

**Knihy došlé do redakce
(Books received)**

Information Processing Machines 22 (*Jiří Raichl, ed.*). Academia, Prague 1982. 288 pages; 75 figs; Kčs 60,—.

Roman Bek: Sémantika přesného popisu reality ve fyzikálně matematických vědách. Academia, Praha 1982. 132 stran; 19 obr.; Kčs 25,—.

Leonard Gillman, Robert H. McDowell: Matematická analýza (2. nezměněné vydání). (Překlad z anglického originálu: Calculus, W. W. Norton and Company, Inc. 1973.) SNTL — Nakladatelství technické literatury, Praha 1983. 608 stran; 324 obr., 2 tab.; Kčs 59,—.

Gunter Schlageter, Wolfried Stucky: Datenbanksysteme: Konzepte und Modelle (2., nebearbeitete und erweiterte Auflage). (Leitfäden der angewandten Mathematik und Mechanik LAMM 37.) B. G. Teubner, Stuttgart 1983. 368 Seiten; 70 Abbildungen; DM 29,80.

PHILIPPE COIFFET

Modelling and Control

Robot Technology Series Volume 1.
Kogan Page, London 1983.
156 pages; 71 illustrations; £ 22.50

The reviewed book represents an English translation of the original French Coiffet's text which appeared in 1981; the author works with the Automation Laboratory, Univ. of Montpellier, France. His monography deals, substantially, with already existing or being just developed industrial robots and manipulators, which are subsumed under a general notion of "articulated mechanical systems". Various theories and methods are presented how to describe and model the statics, kinematics and dynamics of such systems, together with methods how to control the mechanical motion of the robot's articles in order to fulfil given mechanical tasks (manipulation with objects, simple mechanical constructions, etc.). The

interaction with the environment is investigated only at the lowest level and is supposed to be an integral part of the hardware. The books explaining in more details this part of theoretical robotics are not too numerous and it is why Coiffet's work should be taken as an interesting and useful contribution. As a matter of fact, the greatest part of monographies or other papers in the domain of robotics deal either with the applied robotics, supposing the technical and construction problems having been solved, or they deal with the more intelligent cognitive and decision-making functions of "higher-generation" robots from the point of view of artificial intelligence, with the hardware problems being a priori eliminated from the field of investigation.

Let us briefly review the contents of the book, which consists of ten chapters. Chapter 1 offers a very general description of an industrial robot in the terms of system engineering. Several methods which describe and classify the abilities of particular robots with respect to the degrees of freedom of their articles can be found in Chapter 2. Various possibilities are represented by numerical (Roth and Piper), matrix (Khalil), graph (Borrel) and symbolic (Renard-Zabala) descriptions with their advantages and disadvantages briefly presented. Chapter 3 introduced rules and equations enabling to compute the positions and orientations of robot's articles with respect to an initial impuls and the following two chapters, 4 and 5, investigate methods how to control the robot in question to reach a desired position and orientation. The limits of possibilities are discussed in details, separating the technical unsolvability of a problem, caused by the physical and geometrical characteristics of the robot, from the cases when a problem is unsolvable within the framework of the used mathematical apparatus, even if a physical solution may be found by an appropriate heuristic reasoning or even by an "try and error" experimentation.

While Chapters 1 to 5 deal rather with the

statics and kinematics, the following chapters are oriented toward the dynamics of the robot motions. The principles of speed computation and control are explained in Chapter 6, Chapters 7 and 8 being consecrated to the dynamical robot control and manipulation. The presented model takes into consideration some phenomena occurring in the case of higher speeds of particular articles motion (inertia and centrifuge forces, gravitation) and the static equilibre is replaced by the dynamic one. Chapter 9 mentions some simple systems of robot's learning, namely those when the human subject (the designer, the user) takes the robot's article, the arm, e.g., and makes himself the desired sequence of moves, once or a few times, the robot being able to remember this sequence of moves and to repeat them. The memory mechanisms are supposed to be just of the hardware nature with no references to more sophisticated methods of artificial intelligence being made. Finally, in Chapter 10 the author surveys some practical task of technical nature which can be solved by robots and manipulators using the theoretical and construction principles explained in this book. The list of references containing 72 items is enclosed.

The book is written in a very concise and self-explanatory way with no preliminary knowledge requested, just a certain level of common mathematical culture at the undergraduate level seems to be necessary. It is supposed to be the first in a series of volumes dealing also with other problems of theoretical robotics, including the cognitive robots of higher generations. This volume can be recommended to technicians, designers and other specialists interested in the theoretical foundations of robot hardware systems.

Ivan Kramosil

D. R. HOFSTADTER
**Gödel, Escher, Bach:
An Eternal Golden Braid**

Basic Books Inc. Publishers, New York 1979.
Stran 777; cena neuvedena.

Jde o pozoruhodnou rozsáhlou esej, která byla v r. 1980 odměněna Pulitzerovou cenou.

Autor (nar. 1945), který je svým povoláním badatelem v oblasti počítačových věd, spojuje smysluplným způsobem tak vzdálené a dosud neasociované oblasti lidského poznávání a tvůrčí činnosti, jakými jsou dílo matematika Gödela, obrazy malíře Eschera a Bachova hudba. Tato velká jména jsou však také symboly rozmanitostí zdrojů, z nichž autor čerpá materiál, který zobecňuje, a jimiž jsou matematika, logika, lingvistika, psychologie, neurofyziologie, biologie, fyzika aj.

Základní idea díla se týká toho, jakým způsobem a do jaké míry mohou různé systémy manipulující se symboly odrážet a zrcadlit v sobě (tj. také vyjádřit, reprezentovat, nebo poznávat) svoji vlastní strukturu a podstatu. Míra podobnosti (posuzovaných z tohoto hlediska) mezi zdánlivě zcela odlišnými soustavami je skutečně obdivuhodná. Zatímco zakladatelé kybernetiky zdůrazňovali společné rysy informačních aspektů v různých systémech, Hofstadter se zabývá hlavně podobnostmi a shodami strukturálními. Způsob podání látky však je převážně esejistický a svou formou i mírou konciznosti se liší od běžných vědeckých způsobů sdělování. Obsahuje např. četné části psané jako dialog, nejrůznější beletristické postupy a pro anglicky mluvící svět zřejmě nezbytné paralely z literárního díla Lewise Carrola.

