ARTIFICIAL INTELLIGENCE IN CHEMISTRY

Vladimír JAKUŠ

Department of Organic Technology,
Faculty of Chemical Technology, Slovak Technical University,
Detached Laboratory at the Research Institute for Petrochemistry, 971 04 Prievidza

Received March 10, 1992
Accepted June 8, 1992

Dedicated to Dr Miloš Kraus on the occasion of his 65th birthday.

The definition of artificial intelligence and the associated tasks of this branch of science are discussed. The tasks include pattern recognition, adaptation and learning, problem solving by means of expert systems or neural networks, and understanding the natural language and communication with a machine in it. The principles of problem solving are analyzed. It is demonstrated how artificial intelligence-based computer programs in which chemical expertise is encoded assist in structure elucidation, in the investigation of relations between structure and biological activity or chromatographic retention, etc.; problems emerging in the synthesis planning with a retrosynthetic analysis, or in the planning of experiments and intelligent consultations are dealt with. Several models used for structure elucidation and synthesis planning are evaluated. An overview is presented of additional expert systems which, along with artificial intelligence-based robotics, are used in intelligent instrumentation. Also discussed is the role of neural networks, which begin to be successfully employed in structure elucidation, synthesis planning, in intelligent instrumentation and in the treatment of natural languages. They are expected to be an important tool in the implementation of intelligent systems for the classification of chemical databases and prediction of properties of molecules.

1. INTRODUCTION

After 25 years of development, artificial intelligence begins to come to the fore for application not only in science and health service but also in the diverse branches of industry including banking and commerce[1 - 3]. The increased interest of industrial companies in the application of artificial intelligence is a consequence of advancing automation of the management and control process where conventional technology fails to fulfil such tasks.

The most important field of research into and application of artificial intelligence involves expert and knowledge-based systems, which function not only on special, so-called LISP-machines but now also on conventional computers ranging from PC's to mainframes. Work on increasing the efficiency of expert systems is in progress; this can be achieved by a suitable integration of expert systems with the existing data processing environment, i.e. with various data banks, or by applications which are set up, e.g., in COBOL, PL 1 or ASSEMBLER. Another field of interest concerns systems using natural language, speech discrimination, intelligent text recognition, machine translations and neural networks. No rapid progress of these technologies is expected in the near future. The new role of information systems which will actively use artificial intelligence principles will be important as well. On the whole, a rapid growth of intelligent technologies is expected in the 1990's.

2. INTELLIGENCE AND ARTIFICIAL INTELLIGENCE

Intelligence can be related to reasoning, capability of perception, and opinions. The German psychologist W. Stern, the originator of the idea of the intelligent quotient, regarded intelligence as the ability to adapt to new (unknown) tasks and life conditions. Intelligence can also be considered to be the assembly of intellectual abilities, enabling

acquired knowledge to be used efficiently in the solution of new (unknown), theoretical and/or practical problems[4].

Artificial intelligence is the application of mathematical techniques and/or mathematical logic in research devoted to any aspect of intelligence, done by technical or theoretical means[5,6].

Artificial intelligence can also be characterized as follows[7]: "Artificial intelligence is the property of man-made systems having the ability to recognize objects, phenomena and situations, to analyze relationships between them and so to create internal models of the world in which these systems exist, and based on that, to make reasonable decisions and, by means of their capabilities of predicting consequences of such decisions, to disclose new regularities between the various models or their groups". Based on that characteristics, the tasks of artificial intelligence can be divided into the following groups:

a) Pattern recognition
b) Adaptation and learning
c) Problem solving by means of expert, knowledge-based and hybrid systems
d) Understanding natural language and communication with machines in natural language.

2.1. Pattern Recognition

Pattern recognition deals with the problem of dividing objects into classes. For this task to be meaningful, one has first to define the system, i.e. to specify the aspect according to which the object is examined, to establish the set of quantities which will be examined and measured for the object, and to define the time, spatial and resolution levels of measurement. Systems, and hence objects for which they have been defined, can be divided into two or more classes based on identical or mutually closely approaching values of measured quantities or other characteristics extracted from the quantities. The set of measured quantities is usually referred to as the image.

At present, pattern recognition tasks include the processes of quantity measurement, image formation and image classification. Pattern recognition methods are classed in two major groups with respect to the image representation used:

a) Symptom methods
b) Structural methods.

In symptom methods, images are represented by n-dimensional vectors of numerical values – symptoms. To each vector is attributed a single point in the n-dimensional, so-called image space.

