Experience in Developing an In-House Molecular Information and Modeling System[1]

JAMES KAO,* VICTOR DAY, and LORAINE WATT

Research Center, Phillip Morris, U.S.A., Richmond, Virginia 23261

Received December 3, 1984

In this paper we present our views on the pros and cons of buying or developing molecular information and modeling software and possible problems to be encountered if the decision is the latter. We also present our current molecular information and modeling system (MIMS) and planned enhancements. Our own front-end program, MOLBUL, will be described. MOLBUL is a friendly generalized interactive computer program, which allows users to build up 2-D and 3-D structures and to prepare graphic art work for presentation with various graphics devices. The 3-D structures are used for modeling and structure–activity relationships while the 2-D structures are for on-line substructure searches of our chemical information system. The decision to buy or to develop the software is a rather difficult one and is dependent on several important factors. We believe our hands-on experience and recommendations may be useful to persons having an interest in this area.

INTRODUCTION

The advance of computer technology in the past decades has had a revolutionary impact on almost every field of business and science. The manpower expense has increased gradually while the computer hardware costs have declined sharply. Similar trends are expected to continue for the future, and computerized automation will be a major objective of every organization to increase productivity while cutting down the incurred costs. It is now highly justified to encourage chemists to computerize their information processing and to perform molecular modeling.

The aspects of molecular information and molecular modeling can be generally described as (a) the representation of structural formula and chemical reaction, (b) the acquisition and processing of experimental and theoretical data, (c) the display of structure, chemical equations, and graphics data, and (d) the storage and retrieval of chemical information. Many pharmaceutical and chemical companies have recently scaled up activities in chemical information and molecular modeling. Although we are in a different business, we recognize its potential benefits to our research and development efforts. Because of the demands of our scientists, our managers have created a special project, molecular information and modeling system (MIMS), to determine the needs, to perform assessment of available tools, and to integrate both needs and tools in the area of molecular information and modeling to serve the best interests of the entire research and development community.

In this paper we describe our process and efforts in various steps of this project. The discussion will not be on the value of molecular information and molecular modeling but on the approach to setting up such capabilities. Hopefully, our hands-on experience will be useful to persons or companies having an interest in this area.

DEFINING USERS' NEEDS, DEVELOPING A CONCEPTUAL (LOGICAL) MODEL, AND SOFTWARE SPECIFICATIONS

The aim of the MIMS project is to help scientists in the area of molecular information and molecular modeling. Defining users' needs and understanding their priorities are very important steps. Our experience has told us that an incorrect design or specification of a large application system may cause a long delay of delivery time and may even cause the delivery of a system not user friendly to our scientists. To understand priorities prior to design was extremely important since the MIMS scope is so broad and we were certainly unable to attack all problems at the same time. We proceeded to define users' needs and priorities by talking with our scientists and by polling them with a questionnaire. The results obtained from these communications helped us a lot in developing an ideal conceptual (logical) model and its specifications to be implemented in research and development. Our developed conceptual (logical) model of an ideal MIMS is described in Figure 1. The functions and software requirements of each module are described in the following:

Data Handler. Users log into this module to perform any specific tasks. It is also the link among other modules. Users can access this module to extract information from one module and process it in another module.

This module should be able to allow users to deposit or retrieve information by entering molecular (2-D or 3-D) structures (or substructures) through either graphics or text mode. The graphics input/output mode is an important requirement since chemists are so used to graphic representations of chemical structures. A graphics capability will tremendously increase productivity and efficiency of research chemists. Processes incurred in this module should be fast, interactive, and user friendly.

Chemical Information System. This module performs real organization, retrieval, storage, and search of chemical information. It stores both structure and alphanumeric information. Preferably, it should have the advantages of a good commercial relational database management system (RDMS).

The system must provide a wide variety of search types. It should be user friendly and know how to deal with occasional users who have a fear of breaking something. Fast searching is necessary since it would be a waste for scientists to wait in front of a terminal for a search to be done.

The chemical information system will include the following databases: (a) in-house (2-D and 3-D) molecular structures; (b) in-house molecular properties (flavor, MS, IR, NMR, etc.); (c) in-house bibliographic information; (d) external databases (NBS/Wiley MS database, Cambridge Crystallographic database, etc.). Close links between databases and modeling systems are required.

Theoretical Chemistry. Theoretical chemistry uses physical and chemical insight, sophisticated mathematics, and computers to calculate properties of a physical system. Theoretical calculations have been widely applied to interpret and organize experimental results and to uncover chemical mysteries. No theoretical tool is perfect, and complementary use of these tools should be encouraged. Theoretical tools in the areas of quantum mechanics and molecular mechanics are of particular interest to us.

Molecular Modeling. This module makes use of powerful computational software and hardware and interactive graphics

0095-2338/85/1625-0129$01.50/0

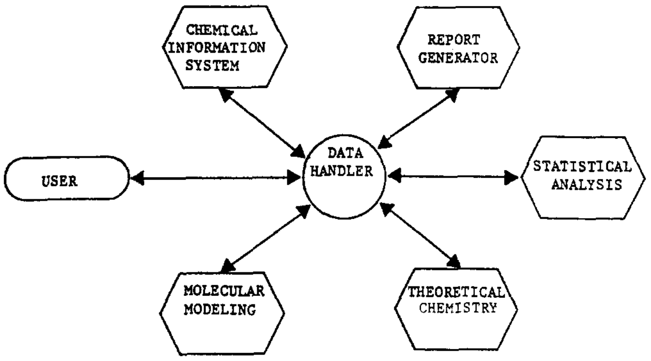


Figure 1. Conceptual structure of our molecular information and modeling system (MIMS).

systems to display molecular structures, molecular shapes (both steric and electrostatic), molecular interactions, and other properties. This is a useful tool for chemists to understand spatial requirements for substrate–receptor interaction. Graphic representations obtained from molecular dynamics simulations are another feature of molecular modeling.

