

V. Bonačić

automatika

SEPARAT

Redukcija i sortiranje podataka eksperimenta računalom

DK 681.14-523.8
IFAC IA 4.5.1

Napisani su programi za redukciju i sortiranje podataka koji se tokom eksperimenta buše na papirnoj traci. Svaki podatak predočen je kao par binarnih brojeva koji predstavljaju dvije varijable u eksperimentu (X, Y). Računalo sortira podatke po varijablama i mjeri njihovu gustoću vjerojatnosti. Rezultate sortiranja računalo otiskava na list papira. X i Y varijablama pridružene su koordinatne osi, a na svakom sjecištu odštampana je gustoća vjerojatnosti dotičnog para. Problem je riješen u strojnom jeziku na laboratorijskom računalu PDP-8, a također i u »real time« fortran jeziku na računalu CAE 90-40 (SDS 930).

Reduction and Sorting of Experimental Data by Computer. The programmes for data reduction and sorting have been written. During the experiment these data are punched onto paper tape. Every item of information has the form of two binary numbers which can be treated as two experimental variables (X, Y). The data are sorted by computer according to their amplitudes (i.e. variables X, Y) and their probability distribution is measured. Sorting results are printed onto a strip of paper. Coordinate axes are associated with the X and Y variables. At the cross points of every pair the corresponding numbers of probability distribution are printed. The problem has been solved in the symbolic language using the laboratory computer PDP-8 and in the »real time« fortran language using the CAE 90-40 (SDS 930) computer.

1. Uvod

Za bolje razumijevanje članka sugerira se čitaocu da pročita »Digitalna računala u mjerenju i regulaciji« u istom časopisu, jer autori polaze od pretpostavke da su čitaoci upoznati s terminologijom i elementarnim operacijama u računalu.

Rezultati niza fizikalnih eksperimenata registriraju se kao vremenski slučajno raspodijeljeni parovi brojeva (1, 2, 3, 4). Takve rezultate daje npr. dvodimenzionalna analiza koincidentnih impulsa.

Podatke dobivene kao rezultat eksperimenta u obliku parova brojeva registriranih na bušenoj papirnoj traci treba sortirati i rezultat sortiranja odštampati na papiru tako, da brojevi pridruženi svakoj tački koordinatnog sistema x, y predstavljaju gustoću vjerojatnosti ili broj ponavljanja istog para. Pri tom su na osi koordinatnog sistema x, y nanese moguće vrijednosti obaju brojeva iz para.

2. Pregled mogućih načina sortiranja

Sortiranje podataka zapisanih na perforiranoj traci moguće je izvršiti ili gradnjom specijalnog elektroničkog uređaja (»hardware«) ili — ako postoji digitalno računalo — pisanjem posebnog programa (»software«).

Primjer upotrebe prvog načina je elektronički uređaj (5), koji koristi postojeću memoriju 256-kanalnog amplitude analizatora. Upotrebljena traka ima sedam mjesta za oznaku informacije, a osmo za indikaciju o kojem se broju iz para radi. U tom je slučaju broj mogućih deskriptora (deskriptor je ovdje spomenuti par brojeva koji opisuje energiju nuklearnog događaja) $128 \times 128 = 16364$, pa traku treba čitati 64 puta i nakon svakog čitanja sortirane podatke — koji se privremeno nalaze u 256-kanalnoj memoriji — štampati na papir kao dva stupca brojeva gdje je svakom deskriptoru (lijevi stupac) uvijek u istom retku pridružena njegova gustoća vjerojatnosti (desni stupac). Da je upotrebjeno samo šest mjesta za informaciju, ovaj sistem bi polje od $64 \times 64 = 4096$ deskriptora čitao 16 puta.

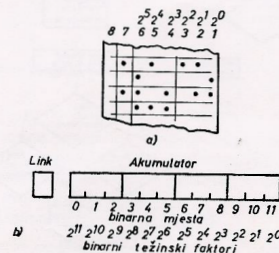
Upotreba digitalnog računala pruža niz prednosti za automatsko sortiranje podataka. To omogućuje veća memorija i fleksibilnost računala. Obično je tada dovoljno traku čitati jedan ili dva puta, a programom se može

postići štampanje rezultata u mnogo prihvatljivijoj formi za korisnika. Sortirane podatke, koji se privremeno nalaze u memoriji, moguće je prenijeti direktno u binarnom obliku na papirnu traku za eventualnu daljnju obradu digitalnim računalom.

3. Format perforirane trake s podacima za sortiranje na računalu PDP-8

Program za sortiranje podataka ovisi o formatu perforirane trake, a ne o tipu fizikalnog eksperimenta ili elektroničkog sistema za registraciju.