In structural methods, images are described by sets of basic descriptive elements – primitives, their properties and interrelations. Primitives are the minimal qualitative characteristics which are determined in the image. The structural methods are also

referred to as syntactic or linguistic methods because they are similar to those applied to the investigation of the structure of sentences and the grammatical structure of the natural language.

Methods of two-class classification (machine learning)[8] and multicategorial pattern recognition methods (KNN[9], branching tree[10], SIMCA[11], cluster methods[12]) have found application in analytical chemistry. Pattern recognition applications in chemistry have been reviewed in publications[13 - 21]. Pattern recognition methods and algorithms are also used in practice such as technical and medical diagnostics, aerial and satellite photograph processing, etc. Computer visualization and robotics are two wide fields of application of pattern recognition techniques.

2.2. Adaptation and Learning

An important property of living organisms is their ability to adapt or to learn based on their experience. Equipping technological systems with this ability is one of the goals of artificial intelligence. In technical learning systems, the learning stage is usually separated from the stage of function of the system. During the learning, a finite so-called training set of cases is submitted to the system; usually, each element of the set is supplemented with tutor's information about the expected response of the system (tutorial learning). If information from the tutor is lacking, the learning is without a tutor. Learning can be regarded as an individual process of adjusting the optimum parameters (occasionally also the structure) of the system[22]. After the learning is finished, the system is optimally adjusted, by means of the criterion chosen, with respect to the training set. In the later function of the system it is assumed that the statistical characteristics of the set of problems solved do not differ appreciably from the characteristics of the training set.

In pattern recognition tasks, learning serves to find the decision rule. In symptom methods, parameters of the decision rule are sought in a form which is known beforehand; the decision rule itself can also be sought. In syntactic methods, appropriate grammar is to be derived from available words of the language.

Learning methods also begin to find use in the field of expert systems, in the inductive building of the knowledge base or its part if some expert knowledge is lacking.

Adaptation and learning methods are used now with success in setting up a mathematical description of objects and systems, in designing adaptive industrial regulators and control algorithms, in pattern recognition when adjusting decision vectors, and in robotics in generating and implementing motions of the robots.

2.3. PROBLEM SOLVING

A significant feature of intelligent systems is their ability to form an internal model of the environment (world) and to work with it. If the starting and target states of the environment are described, a sequence of actions is subsequently sought by which it is possible to pass from the starting model to the target model. This sequence can be referred to as the plan and the methods of creating plans, as problem solving. Various procedures can be applied to the plan creation. Mentally simplest is the "blind" exhaustive search through the state space. Leading to a combinatorial explosion of the possible solutions, this approach has a limited applicability. Search is manageable in a reasonable time only by using problem-oriented knowledge, the so-called heuristics. The extent of unavoidable search depends on the heuristics efficiency. The system will be the more intelligent the less it will have to search through the state space[7].

2.3.1. *Representation of Knowledge*

An important problem which is related to problem solving is that of an efficient representation and use of knowledge. This problem cannot be disregarded in any artificial intelligence project because the computer does not work with physical objects; it invariably works with some symbolic representation. And the choice of a particular representation can have a substantial effect on the resulting effect of the whole project.

The declarative way of knowledge representation[23] has found wide application. The knowledge of the problem state is expressed by descriptions of discrete states and the descriptions of the possible changes are expressed by descriptions of operators. Among declarative procedures of representation of knowledge is that based on predicate calculus[24 - 26]. Predicate calculus also enables the procedural aspect of knowledge representation. In procedural representations, attention is centered on specifying the possibilities of manipulation with objects, with their properties and interrelations[27]. The language PLANNER[28] is the best known language enabling a procedural representation of knowledge.

Efforts to express semantic relationships within the problem environment directly in the formalism of the representation led to the creation of means of semantic representation[29]. So-called semantic networks[30] can be imagined as oriented graphs with evaluated vertices and edges. The vertices of the semantic network correspond to objects, notions, conceptions, ideas, relationships, functions, activities, etc. The edges denote the membership of the object in a higher-level notion, the relation of notions to higher-level notions such as "who", "what", "what about", etc.

The last way of representation is frame representation[31]. The frame is a structure representing a stereotypic situation. It is a date structure containing simultaneously all relevant knowledge concerning the object in question. Frames resemble semantic networks but they have a deeper structure. Frames can also be combined into frame

systems which can act as separate frames and also enable the procedural aspect of the problem environment.