Statistical Analysis. This module applies statistical techniques to correlations of molecular properties and molecular descriptors or SAR (structure–activity relationships). In principle, statistical packages such as MINITAB, BMDP, SPSS, TROLL, or SAS are applicable for this type of application. However, an integrated system is preferred due to the nature of users' environments. It will be able to handle various methods for SAR (additivity, multiple analysis, pattern recognition, mathematical models, etc.).

Report Generation. Users employ this module to effectively produce reports from information obtained from other modules. Ideally, this module will allow users to design their own report format. A screen painter is useful for this purpose. The report may consist of text and/or graphic information. Graphics tools to produce contour maps, surface pictures, etc. belong to this module. An important immediate application of this module is to allow chemists to prepare graphics art work for publications.

EVALUATION OF EXISTING SOFTWARE

After defining users' needs and understanding systems requirements, it is important to look around for existing software. If it is possible, we will obtain existing software instead of reinventing the wheel. The published software may be classified into three different groups according to the origin of sources: commercial systems, nonprofit organizations, and personal channels. The software costs also vary greatly. Software can sometimes be obtained for free through personal channels. Not-for-profit (government) organizations such as NIH, EPA, QCPE, etc. may charge nominal fees (up to ca. $5000), while commercial systems may cost up to $200 000.

It is also important to note that the software in the molecular information area is essentially commercialized or it is company proprietary with the exception of the NIH/EPA SANSS system. Other types of software originated mainly at universities, and they could be obtained almost free when we started about 2 years ago. However, the situation is gradually changing now. Companies such as Molecular Design Ltd., Chemical Design Ltd., Tripos, etc. are marketing molecular modeling software by essentially enhancing user interface and using better hardware.

The following commercial molecular information systems were published and were commonly known when we started to investigate: CAS ONLINE (Chemical Abstracts Service);[2] CHEMPIX (Chemical Information Management Inc.);[3] COUSIN (Compound Search Information System, Upjohn);[4] CROSSBOW (Computerized Retrieval of Organic Structures Based on Wiswesser, ICI-Fraser-Williams Ltd.);[5] DARC (Description, Acquisition, Retrieval and Correlation, University of Paris-Telesystems);[6] MACCS (Molecular Access Systems, Molecular Design Limited).[7]

The areas of particular interest to us for evaluation of software are costs, hardware (both host and terminal) requirements, response speed, search algorithms, user interface features, software functionalities, and source code availabilities. We gathered necessary information by reading published material, talking to salesmen, attending presentations and demonstrations, and communicating with a variety of users. After our initial evaluations, a lucid paper by Warr[8] came to our attention. She also looked into the molecular information software through similar angles as ours and obtained similar results. Her conclusion is "Different organizations have different user requirements and priorities and it is in fact impossible to classify the software packages as 'good,' 'bad' or 'best,' except in terms of a combination of some subjective requirements and the importance placed upon each requirement." Although we will not repeat her results here, we shall make a few comments that are relevant to our following discussions.

The existing molecular information software is highly device dependent and locks into a particular type of host cpu as well as graphics terminal. The required peripheral devices are sometimes very expensive while compatible devices on the market may be much cheaper. In fact, only MACSS has DEC-20 versions, and it still would have needed to be converted to our new operating system at that time. No companies would sell source codes for molecular information software at that time, although the situation is changing now. Most companies offer either molecular information software or molecular modeling systems, and there is no integration between these two systems. Traditionally, molecular modeling and information are handled differently although many companies would like to tie them together now. Apparently, integration of both will increase the usage of a company's resource and increase the productivity of research personnel in research and development efforts. Molecular Design Ltd. also sells software for molecular modeling that can be interfaced to MACCS, but the integration appears to be inadequate for our needs. However, conversion to our machines would still be necessary. Furthermore, Molecular Design Ltd. does not offer other database software for applications (such as spectra and flavor databases) that are important to us. Thus, there is no real turnkey system fitting our specifications.

Another important and interesting finding after talking to a variety of users is that their comments about a particular molecular modeling system are quite different, varying from very positive to very negative. It is obvious that people make comments depending on their computer background and type of applications. People tend to get frustrated because they do not have source codes to make enhancements specific to their applications.

PROS AND CONS OF BUYING VS. MAKING SOFTWARE

Once we decided to have a molecular modeling and molecular information system, the next obvious question was should we buy or develop it in-house? The common pros and cons of buying and developing software are shown in Table I.

For a specific type of application, one tends to buy well-established production-mode software instead of developing it, since black box operation and long-term flexibility are tolerable. In fact, it may be even cheaper to buy than to develop the software for this case. However, the pros and cons become less clear for systems that are under development such

Table I. Common Pros and Cons of Buying and Developing Software

developing	buying
Advantages	
(1) specifically tailored	(1) immediately available
(2) extensible, integratable	(2) potential integration
(3) totally independent	(3) clear total-cost picture
Disadvantages	
(1) lead time	(1) less long-term flexibility
(2) no clear cost picture	(2) black box operation

as molecular information and molecular modeling. This is because no company can offer a complete package for our requirements, and no software can be run immediately and satisfactorily in our DEC-20 machines (vide supra). Specifically, the costs and flexibility issues were our major concerns about buying a turnkey system.