V úvodní části se autor zabývá strukturou Bachových improvizací na téma zadané pruským králem Bedřichem, podobnostmi mezi Gödelovým teorémem a některými Escherovými obrazy, paradoxy pohybu (Zenon aj.) a limitacemi použití počítačů a arteficiální inteligence. V první kapitole se pak pojednává m.j. o vztahu mezi uvažováním, uvažováním o uvažování, uvažováním o uvažování o uvažování atd., o paradoxu "Achilles a želva" a o nekonečném regresu z hlediska možnosti poznávání. Druhá se zabývá hlavně "významy" z hlediska isomorfie formálních systémů a také vztahem: obraz (význam) — pozadí. Třetí vysvětluje podobnost mezi vztahem obraz—pozadí v umění a teorém—neteorem ve formálních (matematických) systémech. Gödelův teorém neúplnosti parafrázuje hudebním příměrem: pro každý gramofon existuje deska, kterou nemůže zahrát, aniž by

přítom zničil vlastní strukturu. To vše vztahuje ke kontrapunktu v Bachových fugách. Čtvrtá je věnována tomu, kdy nabývají symboly ve formálních systémech významu — smyslu, dále definici pojmů a termínů ve spojitosti s vnímáním a myšlením a také rekurzivním strukturám v Bachově hudbě. V páté je idea rekurzivnosti (ve struktuře systému “hnízdí” jakási jiná struktura atd. a mezi strukturami jsou určité vztahy) demonstrována v geometrii, matematických funkcích, fyzikálních teoriích, počítačových programech, v jazycích, v hudbě apod. Šestá se zabývá lokalizací významů v informačních systémech (při kódování a dekódování). Ta je vysvětlována na příkladu desoxyribonukleových řetězců, nerozlučitelných starých nápisů aj. Sedmá se týká m.j. logického významu spojky “a” a toho, proč nabývají symboly automaticky významu v isomorfních systémech. Osmá je věnována rozdílu mezi formálním (logickým) uvažováním a lidským myšlením a dotýká se také Zen-buddhismu. Devátá analyzuje podobnosti mezi zen-buddhismem a některými současnými matematickými a filozofickými idejemi. Desátá se zabývá částí a celkem (redukcionismem a holismem) a úrovněmi organizace různých systémů, přičemž jsou uváděny příklady z oblasti počítačového programování, hudby aj. Jedenáctá pojednává o vztahu mozku a myšlení. Dvanáctá se týká komunikace mezi mozky, toho, co mají různé mozky společné a v čem se liší. Třináctá uvádí počítačové programy realizující algoritmus prediktabilního konečného a jiné neprediktabilního nebo nekonečného hledání, a to z hlediska jednoduchých a obecných rekurzivních funkcí v teorii čísel a Gödelových samovztažných konstrukcí. Čtrnáctá probírá problematiku důkazů Gödelova teorému. V patnácté jsou vyzvočovány z Gödelova teorému úvahy týkající se myšlení. V šestnácté jde hlavně o samoreprodukcii a o “samo-vztažné” struktury zvláště z hlediska přenosu informace mezi úrovněmi. V sedmnácté se pojednává o mimořádných matematických schopnostech některých jedinců a možnostech počítačové simulace lidského myšlení. Osmnáctá rozvíjí toto téma ve směru Turingových myslících strojů a umělé inteligence a uvádí četné

příklady velmi dokonalých počítačových programů. Devatenáctá se m.j. zabývá vztahem vědomých a nevědomých pochodů při myšlení, reprezentaci poznatků v kontextech, interakci koncepcí a tvůrčím myšlením. Poslední, dvacátá kapitola je věnována různým aspektům hierarchických systémů, zvláště z hlediska jejich vztahů k sobě samým (věda zkoumající vědu; lidské myšlení týkající se mozku, který myslí apod.) a filozofickým problémům týkajícím se interpretace Gödelova teorému a jeho aplikace v různých oblastech, dále svobody vůle, podstaty lidského vědomí, umělé inteligence aj.

Samozejmě ve stručném přehledu nelze ani částečně přiblížit rozmanitost a mnohoznačnost celého díla. Je symbolické, že kniha končí odkazem na svůj začátek. Je naznačeno, že kniha sama je vlastně samovztažnou soustavou podobnou těm, o nichž pojednává.

A v tom spočívá zřejmě celý problém: co vyplývá z podobné samovztažné soustavy pro lidské poznávání a činnost? Jde o více než o vtipné a neobvyklé asociace a vnější podobnosti, tj. o inteligentní kratochvilí a jakési intelektuální dobrodružství? Jde o vysoce umělecky a intelektuálně pojatou literaturu o vědě, nebo o zárodek, jakýsi příslib nového integrujícího vědeckého směru? Nejspíše jde o obojí, i když zatím převažuje to první.

Tomáš Radl

A. J. KFOURY, R. N. MOLL, M. A. ARBIB

A Programming Approach to Computability

The AKM Series in Theoretical Computer Science — Texts and Monographs in Computer Science

Springer-Verlag, New York — Heidelberg — Berlin 1982. viii + 251 pages; 36 figs.

The theory of computability, no matter whether based on the notion of recursive function or some other equivalent formal apparatus, has been always considered as an integral part of mathematics offering a theoretical background for effective computational processes realized by computers or other technical devices. However, in our

days the direction of the mutual influence between theoretical and practical computer science is far from being so strict. In other words said, the practice of computer programming influences very much the way of explanation presented by computing theory and computer science papers, monographies and textbooks. The reviewed work can serve as an excellent illustration of these tendencies.

The influence of programming practice and of the author's many years teaching experience seems to be the most remarkable in two aspects. First, the language used when constructing the necessary formal structures is that of the well-known programming language PASCAL (with some detailed modifications). Instead of recursive functions or Turing machines the elements of the present explanation are PASCAL-like programs generated inductively from three elementary assignment statements ($X \rightarrow 0$, $X \rightarrow X + 1$, $X \rightarrow X - 1$) and an elementary *while*-statement. All the usual notions of computing theory are proved to be definable within this framework as programs, properties of programs and relations between programs.

The second aspect of domination of a practical point of view consists in the author's references to an experienced programmer's intuition and common sense as to a quite legal way of argumentation. Methodologically, such an approach possesses the same status as if the authors took profit from the well-known Church thesis (which is often used in monographies and textbooks on computing theory, e.g. by Rogers) and can be accepted or refuted on the ground of the same argumentation. This "programming-like Church thesis" enables to simplify significantly the presented reasoning as it enables to leave aside the greatest part of difficulties arising from the use of a rigorous mathematical apparatus.

Let us briefly review the contents of the reviewed textbook. The first chapter informally presents some problems and methods of computability theory (diagonalization, halting problem) and invites the reader to their more detailed investigation. Chapter 2 deals with the syntax and semantics of PASCAL-like *while*-programs. The following chapter

proves the possibility to enumerate the computable functions and to simulate them by appropriate universal functions, Chapter 4 being devoted to some principal consequences of these fact including the *s-m-n* theorem and the existence of undecidable problems. The following offers some possibilities how to prove the syntactical and semantical correctness of recursive programs. Recursive and recursively enumerable sets are investigated in Chapters 6 and 7 together with their basic properties; Chapter 7 is closed by a "program-like" reformulation of the famous Gödel Incompleteness Theorem. The following chapter develops some other computable properties of sets, the last Chapter 9 mentions briefly some alternative approaches to computability and compares them with that one adopted by the authors. Last but not least, a great number of useful and instructive exercises following each section and left to the reader present a very positive feature of the reviewed textbook and should not be omitted neither in a very short review like this. The examples and exercises are chosen very appropriately from the pedagogical and methodological point of view, are always ordered from the very simple ones to those which are more sophisticated and time consuming, and ought to be considered as an inseparable part of the textbook.

