditható.

3/ Ez az átcsoportositás lehetővé teszi, hogy a technikai korszerüsités, az automatizálás utján gyorsabban menjünk előre, mint ahogy azt az eredeti terv előirta.

4/ Vita volt a pamutiparban arról, hogy a különböző alternativ géptipusok közül melyiket érdemes beszerezni. A programozás eredményei módot adnak több ilyen vitatott kérdés eldöntésére.

5/ Végeredményben a programozás - a diszkontálást alkalmazó célfüggvényben szereplő költségek szerint számolva - kb. 15%-os megtakaritást hozott a hagyományos módszerekkel készült eredeti tervhez képest. /Ez magában foglalja mind a beruházási, mind az üzemeltetési költségeknél elért megtakaritást./

A programozás - a pamutiparban közvetlenül hasznositható eredményeken tulmenően - általánosabb tanulságokat is hozott.

Egyrészt: az eltérő kamatozási formákkal, eltérő kamatlábbal és eltérő devizaárfolyammal végrehajtott szimultán programozások összehasonlitása fontos támpontokat adhat a felső gazdasági szervek számára annak eldöntéséhez: milyen kamatlábat és devizaárfolyamot célszerü alkalmazni a beruházás-gazdaságossági számitásokban.

Másrészt: általánosithatók a programozás során nyert tervezési-metodikai tapasztalatok. Természetesen az itt alkalmazott modellt nem lehet mechanikusan, változtatás nélkül használni más területeken. A gyakorlati tapasztalat azonban bebizonyitotta, hogy a számitás koncepciója, általános módszertani elgondolása - <u>programozási</u> modell segitségével végzett beruházás - gazdaságossági számitás - a népgazdaság más ágaiban is jól alkalmazható.