Knowledge representation is actually a way of recording knowledge in the computer. The choice of a suitable knowledge representation as a scheme for expressing and storing in the computer memory must be an appropriate compromise between the requirement of the semantic association of knowledge (requirements that data concerning one object should be located in the memory block "close to one another" along with information on their interrelations) and the requirement of modularity (requirement that there should be possible a simple expansion and modification of the knowledge stored). Formalisms of semantic networks, frames and scenes are examples of association of related knowledge, whereas production systems or 1st order predicate logic rules are examples of modular representation. For real systems, several basic formalisms of knowledge representation are usually combined so that the representation for the given field of knowledge be as convenient and appropriate as possible.

2.3.2. *Manipulation with Knowledge. Production Systems and Inference Mechanisms*

For the system to be able to solve a given problem, it must be able, by means of general data of the given field (declarative knowledge) and certain rules (procedural knowledge) to find the best solution. The control of this reasoning process depends strongly on the kind of field solved and on the knowledge representation. Production systems proved to provide a very convenient formalism, affording a unified conceptual and methodological basis. A production system is composed of the main data structure (global database), a set of production rules, and a control system[32 - 34].

The global database is a data structure (used as the buffer) in which knowledge of the states of the world described by means of the production system is stored in a specific manner. Fields, lists, or sets of arranged n-tuples can constitute the internal structure of the global database.

The production rule can involve logic elements which are similar to the conditional expressions in the majority of programming languages, which can be formulated in the form: "If a situation has occurred, then derive consequences (their solution)".

The control system works in a simple cycle:

A) Checks whether actual data satisfy the condition for termination (by calling a routine testing its validity on data); if this is the case, the function of the production system ends (the world described by this system arrives at the target state) and the control system issues the sequence of rules that have consecutively been applied on the way to the target database.

Else,

B) a rule whose conditions is met in the actual database is selected; this step assumes that the control system has available a comparison routine (which knows how to

evaluate the conditions of the rules and decide whether they are satisfied in the database) as well as a routine which in some way selects one of several applicable rules. Finally it

C) applies the selected rule (calls a routine which implements the corresponding change in the database), records it along with the database which has emerged from the implementation, and returns to the first step of the cycle.

This way of function of the production system is called forward chaining or data-controlled procedure. The conditions of the rules are tested on given input data, some of the usable rules are applied, the database is then changed, another applicable rule is chosen, the database is additionally changed, and this is repeated until the database meeting the termination condition – the target database – emerges. The reverse approach, viz. backward chaining or the target-controlled procedure, is also applicable.

The strategy for selecting a suitable rule which will actually be applied can be irreversible, such as the steepest ascent method, or reversible. From among the reversible strategies, the backtracking strategy and the graph search strategy or the controlled graph search (heuristic) strategy[33] are worth mentioning.

A specific group is formed by the so-called decomposable production systems[7,35,36]. During their performance, these systems create a more general structure of the so-called AND/OR graph. This consists of a set of knots and a set of connectors. The interpretation of this structure with respect to decomposable production systems is as follows: Knots of the AND/OR graph correspond to particular data. The starting knot corresponds to input data. If these are decomposable, a connectors goes from the starting knot to the set of immediate successors, corresponding to input data components (simple data). Target knots correspond to simple data satisfying the termination condition. The connector corresponds to the application of the production rule. Its parent knot is evaluated with simple data to which the rule is applied, and the knots in which the connector ends correspond to simple data which emerged from the implementation of the rule and decomposition of the obtained compound data into the components.

Decomposable systems thus define an implicit AND/OR graph with a given starting knot and a set of target knots, and during their performance they explicitly create its subgraph – the search graph – by gradual expansion of selected knots. The problem solving consists in controlling the performance of the production system so that the search graph formed should include the so-called solving graph (from the starting knot to the set of target knots).

Problems solving in a production system can also be treated by using the first order theory (axioms and inference rules) or by resolution methods of proof. Various strategies such as the supporting set strategy or the unit strategy[37,38] can be used when implementing the resolution method.

2.3.3. *Problem Solving by Means of Expert, Knowledge-Based and Hybrid Systems*

Expert and knowledge-based systems dominate in the field of artificial intelligence applications[39]. Hybrid systems, which are combinations of expert systems and neural networks, also show promise.