Perhaps the reviewer may be allowed to present also some personal and, of course, subjective opinions concerning the methodology of the reviewed work. Sometimes the authors emphasize a rather defetistic thesis claiming human subordination to computers in such a way that the adoption of computer-fitted formal systems and ways of reasoning is presented as a life necessity. The reviewer believes this choice to be a pure matter of our freedom, in no way implying an a priori abandoning of another point of view and description. The principal dominance of man over machine consists in the fact that man can always freely choose between an internal viewpoint from inside the formal system and a jumping from it, accepting an external position (to see the system from outside, from a distance). Only the first of

these two possibilities is available to a machine or computer and this fact of basic importance is not sufficiently underlined by the authors.

The textbook can be sincerely recommended to experienced programmers or computer specialists who want to take profit of their knowledge and intuition when penetrating deeper into the theoretical foundations of computing theory. From the technical viewpoint the book conserves the high level of all Springer-Verlag scientific editions.

Ivan Kramosil

PETER M. KOGGE

The Architecture of Pipelined Computers

Hemisphere Publishing Corporation, Washington—New York—London 1981.

XIV + 334 pages; \$28.00.

The present demands on the high-speed computers, such as those used, for example, in meteorological picture/scene or signal data processing, could hardly be met only by improving the currently used LSI and VLSI technologies using practically the same, classical architecture of execution, control and I/O subsystems. Perhaps since the very beginning of the electronic computer era teams of engineers and scientists have been carrying out studies and experiments to increase the overall system throughput of the high performance computers by replicating and/or cascading the elementary functional blocks. Such an approach — even when considered from the most general point of view — bears, however, a great deal of new problems, among which cost-performance ratio optimization and operation security are of the utmost importance to the potential users. Such systems are naturally highly complicated and therefore equally expensive and not very common, and thus a paradox situation occurs — to use and design them a good background in related literature is required while, on the other hand, there exist only highly specialized papers and reports not equally accessible and difficult to use due to varying terminology. And when we further consider the development of the

custom LSI technologies that directly support the progress in this area, the lack of reviewing and unifying bibliography is more than apparent.

Dr. Kogge's monography *The Architecture of Pipelined Computers* is one of the first attempts to fill in the gap in the high-speed computer design literature. As the title already indicates the author has limited the scope only to one of the above mentioned approaches (i.e. pipelining or cascading of elementary execution units) even though he is far from discouraging the reader from the other concept (i.e. parallelism) — on contrary, he considers both ideas being complementary and having equal chances of application and even co-existence in the modern designs.

The book may be divided into two main parts, the first three chapters being dedicated to the necessary introduction into the most important problems and to the development of most commonly used analytical and graphical tools, while the remaining four chapters focus on the global aspects of the pipelined computer architecture, programming and finally, on the possible future trends.

The first, introductory chapter outlines briefly the scope of topics to be covered in text to follow, delimiting also the meaning of the pipelining in contrast to parallelism and overlapping and presents the basic classification of systems to be studied later. Chapter 2 is dedicated to the construction of the lower level functional and auxiliary subsystems, such as staging latches and various hierarchical levels of the memory (main and local memories and register files). Based on these elementary building blocks more complex computational structures are described, namely cascade adder and multiplier, multifunction pipeline and a certain attention is paid also to the basic ideas and relations that may be used as a guide for efficiency and cost-performance ratio optimization.

The first part of the publication culminates in Chapter 3 with a very detailed analysis of timing, control and performance evaluation problems. Further analytical tools are introduced, especially the reservation tables and the so-called modified state diagrams and their

application is supported by a set of useful theorems and illustrated by a detailed sample system design.

The second part of the book, with exception of the last chapter, is dedicated to the discussion of the pipelined architecture application in SIMD and SISD machines and to the problems of the sub-instruction level programming. Chapter 4 deals with the vector processors (i.e. SIMD type) concentrating mainly on the attributes typical of machine-level instruction execution (such as operand specification, operation sequencing etc.) as well as on effects of various addressing strategies on the total system performance. These general remarks and relations are used in the following sections that describe several real vector computers (IBM 2938/3838 AP's, TI-ASC, CDC Star-100 and CRAY-1). Chapter 5 then investigates problems related to the vector instruction implementation with a special attention to segmentation, recursion removal and inner loop design and introduces several models suitable for evaluation of effective performance of vector instructions, especially with respect to the boundary conditions (i.e. effects of start-up and flush times) and to the influence of scalar instructions of the overall throughput. The problems are illustrated by a series of examples and a comparison with the parallel vector systems is also included.

Chapter 6 investigates problems associated with the pipelining applied to the more conventional SISD machines and stresses out the importance of maintaining both the execution causality, when several instructions may be carried out simultaneously, and the data integrity when possible external or internal interrupts are to be serviced. Also a fairly detailed discussion is dedicated to the analysis of various types of cache memories.

The monography is by no means an easy-to-read cookbook as it covers a vast range of pipelined computer technology on a rather limited space. Nevertheless, it is a very valuable publication due to a well balanced combination of theory and practical expertise reflecting author's experience as both a university professor (State University of New York, University of Massachusetts) and a research

engineer (IBM Federal Systems Division). Many illustrative examples and problems for the reader throughout the whole text as well as the rich bibliographical references only contribute to the high quality and usefulness of the book.

Boris Dědina

V. SCHMIDT, D. KOLLBACH,
H. - G. METZLER, H. PANGRITZ,
B. UHLMANN

Digitalschaltungen mit Mikroprozessoren

(druhé, přepracované vydání)

Leitfäden der angewandten Informatik.

B. G. Teubner, Stuttgart 1981.

Stran 205; 97 obr., 12 tab.; cena DM 23,80.

Publikace skupiny pěti autorů z různých výzkumných a vývojových pracovišť má charakter příručky pro úvodní seznámení s problematikou používání mikroprocesorové techniky, většinou z produkce firmy Motorola. V osmi kapitolách seznamuje čtenáře s hlavními otázkami konstrukce mikropočítačových systémů, jejich programování a ožívání.

V prvních dvou kapitolách je zaměřena pozornost na návrh a možnosti užití různých typů paměťových struktur a na vyšší typy obvodů pro konstrukci obecných výpočetních systémů (PLA, ALU).

Cílem kapitoly třetí je charakterizovat základní funkční bloky číselných počítačů se zaměřením na další text, v němž se popisují základní principy konstrukce mikroprogramových procesorů s pomocí stavebnice 2900 (kap. 4), a struktura, instrukční soubor a základní aplikační pravidla monolitického mikroprocesoru MC 6800, včetně standardních pomocných obvodů (kap. 5). Informativnímu rázu knížky odpovídá i pasáž věnovaná programování mikropočítačů, kde se autoři zmiňují o jazyku symbolických adres, BASICu a MPL (obdoba PL/M firmy Intel) a uvádějí některé zásady pro vytváření uživatelských programů. V sedmé kapitole může čtenář nalézt několik cenných informací o možných odlaďovacích a oživovacích postupech, včetně základní charakteristiky vhodného přístroje-

vého vybavení (logický analyzátor, obvodový emulátor). Publikaci pak uzavírá kapitola popisující prakticky úplný návrh speciálního automatu pro snímání a vyhodnocování čárkového kódu (bar graph code) s použitím mikroprocesoru MC 6800.