Upotrebjeno digitalno računalo PDP-8 koristi standardnu papirnu traku na kojoj postoji mogućnost bušenja osam rupa u retku. Kako memorija i registri imaju dužinu riječi 12 bita, vrlo je prikladno koristiti šest mjesta za informaciju u jednom retku papirne trake, a sedmo za indikaciju početka para. Memorija PDP-8 ima 4096 riječi, no zbog programa, koji zauzima par stotina lokacija, samo 2048 lokacija uzeto je za podatke. Polje od 4096 (10000 u oktalnom sistemu) deskriptora podijeljeno je u dvije grupe, a primjeri deskriptora iz obje grupe vidi se na



Sl. 1. a) Primjer deskriptora izbušenog na papirnoj traci
b) Organizacija akumulatora

slici 1 a. Kolonama 1 do 6 pridijeljeni su težinski faktori 2^0 do 2^5 u binarnom brojnem sistemu, a grupe od po tri binarne znamenke uzete zajedno daju jednu oktalnu. Ako se mjesta 6, 5 i 4 sjedine u prvu oktalnu znamenku, a 3,

2 i 1 u drugu i postupak ponovi s drugim retkom iz para dobiva se deskriptor 2641₉ (indeks znači bazu brojnog sistema). Drugi par predstavlja deskriptor 5370₉. Spomenute dvije grupe deskriptora (prva: 0 do 3777, druga: 4000 do 7777) razlikuju se u šestom binarnom mjestu prvog broja iz para, koje je kod druge grupe uvijek izbušeno, a kod prve to nikad nije slučaj. Svaki broj iz para može poprimiti vrijednosti 0 do 63, tj. 0 do 77₉.

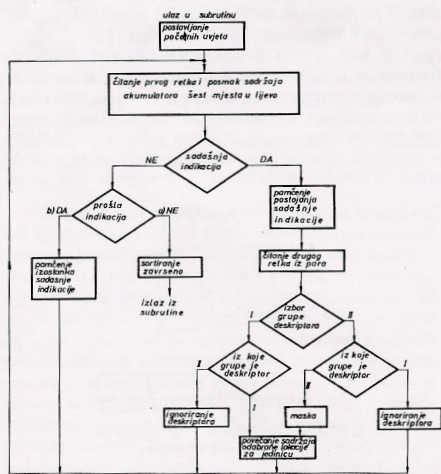
Sedam mjesta u retku upotrebljenih za informaciju dalo bi polje od $128 \times 128 = 16384$ deskriptora. Uz pretpostavku da je gustoća vjerojatnosti za veliki broj deskriptora veća od $2^6 = 64$, što onemogućava upotrebu jedne lokacije memorije za dva deskriptora, postoje dvije mogućnosti sortiranja:

a) na isti način kao kod sortiranja 4096 deskriptora uz povećani broj čitanja za faktor 4;

b) napisati program za asocijativnu analizu (6, 7), koja bazira na činjenici da se u cijelom mogućem polju deskriptora stvarno dogodi samo mali broj događaja. Taj program ne osigurava unaprijed lokacije memorije za cijelo polje nego samo za one deskriptore koji se stvarno dogode, pa prema tome mnogo bolje ekonomizira s prostorom memorije. No program za asocijativnu analizu je mnogo komplikiraniji i ima smisla samo tamo gdje veličina memorije predstavlja nekoliko postotaka od veličine mogućeg polja deskriptora (kod 64×64 je to 100%, a kod 128×128 je 25%).

4. Opis strojnog programa za sortiranje računalom PDP-8

Program za sortiranje napisan je u unutarnjem ili strojnom jeziku digitalnog računala PDP-8. Osnovna prednost prema programu u vanjskom jeziku (npr. Fortran) je veći raspoloživi prostor u memoriji za podatke i veća



Sl. 2. Dijagram toka programa za čitanje i sortiranje podataka u strojnom jeziku računala PDP-8

brzina izvršavanja programa. Podaci se čitaju pomoću čitača i ulaze u osnovni 12-bitni aritmetički registar — akumulator (slika 1 b). Pri tom podatak o indikaciji ulazi

u pomoćni jednobitni registar — link (veza). Na kraju svakog ciklusa programa uzima se deskriptor, tj. sadržaj akumulatora kao ona adresa memorije kojoj treba sadržaj povećati za jedinicu. Princip sortiranja podataka je isti kao kod amplitudnog analizatora, a osnovna razlika je konstantnost repeticije ulaze podataka.