This is a new area, and a lot of development work has to be carried out. In fact, there is no real integration between molecular information and molecular modeling, and there is no single system that can fulfill our total needs. When we started to work in this area, no company would sell source programs. However, source programs may now be purchased from CHEMPIX and CROSSBOW. The lack of source code may be a roadblock to interface with other software or hardware systems.

Both hardware and software technology advanced rapidly. We switched from Xerox's Sigma system to DEC-2060 systems about 3 years ago. We have continued to be one of the first installations to implement new versions of operating systems and system software. Lack of source programs may cause incompatibility problems, and vendors have to be brought here, or we have to wait until the official version is released. In fact, any turnkey system purchased from outside has to be converted. The previous official announcement by DEC that the Jupiter project was canceled further complicated our situation since we had to change our original hardware plan. It is our recent decision to go for UNIX systems for any future hardware and software. Another important point that should be mentioned is that we have an IDM database system in-house. This type of hardware database machine performs 1 order of magnitude faster than software-based systems. It would be interesting and may be potentially beneficial to use this type of database machine for MIMS.

Without the source programs, any modifications or enhancements that are either desirable or necessary would thus require a contract with the vendor with probable great expense and delay. There are potential incompatibilities with existing and future tools. Thus, a turnkey system is not open-ended, and the purchaser has to be totally dependent on the vendor.

In our opinions, these commercial systems are expensive. Our research and development environment is different from a major pharmaceutical or chemical company with numerous chemists. Our managers are not ready to spend that amount of money without further justification. Cost effectiveness can be illustrated by the following simple mathematics. Let us say a turnkey system can save 1 h/month for each chemist, and there are two organizations, organization A with 1000 chemists and organization B with 100 chemists. Without a turnkey system, it would be necessary to hire six persons for the former and 0.6 person for the latter to do normal work. It becomes obvious that a turnkey system is considered to be cheap for organization A while expensive for organization B. By taking into consideration software costs, mainframe and terminal hardware costs, and annual maintenance, we predicted it would be cheaper to develop in-house instead of buying. Our predictions are different from the view of Howe.[9] However, this is presumably due to the fact that there was no public domain software available when he started his project.

RECOMMENDED APPROACH

In light of these concerns and system considerations, we recommended to management that we should start to develop an in-house molecular information and modeling system by adopting the following two approaches: (a) integrate our needs and *public* accessible tools to set up a prototype MIMS and (b) modify the prototype system to the production system.

Step (a) sets up a prototype MIMS that provides the necessary and detailed evaluation of users needs and requirements. A prototype can be used to verify the feasibility of the design. Moreover, it builds in-house expertise for future work. This is beneficial to the company since in-house expertise is required no matter which system we are going to have. One should try to use developed public-domain software as much as possible in order to minimize the lead time. The prototype system should show functionalities but does not need to have all modules integrated. The prototype system can be evolved into the final system. We expect to develop a flexible, open-ended, low-cost, and hardware-independent system. The tools developed in-house may be traded with other companies. This item cannot be overlooked. Today, scientists talk and publish articles about their software but will not sell it. The likely way to get it is to trade yours for theirs. We planned that, after developing the prototype system, we might purchase part of a commercially available software system to complete our MIMS if terms were acceptable to us.

SYSTEM DESIGN, PROTOTYPE, AND APPLICATIONS

After considering our system environment and future hardware and evaluating currently existing software, we have decided to develop our own in-house molecular information and modeling system. The motivation is that we prefer a system that stresses "program portability and maintenance", "on-line interaction", "integration", and "user friendliness". The general system design specifications are thus (a) it shall be able to be executed (with none or minimal modifications) on different machines, (b) it allows the user to promptly deal with the desired 2-D or 3-D structures from any graphics device (including low-price graphics terminals), and (c) it is equally convenient for use by either nonexperienced users or experienced users. We have made the following decisions to meet the objectives as closely as possible.

The programs of the system are preferably written entirely in ANSI 77 Fortran since Fortran is probably the most popular language in scientific communities and almost every computer for scientific applications can do the compilation. The graphics textual information will be mainly handled using the A1 format to increase machine independence since program portability is of more concern to us than memory usage. The program shall be modularly structured and documented to increase its portability and maintainability.

The Fortran programs have to communicate to a graphics device to display chemical structures. However, there are no common standards for communications between a Fortran program and a graphics device.[10] Thus, no fully machine-independent graphics program currently exists, due to a lack of common computer graphics standards. Fortunately, there are a few interactive graphic packages available on the market that come with different device drivers that can relieve the user's program of device-dependent features. We have chosen to use the PLOT-10 IGL of Tektronix,[11] since it is a widely accepted package and is currently available on our machine.

To handle two different types of users, minimal and extensive users, the program dialogues with the user have to be written as friendly and informative as possible for the former and as concise as possible for the latter. The branch-point technique will be used to facilitate this specification. The programs must be "friendly" interactive using plain English commands that are easy to remember. Templates or menus will be implemented when necessary. The programs must have HELP files to provide information about user commands. Information will be entered as free format, and the programs will have certain validating capabilities. Input entered from the keyboard may be upper- or lower-case letters. For instance, where lower-case letters must be retained for plotting (such as chemical symbols), no conversion will be carried out.