Lze konstatovat, že první vydání recenzované knihy mělo ještě určitý význam pro poměrně širokou obec zájemců o mikroprocesorovou techniku. Avšak za období do jejího 2. vydání vývoj v této oblasti pokročil do té míry, že obdobné tituly mohou být vhodné pouze pro skutečné začátečníky se základním odborným vzděláním. A díky výhradní orientaci na produkci firmy Motorola a s respektováním zaměření tuzemské součástkové základny nelze tuto publikaci našim čtenářům doporučit.

Boris Dědina

CHARLES E. HUGHES, CHARLES P. PFLEGER, LAWRENCE L. ROSE

Advanced Programming Techniques

A Second Course in Programming Using Fortran

John Wiley and Sons, New York 1979.
Ruský překlad: Metody programování — Kurs na osnove Fortrana, Mir, Moskva 1981.

Zkušení američtí autoři popisují programovací techniky prostřednictvím jazyka Fortran IV. Je zde rozebírána struktura jazyka a metody zvýšení efektivity programů, vstup a výstup na magnetických páskách a discích, podprogramy a jejich použití, nenumerní operace a datové struktury. Krátce se zmiňují i o programovacích jazycích Fortran 77 a WATFIV i o řízení prací v operačních systémech.

Knihou představuje materiál, který umožní začínajícím programátorům se dostat na profesionálnější úroveň a profesionálům umožní rozšířit si vědomosti. Proto jsou nejprve vloženy zásady a principy, potom konkrétní ilustrativní příklady. Praktické zaměření publikace je umocněno i orientací autorů na počítače typu IBM 360–370 a DEC-10.

Celkem se publikace skládá z 9 kapitol a 2 příloh. Nullá kapitola obsahuje solidní popis Fortranu a s ním souvisejících dialektů, včetně některých specifických a podrobnějších otázek (např. ukládací funkce poli) a konkrétních údajů o konkrétních strojích — viz výše. Kapitola 1. je věnována stylu programování, struktuře programu a jeho modifikaci, ale i problematice ladění a jak se chránit před chybami při programování, efektivitě programu a algoritmu, otázkám dokumentace. Kapitola 2. shrnuje problematiku procedur, jejich parametrů a volání, použití procedur v roli parametrů, COMMON, jeho použití, vztah k EQUIVALENCE atd. V rámci příkladů je popsána HASH-funkce pro hledání v tabulce, generátor náhodných čísel a integrace metodou Monte-Carlo. Programování nenumerních úloh je věnována kapitola 3. Obsahuje: znaky a jejich kódování, znakový V/V, v příkladu řešení bezformátový vstup a výstup celých čísel a služební programy. Kapitola 4. obsahuje údaje o vstupu a výstupu prováděném na magnetických médiích — páskách a discích. Je zde probírána fyzická a logická organizace, instrukce Fortranu pro práci s magnetickými médii, speciálně pak v systémech IBM a DEC. Nechybí ani popis zařízení pro přímý přístup a způsob práce s ním, konkrétně pak v OS-IBM. Závěrečný příklad je věnován hledání záznamů o knihách dle klíčových slov. 5. kapitola se zabývá datovými strukturami a jejich různými druhy: seznamy, zásobníky, frontami, ukazateli, vazbami v seznamech, stromy atd., jakož i příslušnými algoritmy pro jejich používání. Následující kapitoly obsahují vlastně jakési „hardwarové minimum“ pro programátory: 6. zobrazení údajů v počítačích a 7. strukturu samočinného počítače a jeho princip práce. Zde je i ilustrováno použití jednoadresového assembleru. Poslední, 8., kapitola je určena spíše pokročilejším programátorům (jako i některé dříve uvedené pasáže): efektivní programování a využití prostředků operačního systému. Nechybí ani rejstřík a obsah; bohužel není zde uvedena žádná doporučená ani použitá literatura. V příloze I. jsou popsány vnitřní funkce jazyka Fortran, v příloze II. rozdíly mezi Fortranem IV a Fortranem 77.

Kniha obsahuje na konci každé kapitoly řadu vtipných cvičení, která nechtějí opisovat příkazy dle definic, jak se s tím v mnohých učebnicích setkáme, ale vyžadují přemýšlení, vtip a aktivní přístup. V řešených příkladech — konkrétních ukázkách v textu jsou ukázány různé styly programování a dokumentace. Není zde gloriifikován jeden jako nejlepší, ale všechny jsou ukázány se svými přednostmi i nedostatky tak, aby se čtenář seznámil se všemi možnostmi a mohl si vybrat pro sebe nejvhodnější, vlastní způsob. Celkově lze říci, že publikace je psána promyšleným a názorným způsobem a předpokládá více znalostí, než například naše neúspěšnější kniha o programování, Fortran Jiřího Vogela. To umožňuje i lépe uspořádat a přehledněji vykládat celý materiál.

Kniha může být užitečná jak studentům — specialistům informatikům, tak i uživatelům samočinných počítačů a programátorům v praxi. Nevyžaduje předem žádné speciální znalosti kromě základních principů programování v libovolném procedurově orientovaném jazyku, tj. v rozsahu úvodního kursu algoritmizace a programování.

Vladimír Smejkal

JÜRGEN LÄUTER **Programmiersprache DIST — Dateneingabe, Datenstrukturierung**

Akademie-Verlag, Berlin 1981.
 Stran VIII + 168; 26 obr.; cena M 24,—.

Opět nový programovací jazyk? Další přírůstek do početné rodiny univerzálních programovacích jazyků? Ne, tentokrát se nejedná o jazyk univerzální, ale specializovaný: DIST je jazyk pro Data, jejich Interpretaci, Strukturování a Transformaci. Ještě konkrétněji: jde o sekvenční soubory primárních dat, pořizovaných z dotazníkových akcí, ze záznamových zařízení apod. Jedná se většinou o data z medicíny, biologie, zemědělství, technologických procesů, ekonomie, sociologie atd., určená ke statistickému zpracování.

Potřeba specializovaného programovacího jazyka pro tuto oblast vznikla následujícím

způsobem. Metody statistického zpracování jsou již poměrně dobře vyvinuty a mají náležitou obecnost (výpočty průměrů, rozptylů, korelace, četnosti, diskriminační analýza apod.), takže soubory statistických programů mohou být s úspěchem využity v různých oborech. Vstupní data pro tyto programy však vznikají různými způsoby a mají různé tvary. Téměř v každém případě je zapotřebí data předzpracovat, než mohou být použita ve statistickém programu. Je jasné, že toto předzpracování je zapotřebí řešit odděleně od vlastních výpočtových programů. Tato oblast je natolik rozmanitá, že ji nelze pokrýt souborem hotových programů. Nejlepším řešením je zde jazyk, který pružným způsobem umožní zvládnout řadu různých situací.