An expert system is a program system for automatic solution of a class of real problems which are so complex that this is usually done by a person who is specialist in the field in question[40]. The expert system emulates (imitates) the reasoning of the expert – specialist in the narrow problem field when solving tasks which to the user are important but too complex to solve. The aim of expert systems is to support (alleviate) the decision-making of personnel whose professional knowledge and experience standard is relatively lower.

Furthermore, an expert system is (viz. refs[41,42]):

1) Heuristic, i.e. the reasoning is based on formal, theoretically founded knowledge as well as on intuitive informal knowledge

2) Transparent, i.e. provides explanation of its reasoning procedure and answers questions concerning its knowledge

3) Flexible, because it can work with spurious, noise-containing data and is able to gradually integrate new knowledge into its existing knowledge structure.

Expert systems, as intelligent knowledge-based systems, are a particular and very rich but not the single existing subset of knowledge-based systems[43]. A characteristic feature of both systems is a strict separation of knowledge (which is usually problem-dependent) and calculation control – reasoning (which is usually problem-independent). This principle governs the basic architecture of the resulting system. The basic structure of the expert or knowledge-based system includes essentially three components, viz. the knowledge base, the inference mechanism and the global database[44].

The knowledge base comprises a summary of structured general knowledge concerning the problem field, which is usually narrow. The aim of the knowledge base is to represent relations among the various conceptions of the field in question.

The global database is a temporary working database which is filled (instantialized) and updated during the inference (consultation), each concept having no more than one instance during the inference. The inference process is in a sense determined by the global database updating sequence, an explicit record of this sequence being usually a part of the global database. In the concluding stage the expert system provides a suitable representation of the final form of the global database.

Thus, the main functions of expert or knowledge-based systems include:

a) Knowledge acquisition, creation and modification of the knowledge base

b) Knowledge use – inference, consultation

c) Explanation of the knowledge itself (static) as well as of the results of the inference procedure (dynamic).

Expert systems also possess features which differentiate them from knowledge-based systems. While the aim of expert systems is to emulate the expert's reasoning, the aim of knowledge-based systems is to accumulate and use specific knowledge obtained from the expert or from another source (a book for instance) in the problem solving. The way of handling the knowledge of the "behaviour of the system" may be different from that used by the expert. Also, some knowledge-based systems may lack the explanatory function or facility for substantiating the conclusions they have arrived at.

2.3.3.1. Classification of Expert Systems

With respect to the knowledge representation, systems based on production rules are the most widespread among expert systems. Contemporary expert systems employ two techniques for handling the rules:

A) Rules are activated by forward chaining in the order of their logical linking

B) Rules are all activated simultaneously or in a mutual sequence which is not directly dependent on the link between the rules (because this does not function on computers equipped with a single processor).

The second group of expert systems, with respect to the representation of knowledge, comprises systems based on logical programming[45], in which knowledge is expressed in the form of formulas, and proofs of truthfulness of the system of such formulas are derived.

The third group is formed by expert systems which are based on frames (data structures comprising complex information on objects, on object classes, etc.).

With respect to the character of the problems solved, the existing expert systems can be classed in two groups, viz. diagnostic systems and planning systems.

The task of diagnostic expert systems is to perform efficient data interpretation with the aim to find which hypothesis from an a priori given finite set of target hypotheses corresponds best to the real data which concern a given finite case. Such systems of classification nature solve the problem by gradual evaluation of the individual target hypotheses within a firmly given internal (machine) model of the problem, which is entered by the expert, usually in the form of an inference network.

Planning expert systems usually solve problems where the goal and the starting state of the solution are known and the system uses data concerning the particular case to find a sequence of steps (operators) by means of which the goal can be reached. Planning systems are based on the principle of generating and testing permissible solutions. A substantial component of such expert systems is a generator of possible solutions which automatically combines the sequence of operators. Expert's knowledge and data on the particular case solved are used to substantially reduce the combinatorial explosion of the examined solutions proposed by the generator.

With respect to universality and closedness, expert systems can be classed into problem-oriented expert systems with a used representation of knowledge and a control

mechanism to solve problems within a certain problem field only, into shells (problem-independent, without a knowledge base), universal tools for creating shells, and ready (closed) applications of expert systems.

2.3.3.2. Tools for Creating Expert Systems

The principle of strict separation of knowledge and control of the calculation – reasoning in the expert system – also governs the nature and structure of the tool.

Software tools serving the development of expert and knowledge-based systems can be classed in six levels, viz. machine language, operating systems, higher-level programming language, environment, tool, and shell[46,47].