Our initial MIMS system has been developed. We will discuss the current MIMS configuration in the following sections.

MOLBUL: **Program To Build (Create) Structures.** Ways to represent molecular structures are the heart of any molecular information and modeling system. Chemists are used to graphic representations of chemical structures, and in fact, they draw them for their daily work with pencils, with stencils, and transfer symbols, or, more recently, with computer-graphic techniques. Three-dimensional (3-D) structures are important to better understand molecular properties. Chemists therefore use perspective views, wedges, etc. to represent 3-D features in their daily discussions and reports due to the limitation of the common communication medium, paper.

A similar 2-D approach for representing structures has been adopted in computer information processing. The practical reason is probably that experimental 3-D structures are not always known. However, it is also possibly due to the lack of adequate techniques to effectively represent and store 3-D structures at comparable computer costs. The common techniques used to represent 2-D structural (canonical) information are Wiswesser line notation (WLN) and connection tables.[12,13] Most of the chemical databases that are commercially available are based on one of these techniques. Recently, there has been much progress in utilizing computer graphics to build 2-D structures.

The demand for 3-D molecular structures has become much stronger due to the advance of theoretical chemistry and the decrease of computer hardware costs. A 3-D structure is the required input for all theoretical calculations such as molecular mechanics and quantum mechanics to derive optimized geometries.[14] It has been proved that theoretical structures obtained from sophisticated methods are as good as experimental ones. The 3-D molecular structures are necessary for detailed molecular modeling applications.

One way to obtain initial 3-D structures is to convert 2-D structures, which may be either retrieved from databases or constructed from a 2-D structure computer program. However, it is unfortunate that the conversion from 2-D to 3-D does not always work; in particular, it poses many problems in studying large and nonplanar structures. On the other hand, the conversion from 3-D to 2-D always works. For this reason, there have been efforts to develop programs that will make it easier to build 3-D structures. However, these packages have one or more of the following deficiencies: expensive hardware and software, unfriendly, limited in scope, company proprietary, machine dependent, and/or not integrated. A low-cost and friendly system to manipulate and build 3-D structures is then necessary in molecular information and modeling.

In light of these problems,[15] we developed the MOLBUL (*mol*ecular *buil*der) program, the front end of our system. MOLBUL is a versatile, efficient, integrated, and interactive program for chemists to build and display 3-D or 2-D molecular structures for theoretical calculations, graphic art work, and chemical information systems. To the best of our knowledge, no such unique integrated and friendly software is currently available elsewhere in the public domain.[16]

MOLBUL is a multifunction multipurpose tool. Here, we just illustrate features available in the GINPUT mode of MOLBUL for building chemical structures.

Procedures for building the 3-D structure shown in Figure 2, using the GINPUT mode of MOLBUL, are outlined in this example. There are easier and quicker ways to build this structure than the procedures presented here. However, the purpose of this example is to illustrate some of the more important template commands available in the GINPUT mode. The GINPUT mode is based on the interactive use of a graphic device that cannot be fully demonstrated here. Only a few CRT displays are illustrated.

Enter the GINPUT mode by issuing the GINPUT user command. The initial primary structure, a three carbon atom chain, is entered into the working area via the USER template command (Figure 2a). Next, one of the carbon atoms is replaced by a phenyl group with the REPLACE PHENYL template command (Figure 2b). The FRAGMENT template command is then selected to obtain the structures for the two alkyl substituents from the fragment database. This causes the programs to display a fragment template. To display the hydrocarbon fragments, the HYDROCARBON template command is selected (Figure 2c). By use of the SELECT FRAGMENT command, the *tert*-butyl and neopentyl fragments may be selected and then brought back to the main GINPUT template with the RETURN command (Figure 2d).