Problémy, které se vyskytují při předzpracování dat, jsou:

- vstup dat z různých vstupních zařízení (děrná páska, děrné štítky, magnetická páska),
- překódování dat z různých kódů (nejčastěji u děrné pásky),
- různé formáty dat (binární či dekadická data, pevný či volný formát dat uvnitř věty, pevná či proměnná délka věty),
- kontrola chyb (kontrolní součty, kontrola maxima a minima, kontrola délky věty),
- odstraňování nebo oprava vadných dat,
- přepočet měřítka, korekce nelinearit měřících přístrojů, normalizace, různé transformace,
- přestrukturování a přetřídění dat (skupiny dat, odpovídající jednotlivým měřením, je zapotřebí zmínit tak, aby se veličiny, určené pro další společné zpracování, dostaly k sobě).

Překladač DIST má dva vstupy: datový materiál a popis formátů. V popise formátů je popsán tvar datového materiálu a požadavky na předzpracování. Výstupem překladače je předzpracovaný soubor dat, opatřený pomocnými údaji o struktuře a jménech. Soubor má unifikovaný tvar, vhodný pro další zpracování. To se děje buď standardním statistickým vybavením nebo uživatelskými programy (PL/I, Fortran).

Jazyk DIST pro popis formátů je řádně definován a srozumitelně vysvětlen na řadě

příkladů. Způsob popisu formátů vychází ze vstup/výstupních formátů jazyka PL/1 nebo Fortran, je však rozveden do velké řady dalších prostředků pro vystižení struktury a hierarchie a pro transformace. Hlavními znaky jazyka jsou kompaktnost a stručnost: většina problémů předpracování může být zvládnuta bez potíží použitím běžných programovacích jazyků, ale místo několikastránkových programů máme v DISTU pár řádků textu (velmi hutného). Pro pracoviště, kde se neustále řeší úkoly předpracování dat, to může přinést zlepšení práce. Rozhodnutí o použití či nepoužití DISTU je však zapotřebí dělat podle konkrétní situace.

Jazyk je koncipován jako strojově nezávislý. Kromě vlastního jazyka je popsána i implementace na počítačích EC s operačním systémem OS, vyvinutá ve VEB ROBOTRON ve spolupráci s ústavu DAW. Systém se používá ve výpočtových střediscích v NDR.

Jan Ježek

DAVID GRIESE

Kompilátory číslicových počítačů

Překlad anglického originálu *Compilers for Digital Computers*, John Wiley and Sons, New York 1971.

ALFA — Vydavatelstvo technickej a ekonomickej literatury, Bratislava, SNTL — nakladatelství technické literatury, Praha 1981. Stran 512; 155 obr.; cena Kčs 39,— (váz.), Kčs 36,— (brož.).

Programovací jazyky jsou již dnes neodmyslitelnou součástí počítačů a výpočtových systémů. Kromě univerzálních jazyků ze základního vybavení se používají specializované jazyky pro různé obory. Téměř každý větší programový systém obsahuje nějaký řídící, sestavovací či dotazovací jazyk, který vychází vstříc potřebám uživatele. Tomuto trendu odpovídá potřeba, aby tvůrci programů získali znalosti o struktuře a tvorbě překladačů a interpretačních programů.

Učebnic konstrukce překladačů, přístupných širšímu okruhu programátorů, je velmi málo. Publikace v tomto oboru bud řeší teoretické problémy jazyků a gramatik nebo

popisují určité konkrétní překladače. Jednou z prvních učebnic je právě kniha profesora Davida Griese. Jejím cílem je podat ucelený přehled technik používaných při psaní překladačů. Může sloužit jako podklad pro studium (částečně vznikla z přednášek na univerzitách Cornell a Stanford) i jako praktická příručka profesionálního programátora. Autorovy zkušenosti, pedagogické mistrovství i odborný přehled způsobily, že kniha byla přeložena do řady světových jazyků a na mnohých vysokých školách (v USA, SSSR i jinde) slouží jako základní učebnice. Ani dnes, po více než deseti letech od prvního vydání, neztratila na své hodnotě.

Každý překladač je poměrně rozsáhlý program, který musí řešit řadu úkolů: lexikální analýzu textu, syntaktickou analýzu, sémantické zpracování, údržbu a obsluhu tabulek, optimalizaci programu, generování kódu, organizaci paměti ve fázi výpočtu apod. Těmto problémům jsou věnovány jednotlivé kapitoly knihy.

Přípravná část poskytuje čtenáři úvod do teorie formálních jazyků a gramatik. Základní definice, stromy odvození, problém nejednoznačnosti gramatik, různé třídy gramatik. V kapitole o lexikální analýze jsou uvedeny základní poznatky o regulárních gramatikách a konečných automatech. Čtenář se naučí psát analyzátor pro různé jazyky, je zde popsán i univerzální konstruktor, který pro zadanou gramatiku sestrojí analyzátor.

Kapitoly o syntaktické analýze se týkají bezkontextových jazyků. Je vysvětlen obecný algoritmus analýzy metodou shora dolů a uvedeny možnosti zefektivnění pro různé případy. Metody zdola nahoru jsou probrány pro precedenční a operátorově precedenční gramatiky. Problémy z této oblasti nelze řešit univerzálním konstruktorem, lze však proces tvorby analyzátoru automatizovat alespoň částečně, pomocí jazyka pro psaní analyzátorů.

Sémantické zpracování, navazující na syntaktickou analýzu, spočívá v překladu programu do vhodného vnitřního tvaru. To může být polský zápis, trojice, čtveřice, stromy. Popisují se různé způsoby reprezentace a různé algoritmy. Dále jsou popsány různé způsoby obsluhy tabulek (tabulky seříděné,

zřetězené, s transformací klíče, s blokovou strukturou). Optimalizace programů: strojově nezávislá, závislá, lokální, globální. Generování kódu, způsoby adresování operandů, tvary cílového jazyka. Organizace paměti ve fázi výpočtu: statická, dynamická paměť, bloky, pole, struktury, řetězce. Zvláštní kapitoly jsou věnovány tvorbě interpretačních programů a makrogenerátorů. A nakonec praktické rady všeho druhu pro tvůrce překladačů.

Všechny algoritmy jsou srozumitelně vysvětleny a jejich hlavní části popsány v (modifikovaném) Algolu s komentáři. Popis je veden tak, aby čtenář dobře zachytil základní myšlenky a naučil se sám psát podobné programy. Z různých metod jsou vybírány především ty, které mají obecnou platnost. Zvláště se vyzdvihují různé pomocné prostředky a programy, které usnadňují tvorbu překladačů („překladače překladačů“).

Ve slovenském překladu je původní seznam literatury (174 odkazů, do r. 1971) obohacen o nejdůležitější novější publikace (15 odkazů). Překlad je také výrazem snahy přispět k vytvoření jednotné slovenské terminologie v oblasti programovacích jazyků. Vydání této hodnotné knihy je třeba uvítat i proto, že odbourává starý názor, podle kterého je tvorba překladačů záležitostí jen několika vyvolenců. V budoucnu by měly překladače patřit k běžné práci programátora podobně jako zpracování dat.