Although any algorithm can be programmed in any conventional programming language (Fortran, Algol, Pascal, C), the use of such languages for artificial intelligence purposes is rather low efficient and more or less inappropriate. The nature of artificial intelligence usually requires – in contrast to classical calculations – efficient handling of data in a symbolic form and with a rich internal structure (lists, chains, sets, trees), a mechanism for returning during state-space search, associative memory, and a mechanism for automatic inference. Language LISP[48], a tool for functional programming, was set up in the early 1960s; its dialects[49], viz. INTERLISP, ZETALISP and MACLISP followed. The first language to satisfy the above requirements was PLANNER[50], which, however, was never implemented to the full extent. At present, much attention is given to the language PROLOG[51 - 53], which is based on formulas of the language of 1st order predicate logic. While LISP is most popular in the USA, PROLOG is most widespread in Europe[54,55]. Another well-known language is FORTH[56]. The object-oriented languages SMALLTALK[57] and AIRELLE[58 - 61] also have a high standard. Versions of object-oriented Pascal and C, as well as C++, are available from several vendors.

At a somewhat higher level is the language OPS 5 (ref.[62]), which was set up at Carnegie–Mellon University with the initial aim to examine in more detail mechanisms of human reasoning and memory based on production rules. Facts are represented in the system as objects with attributes and values. The inference mechanism encompasses three cyclically repeating algorithms: rule matching, rule choice and rule implementation. Its strategy consists in forward chaining enabling also back chaining simulation. OPS 5 stores objects in the so-called working memory and rules, in the so-called production memory. The advantage of OPS 5 consists in an efficiency of calculation, which is achieved owing to the so-called Rete-algorithm using a special organization of the production memory. At the beginning, rules are compiled and stored in the form of an oriented graph which enables efficient matching to the contents of the working memory. The initial version was implemented on LISP-machines, VAX-11 computers and IBM PC's. Updated versions are augmented with graphic communication and Windows.

In the late 1970's and early 1980's, universal modular systems, shells, began to be developed for the generation of expert and knowledge-based systems. In such shells, or in the programming environment or hybrid tools, only some components (e.g. the inference strategy or representation scheme) are prefabricated and fixed but they are not firmly linked into the functional unit. In this case the process of creation consists in the creation or modification of the remaining components (usually at the programming language level) and linking the fixed and completed parts into a functioning unit. In a sense these universal systems can be regarded as special, very high level programming languages. Among the best-known universal systems are KEE[47], ART[47], Knowledge Craft[47], R1[63], ROSIE[64], M1[65], POPLOG[66,67], KAS[68], EXPERT[69], LOOPS[41], VP-EXPERT[70], NeXPERT[70], HEARSAY-III[71], ES/P Advisor[47], TEIRESIAS[72], EMYCIN[73], EXSYS[47,70] and FEL-EXPERT[74].

KEE, ART and Knowledge Craft are large hybrid tools whose authors did not center on some particular application field or strategy. Their aim was to create a powerful universal tool for the creation of narrower expert systems, shells for instance. The tools, however, require special hardware, viz. LISP-machines or VAX computers.

Some shells are derived from expert systems such as KAS or FEL-EXPERT from the expert system PROSPECTOR[30,75 - 77], EXPERT from the expert system CASNET[78,79], HEARSAY-III from the expert system HEARSAY-II[80] (a system for understanding speech) and EMYCIN from the expert system MYCIN[81,82].

EXSYS[70] is an inexpensive and simple shell. It is used on XT/AT PC's or on VAX/VMS minicomputers. It uses a frame and rule scheme of knowledge representation with a facility for organizing knowledge into hierarchies. The implementation language is LISP or C.

FEL-EXPERT system, which has been developed at the Faculty of Electrical Engineering, Czech Technical University in Prague, is now available in several versions at several types of minicomputers and personal computers. This is a shell with the highest number of installations in Czechoslovakia.

More detailed overviews of existing state-of-the-art expert systems, shells and hardware can be found in publications by Hayes-Roth and coworkers[83], Gevarter[84], Waterman[85], Harmon and coworkers[86], Holme and Pierce[87], and Wade and coworkers[88].