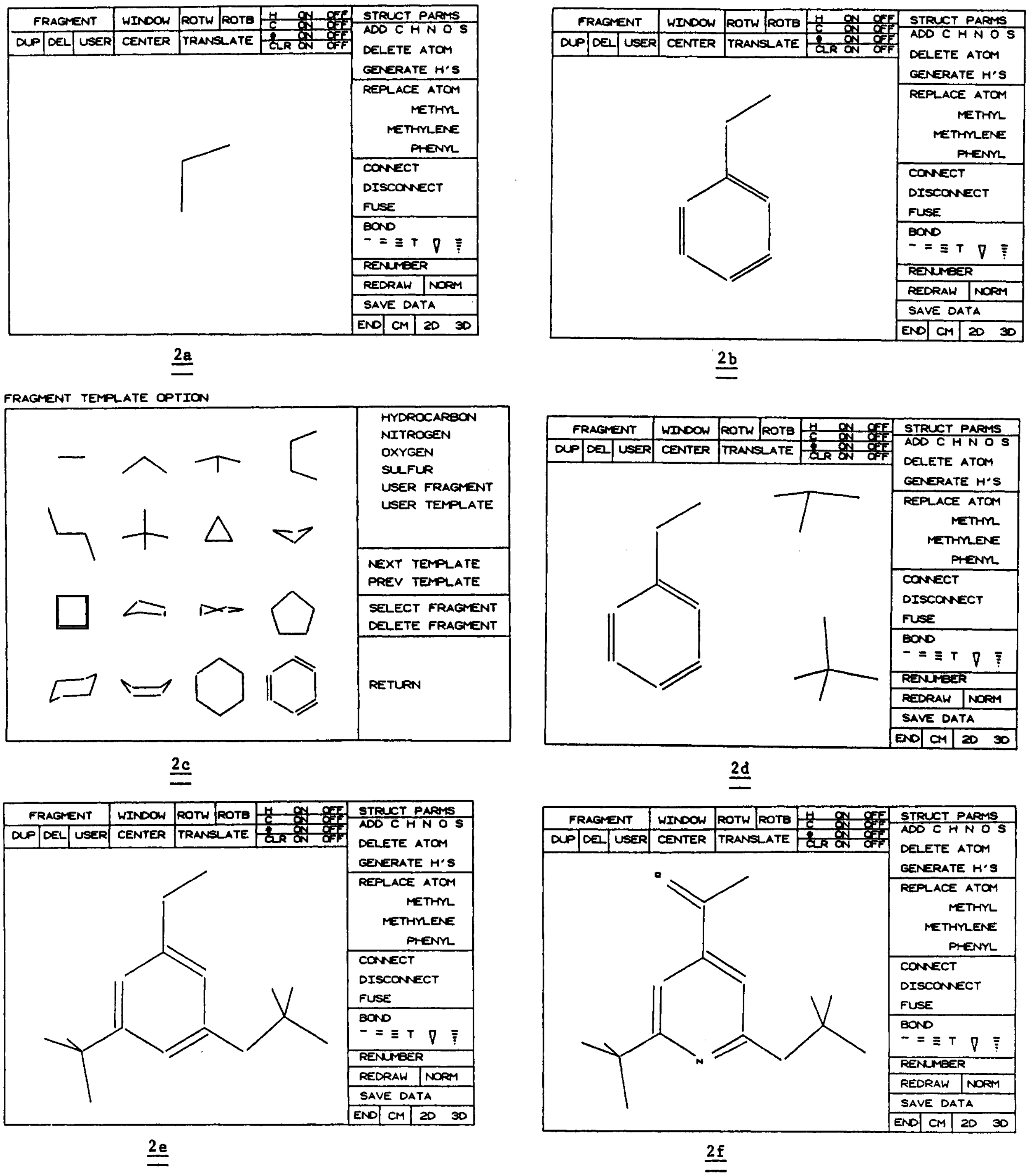
The two fragments are then connected by employing the CONNECT command (Figure 4d). To add the ketone functionality, the ADD O command is used followed by the BOND = command. Finally, the structure is completed by replacing a ring carbon with a nitrogen atom with the REPLACE atom command (Figure 2f).

Theoretical and Analytical Tools. Theoretical chemistry uses physical and chemical insight, sophisticated mathematics, and computers to calculate properties of a physical system. These computer programs can be readily obtained from QCPE for nominal fees or through personal channels without any charge. We have obtained many important tools in the areas of classical mechanics, quantum mechanics, and statistical mechanics, and most of them are now installed in our machines. Conversions often had to be done because of machine incompatibilities, and it took from a few hours to a few weeks. In addition, we have extended the Allinger's molecular mechanics program, MM2, to handle heteroconjugated systems,[17] as part of our ongoing research effort in flavors and organic conductors.

Theoretical calculations have become a much easier job after the development of our MOLBUL program. MOLBUL provides users with a convenient and efficient way to build structures through graphics devices. It has many data manipulation and display commands to facilitate the building process. The built structures can be saved and submitted for various theoretical calculations. By using MOLBUL, chemists no longer have to be bothered by routine input formats for theoretical calculations. With MOLBUL, the data preparation time is drastically reduced as compared with other traditional means.

In addition to existing statistical analytical tools, we developed two other programs, PMPLOT and NEWMAN,[18] to help chemists in analyzing and presenting their data. NEWMAN is very simple to use and is a versatile tool to draw quality Newman projections, while PMPLOT provides a friendly, efficient, and interactive environment for research scientists to display quality (smooth) *xy* plots, contours, and surfaces. Both programs can be either used as a stand-alone version or called from another program.

MOLINF: **System of Programs To Store and Retrieve Molecular Information.** The NIH/EPA Chemical Information System (CIS)[19] has all the molecular information modules that