Jan Ježek

RAINER HAHN
**Höhere Programmiersprachen
im Vergleich —
Eine Einführung**

Studententexte: Informatik.
Akademische Verlagsgesellschaft, Wiesbaden 1981.

Stran 281; 42 obr., 17 tab.; cena DM 29,80.

Publikace je vysokoškolským učebním textem informatiky. Vychází z poznatku, že obvyklá výuka studentů v jednom určitém programovacím jazyce neodpovídá tomu, co budou ve svém budoucím povolání potřebovat. Programovacích jazyků je mnoho a student se

musí na tuto nepříjemnou situaci připravit. Není možné učit jeden jazyk za druhým. Je zapotřebí vyzdvihnout základní prvky pro organizaci a řízení výpočtů, ukázat různé způsoby jejich jazykové realizace a porovnat je. Tím dostane student celkový přehled, který mu umožní přecházet podle potřeby na různé jazyky.

Pro výklad byly vybrány nejrozšířenější programovací jazyky, s kterými přijde čtenář nejspíše do styku: Ada, Algol 60, Algol 68, Basic, Cobol, Elan, Fortran 66, Fortran 77, Pascal, PL/1, PLZ/SYS, Simula 67. Některé z nich poněkud vybočují z řady: Ada je zatím ve stadiu prvnick realizací, ale očekává se rychlé rozšíření. Elan se hodně používá v NSR. PLZ/SYS je zde jako zástupce jazyků pro mikropočítače, kde se také očekává prudký rozvoj. Žádný z těchto jazyků není pochopitelně v knize systematicky vyloučen, vybrané jazyky slouží pouze pro konkrétní ilustraci obecných principů a konstrukcí.

Po úvodních kapitolách, kde jsou stručně vysvětleny základní pojmy a uveden historický vývoj jazyků, postupuje kniha podle jednotlivých jazykových a organizačních prvků. Jsou to: lexikální struktura jazyka, datové objekty, typy dat, deklarace, příkazy, podprogramy, organizace paměti, rozsahy existence a přístupnosti objektů, programové moduly, tvorba nových typů dat, přístup k datovým objektům pomocí referencí, ošetření mimořádných stavů. U všech prvků se probírají problémy, které měli tvůrci jazyků před sebou a jak se jim podařilo či nepodařilo je řešit. Každý prvek se hodnotí především podle toho, jak přispívá ke tvorbě jasných, dobře čitelných a verifikovatelných programů. Pro další studium je uvedena literatura (243 odkazů).

O jak aktuální a potřebnou tematiku se jedná, svědčí to, že v témže roce vyšla podobná publikace — H. J. Schneider: Problemorientierte Programmiersprachen, recenze viz Kybernetika 18 (1982), 6. Je těžké porovnat obě knihy a doporučit přednosti některé z nich. Obě jsou vedeny podobnou myšlenkou a podobně zpracovány, takže mohou být používány ekvivalentně. Hahnova kniha je snad trochu podrobnější a snaha po utřídění jde do větší hloubky.

Jan Ježek

NIKLAUS WIRTH

Systematické programovanie

Překlad německého originálu *Systematisches Programmieren*, B. G. Teubner, Stuttgart 1975.

ALFA — Vydavatelstvo technickej a ekonomickej literatúry, Bratislava, SNTL — nakladatelství technické literatury, Praha 1981.

Stran 204; 45 obr.; cena Kčs 19,— (váz.), Kčs 14,50 (brož.).

Profesor Niklaus Wirth je známý a uznávaný odborník v oblasti teorie programování a informatiky. Jeho jméno je spjato se známými principy strukturálního programování a s jazykem Pascal. Kniha *Systematické programovanie* je překlad druhého vydání jeho učebnice, určené pro základní kurs programování, pro studenty, kteří se s tímto oborem setkávají po prvé.

Kniha se dost liší od obvyklých učebnic programování. Obvyklý postup je seznámit žáka s určitým programovacím jazykem tak, aby byl schopný zapsat známé algoritmy ze svého oboru a uvést je do chodu na počítači. Profesoru Wirthovi zde však jde o něco jiného: naučit žáka systematicky vytvářet a formulovat algoritmy, t.j. zvládnout ten doposud poměrně málo známý proces, probíhající v hlavě programátora, kdy na začátku je přibližná formulace problému v hrubých rysech a na konci je přesný zápis algoritmu pro počítač.

Další odlišností knihy je to, že autor učí svého žáka již od začátku základní myšlenky a techniky verifikace programů. Tento nový přístup spočívá v tom, že správnost algoritmu ověřujeme rozбором toho, jak se mění vztahy mezi proměnnými od příkazu k příkazu, které vztahy zůstávají invariantní apod. Programátor se tak od začátku učí pracovat podobně jako matematik, který dokazuje své věty logickým odvozováním, a ne jako počítáč, který věty "dokazuje" dosazováním několika číselných hodnot.

Kniha je psána srozumitelně a přehledně, vyzdvihuje základní abstraktní pojmy v idealizovaném tvaru a upouští od detailů. Čtenář se vše učí na podrobně vypracovaných příkladech a pak může přejít k samostatné práci na dalších cvičeních. Příklady jsou voleny tak,

aby nebyla preferována žádná aplikační oblast. Kromě ilustrace obecných principů poskytnou příklady i přehled o základních algoritmech: hledání, třídění, úprava textů, rekurentní postupy, převody čísel (desítková, dvojková soustava, pevná, pohyblivá čárka), elementy syntaktické analýzy, základní metody numerické matematiky. Ruku v ruce se strukturou algoritmů (složené, podmíněné příkazy, cykly, podprogramy) se vykládá i struktura dat, t.j. způsoby, jimiž se ze základních typů dat (real, integer, boolean, character) vytvářejí složené typy (pole, soubor). Konkrétní programovací jazyk ustupuje do pozadí, pro účely výkladu autor používá Pascal, který byl vytvořen právě tak, aby odpovídal myšlenkám strukturování a verifikace. Ke zvládnutí knihy stačí středoškolská matematika (s výjimkou několika příkladů z diferenciálního a integrálního počtu), je však zapotřebí odvaha samostatně myslet.

Knihu lze uvítat za to, že usiluje o pozdvižení programátorské práce na vyšší úroveň. Lze ji doporučit studentům, pracovníkům výpočetových středisek i výzkumných ústavů, a to především těm, kteří považují systematické vytváření algoritmů a programů za součást svého matematického vzdělání. Méně je určena těm, kteří si chtějí pouze čas od času něco naprogramovat a vypočítat. Rovněž tak ne těm, kteří při programování přijdou do styku především pouze s kódovací a provozní stránkou této práce.

Jan Ježek

D. J. BELL, P. A. COOK, N. MUNRO
Design of Modern Control Systems

IEE Control Engineering Series 20.
Peter Peregrinus Ltd. (on behalf of the Institution of Electrical Engineers), Stevenage, UK — New York 1982.
xii + 332 pages; £ 22.50.

This volume is a collection of lectures given at the vacation school organized by the Science and Engineering Research Council which was held at the Control Systems Centre in the University of Manchester Institute

of Science and Technology on March—April 1982.