Dijagram toka za program čitanja i sortiranja podataka vidi se na slici 2. Taj program predstavlja subrutinu, koja je sastavni dio jednog posebnog — glavnog programa. Nakon ulaza u subrutinu postavljaju se početni uvjeti, a čitač čita prvi red papirne trake. Zatim se on upiše u akumulator na mjesta 5 do 11 i izvrši posmak sadržaja akumulatora za šest mjesta u lijevo (od manje značajnih prema značajnijim mjestima). Podatak o indikaciji koji je prilikom čitanja bio na petom mjestu dolazi nakon posmaka u link (jer su link i akumulator kod posmaka vezani kao jedan registar), a informacija dolazi na mjesta 0 do 5.

Slijedeći korak u programu je test, koji provjerava da li postoji »sadašnja indikacija«, tj. da li je izbušeno sedmo mjesto u retku koji je upravo pročitan. Nepostojanje indikacije, dakle negativni rezultat testa može imati jedan od dva uzroka: a) sortiranje završeno — kraj trake, b) čitanje je pogrešno započeto s drugim brojem iz para ili se radi o greški čitača ili bušača, dakle izgubljeni sinhronizam. Odluku o tome radi li se o slučaju a) ili b) donosi slijedeći test: »prošla indikacija«. Taj test ispituje da li je u prethodnom koraku postojala indikacija. Ako je test negativan (slučaj a), što znači dva uzastopna izostanka indikacije, čitanje i sortiranje trake je završeno i s ovog programa automatski se izlazi iz subrutine i prelazi na tipkanje rezultata sortiranja. Ako je posljednji test pozitivan (slučaj b), što znači da je u prošlom koraku postojala indikacija, treba zapamtiti da je test »sadašnja indikacija« bio negativan. To se radi upisivanjem nule u pomoćnu lokaciju memorije. Zatim program ignorira pročitani redak i prelazi na čitanje slijedećeg, koji, ako se ne radi o kraju čitanja, mora imati indikaciju.

Ako je nakon čitanja prvog retka test »sadašnja indikacija« pozitivan, u pomoćnu lokaciju se upisuje jedinica, a zatim se čita drugi redak u paru i zapisuje na mjesta 6 do 11 akumulatora.

Nakon što je cijeli deskriptor smješten u akumulator program testira najprije koju grupu podataka želimo sortirati, a zatim iz koje grupe je stvarno podatak došao. Prvi test ispituje u kojem se položaju nalazi jedan posebnih preklapnik*, a drugi ispituje nulti bit akumulatora.

Podatak koji se nalazi u akumulatoru ignorira se ako je došao iz suprotne grupe onoj koju želimo sortirati prilikom tog prolaza trake kroz čitač. Program se ponavlja, a postupak čitanja nastavlja s drugim parom.

Ako je podatak iz željene grupe, on se uzima kao adresa kojoj se povećava sadržaj za jedinicu. Kod prve grupe to se radi direktno, a kod druge — budući da obje grupe podataka ulaze u iste lokacije memorije — prije adresiranja koristi se maska, koja postavlja nulti bit akumulatora u nulu. Nakon povećanja sadržaja odabrane lokacije za jedinicu, ciklus programa se ponavlja čitanjem slijedećeg para.

* Bilo koji od 12 konzolnih preklapnika [»switch« registra] moguće je ručno postavljati u stanje 0 ili 1 po želji. Nulti preklapnik u gornjem položaju znači nulti bit konzolnog registra u stanju 1.

Program za tipkanje rezultata u dekadskom obliku duži je i komplikiraniji od programa čitanja i sortiranja. No on nije detaljno opisan, jer nije vezan za samo sortiranje nego za teleprinter, koji štampa rezultat sortiranja i za format izlaza podataka. Rezultati sortiranja tipkaju se na papiru približno kvadratnog oblika. Na svaku od koordinatnih osi nanesene su sve moguće vrijednosti od 0 do 63 obaju brojeva iz para, a u koordinatni sistem — gustoća vjerojatnosti. Slika 3 pokazuje rezultate sortiranja

5. »Real time« fortran program za sortiranje podataka računalom CAE 90-40

Program za redukciju i sortiranje podataka napisan je također i u »real time« fortran jeziku za digitalno računalo CAE 90-40 (SDS 930).