2a

2b

2c

2d

2e

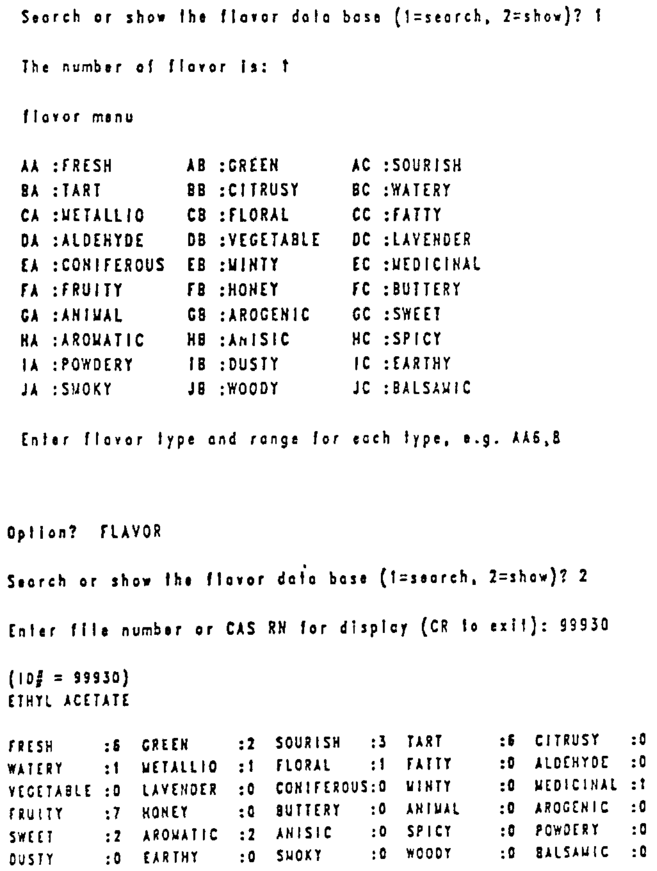
2f

Figure 2. Displays to illustrate the GINPUT operation.

we wanted to start with. The CIS SANSS (*S*ubstructure *a*nd *N*omenclature *S*earch *S*ystem) and MSSS (*M*ass *S*pectrum *S*earch *S*ystem) are of particular interest to us. The system can run on a DEC-20 machine; however, the database files are essentially updated batchwise on an IBM mainframe. We were encouraged by its successful usage at the SERC Laboratory at Daresburg in the U.K. and Lederle in the U.S.

The NIH/EPA CIS appeared to be an ideal candidate for our system environment. In principle, we thought we should not have much difficulty in bringing SANSS and MSSS up and running on our machines. However, as soon as we received the appropriate tapes from National Technical Information Service (NTIS), we encountered many problems. The NTIS tapes we received were more or less like archived tapes that contain thousands of files together for the entire CIS. Several files have about 20 versions, and there is little documentation. An even more serious problem is that there are missing subroutines. These types of problems have drastically slowed down our progress. However, we should point out that although there are problems with implementing CIS, it is still good software.

Although we have encountered many difficult problems through all phases of development, we have solved most of these problems through various approaches. We have written



```
 Search or show the flavor data base (1=search, 2=show)? 1

 The number of flavor is: 1

 flavor menu

 AA :FRESH        AB :GREEN        AC :SOURISH
 BA :TART         BB :CITRUSY      BC :WATERY
 CA :METALLIO     CB :FLORAL       CC :FATTY
 DA :ALDEHYDE     DB :VEGETABLE    DC :LAVENDER
 EA :CONIFEROUS   EB :MINTY        EC :MEDICINAL
 FA :FRUITY       FB :HONEY        FC :BUTTERY
 GA :ANIMAL       GB :AROGENIC     GC :SWEET
 HA :AROMATIC     HB :AnISIC       HC :SPICY
 IA :POWDERY      IB :DUSTY        IC :EARTHY
 JA :SMOKY        JB :WOODY        JC :BALSAMIC

 Enter flavor type and range for each type, e.g. AA6,8


Option?  FLAVOR

Search or show the flavor data base (1=search, 2=show)? 2

Enter file number or CAS RN for display (CR to exit): 99930

(ID# = 99930)
ETHYL ACETATE

FRESH      :6  GREEN     :2  SOURISH   :3  TART      :6  CITRUSY   :0
WATERY     :1  METALLIO  :1  FLORAL    :1  FATTY     :0  ALDEHYDE  :0
VEGETABLE  :0  LAVENDER  :0  CONIFEROUS:0  MINTY     :0  MEDICINAL :1
FRUITY     :7  HONEY     :0  BUTTERY   :0  ANIMAL    :0  AROGENIC  :0
SWEET      :2  AROMATIC  :2  ANISIC    :0  SPICY     :0  POWDERY   :0
DUSTY      :0  EARTHY    :0  SMOKY     :0  WOODY     :0  BALSAMIC  :0
```

Figure 3. Displays to illustrate the FLAVOR database.

database loading programs for the DEC-20 systems and modified search algorithms. Templates have been designed to improve user interface. Standard file format has been implemented, and interface with molecular modeling is possible. We have rewritten the graphics subroutines to interface with PLOT-10 IGL in order to achieve device independence. We also upgraded the system from TOPS-20 3.1 to TOPS-20 4.1 operating system and from Fortran-10 to Fortran-77 compiler.

Our current molecular information system (MOLINF) consists of three prototype databases, STRUCTURE (2-D and 3-D structures), MASS (mass spectra), and FLAVOR. Here, we shall give a little flavor of our FLAVOR database. As depicted in Figure 3, the user can search or show the flavor profile. By cross-reference between FLAVOR and STRUCTURE, chemists will be able to perform structure–activity correlations.

MOLPRO: **Program To Calculate Molecular Descriptors.** The molecular properties program (MOLPRO) is used to calculate molecular descriptors such as surface area, volume, and shape analysis. Most of the Fortran code came from CHEMLAB and other public-domain software.[20] The program can calculate the surface area for the molecular structure only or for the structure and a layer of chosen solvent. The shape-analysis algorithm is based upon the comparison of overlap volume of molecular structures.

The modifications of the Fortran code include the converting of CHEMLAB atom types to molecular mechanic atom types and the changing of the I/O statements to read files in the MOLBUL format.