The book is aimed to inform practising system engineers on some recent design methods for multivariable systems.

The first chapter, *State-space theory* (P. A. Cook), gives a sketch on state-space concepts.

The second chapter, *Complex variable methods in feedback systems analysis and design* (A. G. J. MacFarlane) is devoted to generalization of Nyquist and Root-locus diagrams to multivariable systems.

In *Robustness in multivariable control system design* (I. Postlethwait), some techniques are described assessing the relative stability of a control system.

In the chapter *A design study using the characteristic locus method* (J. M. Edmunds), a 3-input control system is developed for the regulation of the product concentration and temperature of 2-bed exothermic catalytic reactor.

In the chapter *The inverse Nyquist array design method* (N. Munro), the concept of diagonal dominance and Rosenbrock's Inverse Nyquist Array Method are described.

The practical case of the control of a nuclear boiler is considered in *Analysis and design of a nuclear boiler control scheme* (F. M. Hughes).

An overview of dynamic optimization is presented in *Optimal control* (D. J. Bell).

In the chapter *Control system design via mathematical programming* (D. Q. Mayne), it is shown that many control problems can be formulated and solved as mathematical programming problems with infinite-dimensional constraints.

Three algorithms to achieve diagonal dominance are described in *Optimisation in multivariable design* (G. F. Bryant).

Some methods of pole assignment using state feedback and output feedback are described in *Pole assignment* (N. Munro).

Some methods for analysing the behaviour of nonlinear systems are discussed in *Non-linear systems* (P. A. Cook).

In the chapter *Some DDC system design procedures* (J. B. Knowles), a microprocessor based digital control is treated.

In the nice chapter *Robust controller design*

(E. J. Davison), the important questions of robustness in the servomechanism problem are reviewed.

In *Control of distributed parameter systems* (A. J. Pritchard), the concepts of controllability and optimal control and estimation are investigated for systems described by partial differential equations.

In the last chapter *Decentralised control* (M. G. Singh), a brief overview of some of the recent results in decentralised control is given.

Each chapter includes many references and examples solutions of which are given at the end of the book.

In summary, a number of contributions describing different fronts of the control theory has been brought together under one cover. Although all the chapters are so brief that they naturally cannot provide any deep insight into the subjects, the book will well serve the practising system engineer in industry in obtaining first information on many recent design methods for multivariable systems.

Michael Šebek

PETER NASH (Ed.)

System Modelling and Optimization

IEE Control Engineering Series 16.

Peter Peregrinus Ltd. (on behalf of the Institution of Electrical Engineers), Stevenage, UK — New York 1981.

Stran xii + 201; 36 obr.; cena £ 17.00.

Recenzovaný sborník obsahuje materiály přednesené na letní škole konané na univerzitě v Cambridgi v r. 1980. Cílem organizátorů této akce bylo seznámit především graduované studenty pracující v oboru teorie řízení s nejrůznějšími metodami optimalizace a ukázat jim, jak optimalizační metody rozpracované nezávisle v oblasti operačního výzkumu a v oblasti teorie řízení mohou být užitečné při řešení konkrétních úloh praxe.

Sborník lze proto považovat za úvodní přehledovou publikaci o teorii optimalizace a optimalizačních metodách, který přináší i řadu příspěvků ilustrujících praktické aplikace optimalizačních metod pro nejrůznější systémy.

V aplikačních příspěvcích je důraz kladen jak na sestavení modelu, tak i na problémy vznikající řešením specifických optimalizačních úloh.

Prvých 5 kapitol seznamuje čtenáře se základními optimalizačními metodami. Výklad základů matematického programování v první kapitole je založen na nutných a postačujících podmínkách optimality a využití teorie Lagrangeových multiplikátorů. Je zde uveden i přehled řady algoritmů různých typů, zejména metody pokotovými funkcí a metody zobecněných Lagrangeových multiplikátorů pro úlohy s omezeními. Druhá kapitola je věnována dynamické optimalizaci a přináší přehled hlavních výsledků klasických metod variačního počtu i moderní teorie optimálního řízení (jsou zde uvedeny např. Eulerovy rovnice, princip maxima i dynamické programování). Krátká třetí kapitola pojednává o dekompozičních metodách rozpracovaných v teorii optimálního řízení. Klasickému lineárnímu programování jsou věnovány kapitoly čtvrtá a pátá. Výklad zahrnuje všechny základní problémy lineárního programování (včetně citlivostní analýzy) a dekompozice podle Dantzigova-Wolfeova algoritmu.

Kapitoly 6, 8 a 9 jsou věnovány praktickým aplikacím teorii vyložených v úvodních kapitolách (optimální řízení části petrochemického závodu, optimální regulace hladiny vodních nádrží v systému plavebních kanálů ve Velké Británii a využití lineárního programování v marketingových modelech odbytu uhlí). Značná pozornost je zde věnována formulaci problému i tomu, jak je tvorba modelu ovlivňována zvolenou technikou jeho řešení i účelem, ke kterému je model navržen. Sedmá více teoreticky zaměřená kapitola, pojednává o využití metod teorie optimálního řízení na makroekonomické modely.

V předposlední desáté kapitole se výhradně diskutují otázky tvorby modelu v úloze o roli telekomunikačních systémů v rozvojových zemích.

Podstatným výsledkem optimalizačních modelů jsou rozhodnutí a rozumná rozhodnutí vytváří důvěru v navržené modely. V poslední jedenácté kapitole se proto rozebírají formální a neformální přístupy pro vyhodnocování

navržených modelů a kritérií, podle kterých lze posuzovat vhodnost modelů.

Sborník je pozoruhodným pokusem v celosvětové tendenci sjednocovat myšlení a způsob práce odborníků odchovaných školami zaměřenými na teorii řízení nebo operační analýzy a ukázat jim výhody a slabiny různých optimalizačních metod při řešení konkrétních úloh praxe. Je zhotoven standardní fotooffsetovou technikou a doplněn i věcným rejstříkem. I když tuto publikaci není možno považovat za učebnici optimalizačních metod, vážný zájemce získá po jejím prostudování dobrý přehled o optimalizačních metodách a jejich nejrůznějších aplikacích.

Karel Sladký

D. G. LAINIOTIS, N. S. TZANNES (Eds.)

Advances in Control

Volume II of selection of papers from INFO II, The Second International Conference on Information Sciences and Systems, University of Patras, Greece, July 9—14, 1979.

D. Reidel Publishing Company, Dordrecht—Boston—London 1980.

Stran xvi + 592; cena neuvedena.

Recenzovaná publikace je druhým dílem třísvazkového sborníku vybraných referátů přednesených na 2. mezinárodní konferenci o informačních vědách a systémech konané v červenci 1979 na univerzitě v Patras v Řecku pod záštitou řady řeckých i mezinárodních organizací a institucí. Druhý díl obsahuje příspěvky věnované teorii systémů a teorii řízení a je rozdělen do 4 kapitol. Většina referátů je napsána autory z Francie, Španělska a USA; méně jsou už zastoupeny Itálie, Řecko a Velká Británie a najdeme zde i příspěvky z Brazílie, Japonska a Austrálie.