To računalo ima veću memoriju od računala PDP-8, brži čitač bušene trake i mnogo kompleksniju logiku (»hardware«). Zbog kompleksnije logike posjeduje mnogo

DVODIMENZIONALNA ANALIZA		01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28										
01		1											1	1	1				2																				
02			1											1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1										
03				1											1	1	1	1	1	1	1	1	1	1	1	1	1	1	1										
04					1											1	1	1	1	1	1	1	1	1	1	1	1	1	1										
05						1											1	1	1	1	1	1	1	1	1	1	1	1	1										
06							1											1	1	1	1	1	1	1	1	1	1	1	1										
07								1											1	1	1	1	1	1	1	1	1	1	1										
08									1											1	1	1	1	1	1	1	1	1	1										
09										1											1	1	1	1	1	1	1	1	1										
10											1											1	1	1	1	1	1	1	1										
11												1											1	1	1	1	1	1	1										
12													1											1	1	1	1	1	1										
13														1											1	1	1	1	1										
14															1											1	1	1	1										
15																1											1	1	1										
16																	1											1	1										
17																		1											1										
18																			1											1									
19																				1											1								
20																					1											1							
21																						1											1						
22																							1											1					
23																								1											1				
24																									1											1			
25																										1											1		
26																											1											1	
27																												1											1

Sl. 3. Rezultati sortiranja podataka dvodimenzionalne analize

podataka jednog stvarnog fizikalnog eksperimenta. Na slici je pokazan samo jedan dio cijelog izlaznog formata. Kako se štampaju dekadski brojevi treba prije štampanja izvršiti konverziju binarnog sadržaja memorije u dekadski. Program za tu konverziju vrši najprije odbijanje binarnog ekvivalenta broja 1000 od binarnog broja, koji predstavlja sadržaj lokacije, kojeg treba otipkati. To odbijanje vrši se dokle god je sadržaj veći od 1000, a broj odbijanja odštampa se kao prva decimalna znamenka. Zatim se od ostatka odbija binarni ekvivalent broja 100 itd. Program osigurava da se same nule ili nule ispred vrijednih znamenaka ne štampaju dok se vrijedne znamenke i nule iza njih štampaju.

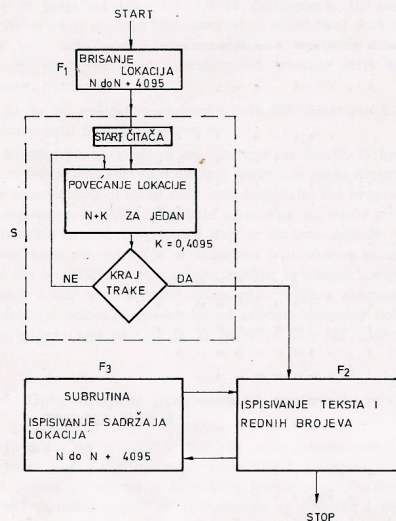
veći repertoar strojnih instrukcija. Neke od njih mogu izvršavati operacije za koje kod računala PDP-8 treba pisati mali program. Uz podjednako trajanje osnovnog ciklusa ovo računalo omogućuje mnogo brže rješavanje kompleksnih matematskih problema.

Pomoću strojnih instrukcija program može izvršavati razne »mikrooperacije« s riječima ili čak dijelovima riječi — bitovima.

Komande u fortran jeziku imaju makrostrukturni karakter, pa se komplicirane operacije jednostavno i kratko programiraju. Ali zato je vrlo nespretno pomoću fortran komandi pisati program za izvršavanje nekih mikrooperacija.

»Real time« fortran program ujedinjuje prednosti strojnog i fortran jezika: fleksibilnost strojnog i jednostavnost fortrana. Zato je on istovremeno i fleksibilan i jednostavan za programiranje.

Slika 4 prikazuje dijagram toka događaja programa za dvodimenzionalnu analizu napisanog u »real time« fortran jeziku. Dijelovi programa F_1 , F_2 i F_3 napisani su u fortranu, a S u strojnom jeziku.



Sl. 4. Dijagram toka programa u »real time« fortranu

Prvi dio F_1 ovog programa briše za podatke predviđeni prostor od 4096 lokacija memorije počevši od startne lokacije N . On izgleda ovako:

```
DO 10 I = 1,4096
10 N(I) = 0
```

Slijedeći dio programa S u strojnom jeziku izvršava redukciju i sortiranje. Pročitani par brojeva čitačem pretvara se u jedan deskriptor čija je numerička vrijednost K . Kod dvodimenzionalne 64×64 analize K može biti broj od 0 do 4095. Dobiveni deskriptor K određuje adresu, kojoj se sadržaj povećava za jedinicu. Taj dio je ovdje bitno kraći nego program opisan u tački 4 upravo zbog spomenute razlike između strojnih instrukcija obaju računala.

Rezultati analize ispisuju se u istom obliku kao i kod računala PDP-8, a prikazani su na slici 3.