Prototype Demonstrations. Prototyping is a key technique employed throughout the development of this project. During the development phase, prototype demonstrations were conducted to a selected individual or group with the following purposes in mind: (1) brief on the progress and status of the MIMS project; (2) solicit suggestions about future enhancements and developments; (3) complete a wish list and set up priorities; (4) pave a way to scale up limited applications.

The tools were designed to help chemists and were designed from the beginning with them (end users) in mind. The prototype demonstrations provided a communication channel between chemists and computer scientists. After a demonstration, the chemist usually provided a more accurate definition of what was needed for his applications. On a few occasions, we even let chemists apply our prototype system to their chemical problems, and we have benefited a lot from this type of limited application. The chemists can test the system in a more physically meaningful way than a programmer. The chemists generally enjoyed the participation and quickly got intellectually involved in the system. Their use of the prototype system has resulted in several publications.[21]

Enhancements and Future Plans. The prototype MIMS has proved that it is feasible to implement an in-house system for our research and development needs. The system is currently evolving into a final system. Some of developed modules have been released to our scientists. Many scientists have attended training sessions, and the responses are very positive. Many more activities are under way or planned; some of them will be described below.

The following databases have been required to facilitate MIMS applications, and they will be loaded into the system for general usage: NIH/EPA SANSS database, Cambridge crystallographic database, Wiley/NBS mass spectral database, and private collections of theoretical structures (MM, AB INITIO, etc.).

These databases will become the so-called centralized database. We shall provide links between these databases and private ones. Interface or integration with laboratory systems and external databases will be studied.

SANSS uses inverted files in order to speed up search but makes on-line updating a more complicated problem. Redesign of these index files with commercial DBMS techniques may be more preferable in the long run. Integration of statistical tools into the MIMS system should be addressed to expedite SAR activities. To make the SANSS system machine independent and to install it in a UNIX environment is one of our highest priorities. The potential for using new hardware technology will be further explored. Finally, our ultimate goal is to perform successful applications of MIMS in research and development.

CONCLUSIONS

The existing commercial systems in the molecular information and modeling area are not mature yet from the viewpoint of flexibility, integration, and general applications. It will be interesting to see how these systems evolve and to what extent various difficulties are overcome. Due to its complexity, a major effort is required to develop (customize) an in-house molecular information and modeling system. To develop a system is not a simple task, and it needs full support from management. Coordination among a multidisciplinary team and devotion of personnel involved are key factors for success. Data control is essential for setting up useful databases. The decision to buy or develop the software is rather difficult and is dependent on several important factors discussed above.

After considering our system environment, type of applications, and future hardware development, we have decided to develop our in-house system. Although we have encountered many difficult problems through all phases of development, we have now solved most of these problems through various approaches. In particular, we have developed our own friendly front-end program, MOLBUL, which allows users to build up both 2-D and 3-D structures for modeling, structure–activity relationship, and information system. MOLBUL can also be used to provide graphic art work (with text and graphics) for presentations. We believe our hands-on experience and recommendations would be useful to persons having interest in this area.

ACKNOWLEDGMENT

We thank our managers, R. Thomson and R. Waugh, and our numerous outstanding scientists for their support, participation, and various contributions. Without their help, this project will never be successful.

REFERENCES AND NOTES

(1) Presented in part at the 188th American Chemical Society National Meeting, Philadelphia, Pennsylvania, 1984.

(2) *CAS Report*, Issues 9 (Oct 1980), 10 (July 1981), and 11 (Nov 1981). Available from Marketing Communications Department, Chemical Abstracts Service, P.O. Box 3012, Columbus, OH 43210.

(3) Moreau, G. *Nouv. J. Chim.* **1980**, *4*, 17.

(4) Howe, W. J.; Hagadone, T. R. *J. Chem. Inf. Comput. Sci.* **1982**, *22*, 8, 188.

(5) Eakin, D. R.; Ash, J. E.; Hyde, E. "*Chemical Information Systems*"; Horwood: Chichester, U.K., 1975. Townsley, E. E.; Warr, W. A. *ACS Symp. Ser.* **1978**, *84*, 73–84. The marketing rights for CROSSBOW are held by Fraser-Williams (Scientific Systems) Limited, Glendower House, Poynton, Cheshire SK12 1NJ, England.

(6) Dubois, J. E.; Viellard, H. *Bull. Soc. Chim. Fr.* **1968**, 900. Dubois, J. E.; Anselmini, J. P.; Chastrette, M.; Hennequin, F. *Bull. Soc. Chim. Fr.* **1969**, 2439. Dubois, J. E.; Viellard, H. *Bull. Soc. Chim. Fr.* **1971**, 839. Attias, R. Presented at the 182nd National Meeting of the American Chemical Society, New York, Aug 1981. DARC is marketed by Télésystèmes Darc, 92110 Boulogne, France.

(7) Dill, J. D.; Hounshell, W. D.; Marson, S.; Peacock, S.; Wipke, W. T. Presented at the 182nd National Meeting of the American Chemical Society, New York, August 1981; MACCS is marketed by Molecular Design Ltd., 1122 B Street, Hayward, CA 94541.

(8) Warr, W. A., personal communication.

(9) Howe, W. J. *Drug. Inf. J.* **1984**, *18*, 179.

(10) For example, Martin, P. *Economist* **1982**, *Aug. 67*.

(11) Tektronix, Inc., P.O. Box 500, Beaverton, OR 97077.