Úvodní kapitola s názvem "A Short Report on the Optimal Control of Distributed Systems" přináší pouze jediný stejnojmenný referát J. L. Lionse věnovaný přehledu nejdůležitějších poznatků o optimálním řízení systémů s rozloženými parametry (tj. systémů popsaných parciálními diferenciálními rovnicemi).

Naproti tomu zbývající 3 kapitoly přináší pouze kratší vědecká sdělení. Ve druhé kapitole (Systems — 17 příspěvků) převažují refe-

ráty věnované obecné teorii systémů a matematickým metodám souvisejících s touto problematikou (např. referáty věnované teorii grafů a náhodným procesům), i když řadu referátů (např. sdělení věnovaná podmínkách autonomnosti mnohazměrových systémů nebo problematice snižování řádu soustavy) lze považovat za typické teoretické inženýrské práce. Ve třetí kapitole (Control Theory — 23 příspěvků) převažují referáty věnované spíše inženýrsky pojatým úlohám o řízení systémů a objevují se zde i příspěvky vyložené aplikačního charakteru. Zajímavé je, že řada příspěvků (zejména francouzských autorů) souvisí s identifikací technických systémů, i když problematice identifikace je věnována další kapitola. Závěrečná čtvrtá kapitola (Estimation and Identification — 20 příspěvků) je opět věnována spíše obecným teoretickým problémům odhadování a identifikace, i když jsou v této kapitole obsaženy i příspěvky vyložené aplikačního charakteru (např. aplikace v radiolokaci nebo při návrhu evropské umělé družice).

Sborník přináší široké spektrum prací z oblasti systémů a řízení včetně jejich aplikací a poskytuje tak široký přehled o aktuálních problémech v této oblasti. Je zhotoven obvyklou fotoofsetovou technikou z originálů dodaných autory. Pro snadnější orientaci je doplněn autorským a věcným rejstříkem.

Karel Sladký

Z. ZLATEV, J. WASNIEWSKI,
K. SCHAUMBURG

Y12M

Solution of Large and Sparse Systems of Linear Algebraic Equations

Lecture Notes in Computer Science 121.
Springer-Verlag, Berlin—Heidelberg—New York 1981.

Stran IX + 128; cena DM 18,—.

Obsahem publikace dánských autorů je dokumentace vlastních procedur souboru Y12M v jazyku FORTRAN určeného k řešení velkých a řídkých soustav lineárních algebraických rovnic a vyvinutého v Regional Computing Center Kodaňské university (RECKU).

Publikace je členěna na předmluvu, devět kapitol, devátou jsou reference, a rejstřík.

První kapitola, úvod k souboru Y12M, je výčetem pěti úloh řešitelných souborem Y12M, tedy úloh opakovaného řešení soustav lineárních algebraických rovnic rozlišených strukturou soustav, dále je popisem matematické metody, zde Gaussovy eliminace s volbou strategií pro výběr hlavního prvku, a doporučením k použití uvedených vlastních procedur. Doporučení se týkají volby strategie pro výběr hlavního prvku, robustnosti výpočtu, zejména s ohledem na přetečení, podtečení a singularitu a dále výpočtu s iteračním vyhlazením umožňujícím v případě řídkých matic zprůměrnění a zkrácení výpočtu s volbou velkého činitele stability a relativně velkou hodnotou definiční nulu. Dále je popsán obsah souboru Y12M.

Druhá až šestá kapitola mají jednotnou formální strukturu a jsou dokumentací vlastních procedur, postupně: Y12MA — Gaussova eliminace bez iteračního vyhlazení, Y12MB — uspořádání podle řádků, Y12MC — faktORIZACE na dvě trojúhelníkové matice, Y12MD — řeší soustavu algebraických lineárních rovnic dle Y12MC, Y12MF — Gaussova eliminace s iteračním vyhlazením a volitelnou strategií pro výběr hlavního prvku soustav s řídkou maticí. S výjimkou vlastní procedury Y12MF, použitelné pouze v jednoduché aritmetice, je u ostatních možná volba jednoduché nebo dvojnásobné aritmetiky. Uvedené vlastní procedury jsou doplněny vlastní procedurou Y12MG pro výpočet čísla podmíněnosti a generátorem testovacích matic.

Sedmá kapitola je diagnostikou chyb, osmá je věnována přenosnosti souboru s příklady testů na počítačích řady UNIVAC 1100, IBM 3033, CDC Cyber 173.

Soubor Y12M je vybudován na plném využití základních možností Gaussovy eliminace k řešení velkých a řídkých soustav lineárních algebraických rovnic s vyčerpávajícím členěním efektivního řešení formulovaných úloh. Soubor je k dispozici u autorů za nominální cenu. Použití pouze standardních příkazů v jazyku FORTRAN umožňuje maximální přenosnost souboru. Publikace má informativně dokumentační charakter a lze ji doporučit numerickým matematikům, ana-

lytikům a programátorům, zejména však specialistům v oboru řídkých matic.

Lubomír Bakule

TSUYOSHI MATSUO
**Realization Theory
of Continuous-Time
Dynamical Systems**

Lecture Notes in Control and Information Sciences 32.

Springer-Verlag, Berlin—Heidelberg—New York 1981.

Stran vi + 329; cena DM 36,50.

Knížka je výzkumnou monografií na oblíbený námět matematické teorie soustav — problém realizace, tj. konstrukce vnitřního popisu z vnějšího. Je již druhou monografií na tento námět v řadě Lecture Notes — recenzi Sontagovy monografie „Polynomial Response Maps“ viz *Kybernetika 18* (1982), s. 455.

Autor začíná zavedením experimentálních podmínek a to pomocí monoidu všech úseků časových funkcí nebo distribucí, definovaných

na časovém intervalu $(0, t] \subset \mathbb{R}$. Dále uvažuje soustavy počínaje lineárními, přes soustavy lineární nestacionární a končí algebraickými nelineárními stacionárními soustavami. K těmto typům soustav konstruuje pro dané zobrazení vstupu na výstup některé přirozené realizace, alternativně založené na dosažitelnosti a pozorovatelnosti. Autorovy konstrukce v sobě zahrnují konstrukce podle Nerodeho, B. L. Ho, Kalmana a dalších. Kromě nich uvádí autor množství jim dosažitelných výsledků.

V závěru uvedeme jednu kritickou výtku. I pro lineární stacionární soustavy, autorem zavedené monoidy vstupních úseků tvoří lineární prostor. To je v souladu s běžnými knihami věnovaným Laplaceově transformaci a lineárním soustavám. Autoři těchto knih, přicházející k problematice lineárních soustav z jiných oborů, zavádějí vstupy lineárních stacionárních soustav ze třídy funkcí pro něž existuje Laplaceova transformace. Domnívám se, že jedinými přirozenými vstupními funkcemi jsou racionální funkce, tj. funkce patřící do téhož algebraicky uzavřeného světa jako lineární stacionární soustavy.

Antonín Vaněček