2.3.4. *A New Approach to Problem Solving by Means of Neural Networks*

Today's mass-scale produced computers starting from the Notebook category to vector and parallel supercomputers are set up following the von Neumann architecture. Any computer comprises a central computer unit, memory (which contains instructions for operation management and data files) and busses through which there pass data from input or output devices such as the monitor, keyboard or printer. A von Neumann-type computer works sequentially and follows a precisely defined program. Furthermore, it has a direct access to the memory and exhibits a high arithmetic accuracy and a high

reliability. This kind of computer performs in an outstanding way in solving well-defined problems by using algorithms based on procedural logic, but it is not so good in simulating human cognitive abilities such as pattern recognition, speech understanding, decision making, learning, inference, induction, association, and handling incomplete (uncertain or noise-containing) information.

A solution to these artificial intelligence problems may be in computers with the neural architecture, which attempt to simulate the way human brain functions. The hardware solution is at its infant stage but there exist a number of simulation programs which can be implemented on von Neumann-type computers. This approach is highly acknowledged because it enables one to get into the basic functions of neural networks. However, the principal asset of neural networks, viz. parallel architecture, is lacking there.

Neural networks are a relatively young part of artificial intelligence but their study opens up never-thought-of opportunities in their development. The history of these networks is even longer than that of electronic computers. The British scientist Turing[89] examined theoretically the possibility of imitating the function of the human brain as early as 1936. In 1943, McCulloch and Pitts[90] set up an electronic model of the brain cell – the neuron, and in the sixties, Rosenblatt developed the so-called single-layer perceptron[91] (abstract system), which is a simple network of neurons capable of responding to external models and identifying features in which these resemble one another. In 1968, Minsky and Papert[92] demonstrated that there exist cases which the perceptron fails to discriminate. In 1982, Hopfield developed an interesting model which was later named after him[93]. In 1986, Rumelhart, Hinton and Williams[94] analy-

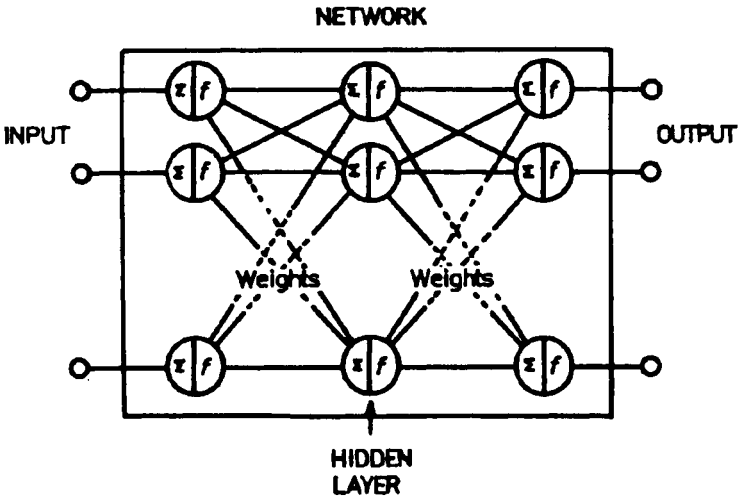
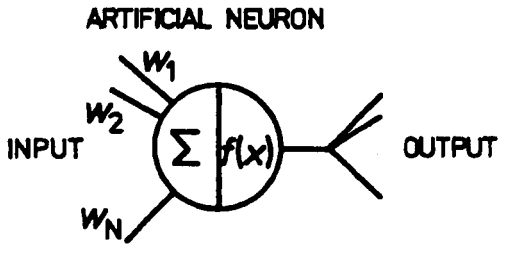


FIG. 1
Neural network

zed also multilayer perceptrons. The structure of a neural network as a computer or algorithm system is shown in Fig. 1, demonstrating the input and output layers between which one or more so-called hidden layers occur. On these layers there proceed the sorting operations; it is impossible, from the result to infer back in a detailed manner what took place on the layers. It is a limitation of this neural network that the signals can only be transmitted forward to the next layer.

Neural networks are in principle not programmable but they learn by training and they behave accordingly. Their learning ability is so organized that a known pattern is entered at the input and the corresponding results is formed at the output. The weights w are varied by means of a suitable algorithm until the network forms a result which approaches the input pattern. The ability to learn is an important property of the human brain, and it can be derived from some changes at the boundary of the junction between neurons (synapse). Technically, this process is accomplished in neural networks so that a weight w is attributed to the junction between neurons. During the learning, the weights are varied according to the network model and according to a given algorithm in time. When the learning process is terminated, the weights are fixed and are not changed any more. Another important property of the human brain is the associative memory. The so-called Hopfield networks[95 - 97] function as binary associative memories with symmetrical weights and feedbacks of each neuron to all the remaining neurons. The networks are one-way and are most frequently applicable to problems which are binary in their nature. The weights are determined by training and calculated