Ispisivanje teksta »DVODIMENZIONALNA ANALIZA, BETA, GAMA« i rednih brojeva 0 do 63 na vertikalnoj osi (y varijabla) izvršava dio programa F_2 :

```
TYPE 1
1 FORMAT ($ DVODIMENZION $, $ ALNA ANALIZA $,
$ BETA GAMA $)
L = 0
DO 20 L = 0, 63
20 TYPE 2, L
2 FORMAT (12)
```

Dio programa F_3 štampa redne brojeve 0 do 63 na horizontalnoj osi (x varijabla) i sadržaj lokacija (dakle gustoću vjerojatnosti) N do $N + 4095$, prema slici 3.

6. Zaključak

U ovom članku opisana su dva načina redukcije i sortiranja podataka. Jedan koristi računalo PDP-8 i strojni program, a drugi računalo CAE 90-40 i »real time« fortran program. Razlike između jednog i drugog rješenja mogu se promatrati s obzirom na utrošak vremena

- a) računala
- b) programera.

Korisnik je zainteresiran za skraćivanje vremena, kroz koje je okupirano računalo i programer. Jedan razlog za to je cijena usluge, koju treba platiti. Ona je proporcionalna potrošnom vremenu računala i programera. Drugi razlog je taj, što korisnik želi u najkraće mogućem roku dobiti rezultate eksperimenta sortirane i ispisane u obliku prihvatljivom za daljnju analizu.

Vrijeme okupiranosti stroja ovisi o veličini memorije i brzini perifernih jedinica.

PDP-8 ima malu memoriju i zahtijeva učitavanja trake dva puta. Veća memorija CAE 90-40 omogućuje sortiranje nakon jednokratnog čitanja trake.

Brzi čitač uz računalo CAE 90-40 čita ulazne podatke 30 puta brže i znatno skraćuje vrijeme čekanja na interpretaciju rezultata eksperimenta.

Svojstva jezika, kojim se programer prilikom programiranja služi, određuju vrijeme potrebno za pisanje programa.

Zbog kompleksne logike računala CAE 90-40, dio programa pisan u strojnom jeziku znatno je kraći.

»Real time« fortran jezik jednostavan je i bogat po mogućnostima. On omogućuje bitnu uštedu u vremenu pisanja programa. Za pisanje spomenutog programa redukcije i sortiranja podataka u strojnom jeziku PDP-8 potrebno je nekoliko dana, a za rješenje istog problema u »real time« fortranu nekoliko sati.

Kod računala CAE 90-40 lakše je preći s dvodimenzionalne analize 64×64 , dakle polja od 4096 deskriptora, na veće. Tome doprinosi i činjenica, što postojeća konfiguracija uz CAE 90-40 sadrži i sistem s magnetskom trakom.

Na kraju valja naglasiti da su ovdje komparirana dva sistema bitno različita u kompleksnosti. Zato im je i odnos cijena otprilike 1:10.

LITERATURA

1. V. Valković, I. Šlaus, P. Tomaš, M. Cerineo: The Study of the Reactions $^{10}\text{B}(n, d)_{\alpha\alpha}$, $^7\text{Li}(n, d)_{\alpha n}$ and $^6\text{Li}(n, d)_{\alpha n}$ at 14.4 MeV. Nuclear Physics, 1967 (A98) 305–322.
2. G. Pačić, I. Šlaus, P. Tomaš: Counter Telescopes for the Study of (n, charged particle) Reactions. Nuclear Instruments and Methods, 1965 (34) 40–44.
3. L. G. Kuo, M. Petračić, B. Turko: A dE/dx-E Counter Telescope for Charged Particles Produced in Reactions with 14 MeV Neutrons. Nuclear Instruments and Methods, 1961 (10) 53–65.
4. B. Turko: Sistemi za višedimenzionalnu amplitudnu analizu. Zbornik materijala I međunarodnog skupa elektronike i automatike, Hercegnovi, oktobar 1966, 44–50.
5. M. Konrad, B. Turko: Punched Tape Data Sorting Using a 256-Channel Memory. Elektrotehnika, 1965 (4) 254–256.
6. G. C. Best: Associative Storage with a Small Computer. IEEE Transactions on Nuclear Science, June 1966 (NS-13) 566–570.
7. B. Souček: Stored Program Computer as an Associative Radiation Analyzer. The Review of Scientific Instruments June 1965 (36, No 6) 750–753.

NASLOV AUTORA:

K. Čuljat — V. Bonačić — B. Souček, Institut »Ruđer Bošković«, Zagreb