(12) Wiswesser, W. J. *Comput. Automat.* **1970**, *19*, 2. Smith, E. G. "The Wiswesser Line-Formula Chemical Notation"; McGraw-Hill: New York, 1968. Hyde, E.; Matthews, F. W.; Thomson, L. H.; Wiswesser, W. J. *J. Chem. Doc.* **1967**, *7*, 200. Thomson, L. H.; Hyde, E.; Matthew, F. W. *J. Chem. Doc.* **1967**, *7*, 204. Wipke, W. T.; Heller, S. R.; Feldmann, R. J.; Hyde, E., Eds. "Computer Representation and Manipulation of Chemical Information"; Wiley: New York, 1974.

(13) Feldmann, R. J.; Milne, G. W. A.; Heller, S. R.; Fein, A.; Miller, J. A.; Koch, B. *J. Chem. Inf. Comput. Sci.* **1977**, *17*, 157. Fujiwara, Y.; Nakayama, T. *Anal. Chim. Acta* **1981**, *133*, 647. Moreau, G. *Nouv. J. Chim.* **1980**, *4*, 17. Farmer, N. A.; O'Hara, M. P. *Database* **1980**, *3*, 10. Howe, W. J.; Hagadone, T. R. *ACS Symp. Ser.* **1978**, *84*, 107. Wipke, W. T.; Dyott, T. M. *J. Am. Chem. Soc.* **1974**, *96*, 4825; 4834.

(14) Burkert, U.; Allinger, N. L. "Molecular Mechanics"; American Chemical Society: Washington, DC, 1982. Osawa,; Musso, H. *Top. Stereochem.* **1982**, *13*, 117. Kao, J.; Allinger, N. L. *J. Am. Chem. Soc.* **1977**, *99*, 975. Kao, J.; Huang, T. N. *J. Am. Chem. Soc.* **1979**, *101*, 5546.

(15) Kao, J.; Eyermann, C.; Watt, L.; Maher, R.; Lilly, D., submitted to *J. Chem. Inf. Comput. Sci.*; presented in part at the 2nd Conference on Computers and Chemistry, Tallahassee, FL, 1983, and the 25th Annual Medicinal Chemistry Symposium, Buffalo, NY, 1984.

(16) This program will be submitted to QCPE for distribution.

(17) Kao, J.; Leister, D.; Sito, M. *Tetrahedron Lett.*, in press.

(18) Kao, J.; Watt, L., *Comput. Chem.*, in press.

(19) Milne, G. W. A.; Heller, S. R. *J. Chem. Inf. Comput. Sci.* **1980**, *20*, 204.

(20) Hopfinger, A. J. "*Conformation Properties of Macromolecules*"; Academic Press: New York, 1973. Hopfinger, A. J. *Arch. Biochem. Biophys.* **1981**, *206*, 153–163. Hopfinger, A. J.; Potenzone, R. *Mol. Pharmacol.* **1982**, *21*, 187.

(21) Kao, J.; Seeman, J. I. *J. Comput. Chem.* **1984**, *5*, 200. Kao, J.; Katz, T. *THEOCHEM* **1984**, *108*, 229. Kao, J.; Eyermann, C.; Southwick, E.; Lilly, D. *J. Am. Chem. Soc.*, in press. Houminer, Y.; Kao, J.; Seeman, J. I. *Chem. Commun.* **1984**, 1608.

BOOK REVIEWS

Pharmacochemistry Library. Volume 7. Theoretical Drug Design Methods. By Rainer Franke (Institute of Drug Research). Elsevier, New York, 1984. 412 pp. $75.00.

This book translates and updates a 1980 German book on quantitative structure–activity relationships (QSAR) published by Academie Verlag, Berlin. The book consists of 14 chapters, an appendix, and index. The first two chapters present a brief introduction to the general combinatorial problems of drug design and mathematically describes some techniques used to define biological response.

Chapter 3 introduces extrathermodynamic approaches to QSAR. A series of chapters follow that address biophysical interactions (hydrophobic, electronic, or steric) that have been shown to be important for structure–activity studies.

A chapter of the interrelations between various biophysical parameters follows. This chapter covers well the problems of collinearity and multiple collinearities between the commonly used hydrophobic, electronic, and steric molecular parameters. It is followed by an extensive discussion of specific applications for the extrathermodynamic QSAR approach. Sets of data are discussed in detail and carried through the various phases of QSAR analysis in this chapter. A brief description of factor analysis as a tool for preprocessing data in QSAR analysis includes literature examples. Extensive sections on pattern recognition techniques in QSAR and non-computer-assisted techniques, such as the Topless decision tree, are covered next.

A discussion of QSAR approaches that use structural parameters includes and extensive presentation of the Free–Wilson approach and compares these approaches to extrathermodynamic approaches. Substructural techniques, including the development of topological and geometrical descriptors, are covered in detail. This book also covers rapidly developing areas of computational chemistry including receptor mapping, pharmacophore structure development, and computer graphics used for molecular matching and superimposition.

The appendix presents a short overview of the principles of linear regression analysis. This appendix would not be sufficient to learn the theory and application of the statistical techniques of regression analysis but does serve as a centralized source of the most commonly used equations and includes definitions of relative terms.

This book presents the advanced investigator with an overview of the rapidly changing QSAR field and provides a sufficient number of references to the primary literature. The book is mathematically oriented and is probably too detailed for a text. In many places, it is apparent that the book is translated from German. There are more typographical errors than usual.

Michael Cory, *Burroughs Wellcome Co.*