$$\text{OUTPUT} = f\left(\sum_{i=1}^{N} w_i \; x \; \text{INPUT}_i\right)$$

$$\text{DISCRETE HOPFIELD: } f(x) = \begin{cases} 1 & \text{if } \geq 0 \\ 0 & \text{if } < 0 \end{cases}$$

$$\text{BACK PROPAGATION OR CONTINUOUS HOPFIELD: } f(x) = \frac{1}{1+e^{-x}}$$

FIG. 2
Artificial neuron

from the value of the input vector of the input pattern (Fig. 2). The best-known learning algorithm for these neural networks is the back propagation algorithm.

The next kind of neural network after Kohonen[96] is referred to as the feature map. Here the elements from the output layer are suitably mutually arranged into a two-dimensional lattice. An algorithm attributing neighbouring output elements to patterns according to their similarity is used. Self-organization takes place during the learning process, and Kohonen's network does not need any help. It is used in solving complex pattern recognition problems. Its drawback is in the fact that the training is rather time-consuming.

2.4. Understanding Natural Language and Communication with Machine in Natural Language

Natural language is the most perfect tool in knowledge representation. Great attention is therefore devoted to understanding it. The basic goal is to set up a program that would "understand" sentences in natural languages, entered, for instance, from a terminal.

The basis in solving this problem is the natural process of human reasoning, which is transferred, by using artificial intelligence tools, as a process of knowledge representation (it is a unification of the language of representation of meaning and language of representation of sense) with the possibility of combining language structures (machine reasoning) and also of interpreting them in the inference engine mechanism. The language structure understanding proceeds in a cyclic process and some authors refer to this as the hermenautic cycle[98] which consists of tentative understanding, the understanding proper, and revised understanding. There exist several systems for understanding language structures (text and speech) which are based on text models – the general model of the standard theory of understanding[99], the model of approaches to text decomposition[99], Charniak's model of text understanding[100], and the model of the theory of directional text recognition[101].

3. APPLICATION OF ARTIFICIAL INTELLIGENCE IN CHEMISTRY

The various artificial intelligence methods and algorithms can also be used to solve practical problems in chemistry. The development of applications of artificial intelligence in chemistry proceeds virtually in parallel to the development of chemometrics, which is based on statistics and numerical calculations.

The most important artificial intelligence methods employed in chemistry include:

1) The use of chemical knowledge (expert systems) and neural networks in

a) structure elucidation,

b) synthesis planning,

c) elucidation of quantitative structure-biological activity relations (QSAR) and quantitative structure-chromatographic retention relations (QSRR),
d) experiment planning.
2) Intelligent instrumentation and robotics.
3) Chemical databases and retrieval systems.
4) Intelligent tutorial methods.
5) Processing of sensoric data and natural languages.

3.1. STRUCTURE ELUCIDATION

One of the first tasks of artificial intelligence in chemistry was structure elucidation based on spectral data. Elucidation of the structures of compounds which have been obtained synthetically or by isolation from natural material is among the most important problems of contemporary chemistry, and spectral methods play a prominent role in this problem solving. Such spectral methods include, in addition to X-ray diffractometry which directly determines the structure, also mass spectrometry (MS), electronic spectroscopy (in the ultraviolet and visible regions), vibrational spectroscopy (infrared and Raman spectroscopy) and nuclear magnetic resonance.

A rational application of molecular spectroscopy to structure elucidation is conditional on the ability to obtain spectral data by the purely theoretical way with an accuracy which is sufficient in comparison with the accuracy attained experimentally. The procedure in the structure elucidation is such that the spectrum of the unknown compound is taken, and a theoretical calculation of the spectrum is performed for the presumed (hypothetical) structure. According to the better or poorer agreement between the theoretical and experimental spectra the hypothesis is then adopted or rejected.

Automatic structure elucidation methods can be classed in three categories: direct library search (direct database techniques)[19,20,102 - 105], indirect library search (pattern recognition)[21,102,106,107], and expert systems.

In the direct library search, the spectrum of the unknown is compared with a collection of reference data by using search algorithms. The search is largely sequential or in hierarchic trees[105].

In the indirect library search, the pattern recognition methods are based on the principle that organic compounds can be classed, with respect to their spectra of some kind, into groups (classes) which have a certain substructure.

Expert systems attempt to encode the logic of visual spectra interpretation, as done by the expert, into automated computer routines.

3.1.1. *Overall Strategy of Structure Elucidation by Means of Expert Systems*

The overall strategy of computerized elucidation of chemical structures comprises four steps[4,108]:

1) INTERPRETATION of the spectrum to obtain the probable substructures
2) GENERATION of proposed structures containing the probable substructures
3) PREDICTION of spectra for the proposed structures
4) RANKING of the proposed structures by comparing the predicted and observed spectra.

Spectrum interpretation to obtain the probable structures is done by correlation and consistency testing. By correlation, the possible structural fragments (substructures) are derived by using some spectral method or combination: MS, ^{13}C NMR, ^{1}H NMR, IR, Raman and UV spectra (this arrangement corresponds to the decreasing information entropy of the spectrum). Correlation provides automated interpretation of the spectrum so that both the precise and the feasible structural fragments which are related to the form of the molecule should be disclosed. This interpretation is done by means of identification algorithms, viz. the AND/OR tree identification algorithm, the network algorithm or the matrix algorithm.

The AND/OR tree identification algorithm is the most popular, presumably owing to its clear internal structure and the rather short computing time necessary to identify the structure fragments.

Network algorithms resemble networks with extended internal logic connections. The internal structure of such algorithms is extremely intricate.

In matrix algorithms, the correlation parameters (group functions, chemical shifts) are stored as fields in computer programs. This kind of identification algorithm is best suited to computer processing because their inner structure is quite clear and the computer time necessary to identify the fragments is short. The efficiency of the three kinds of identification algorithms is difficult to compare due to their "diffuse" nature and, to some extent, because they also mirror the algorithm designer's skills and experience.

Spectral parameters which are taken into account during the empirical interpretation of the spectrum are closely related to the spectroscopic technique of choice. At least two parameters are used to describe an absorption band: its position and intensity. Other spectral band parameters are also frequently employed[109,110]: the half band width, the interaction constant, etc. A decisive factor is the band intensity because it depends appreciably, e.g. in IR spectroscopy, on the sample preparation procedure and on the sample concentration. This problem can be bypassed by the so-called self-normalization[111], which consists in standardization of each band intensity (if this obeys Beer's law) with respect to the most intense band to which a value of 100% is attributed. Of importance are the quality of available spectra-structure correlations as well as the character of the spectral methods. Mass spectra are considered to be the most informative.

The consistency test usually employs cross-correlations of results obtained by means of different spectroscopic techniques. The results of correlation can confirm, comple-

ment, or rule out any substructure. This depends on the identification power of the method in question (e.g. the CH_3–C=O substructure suggested based on ^1H NMR spectroscopy must be rejected if no carbonyl group signal is present in the IR spectrum). Consistency tests can use, in addition to cross-correlations, various programming conditions based on a built-in chemical theory, chemical stability, etc. The aim of all of them is to reasonably reduce the recognized substructures assuming that none of the chemical groups present in the molecule under study is ruled out.

Structural assembly involves combination of substructures into a meaningful total structure (tentative or candidate structure). Structural assembly relies on the creation of all the possible structures compatible with the detected substructures, with empirical rules, and with the stored chemical bonding and chemical stability theory. This can be organized in various ways. The most sophisticated structure-generating program is GENOA[108,112], which can be combined with STEREO program in generating stereoisomers of the proposed structures (DENDRAL project[113]). GENOA solves the structure problem by using a constructive substructure search algorithm.

In spectrum prediction, selected spectral features or the entire spectrum are predicted for the candidate structure. In the final step the predicted and observed spectra are compared. If they are identical, the candidate structure is the true structure. Spectrum prediction and spectra comparison actually constitute another consistency test.

3.1.2. *DENDRAL Project and Other Expert Systems*

Automated, computer-assisted structure elucidation was performed at Stanford University in 1969 in relation to the interpretation of mass spectra. Since then the so-called DENDRAL project[114,115] has been developing and improving constantly. The advanced versions of this expert system[116] employ other data as well, e.g. ^1H or ^{13}C NMR data[117 - 122].

The DENDRAL system knowledge base relies on molecular chemistry (atoms, their bonds and stability), on mathematical chemistry (graph theory – constructive enumeration of molecular graphs[114,123 - 125]), and on spectroscopy (rules of spectroscopic methods). It also involves contextual properties of sample (its origin, way of separation) and rules for the evaluation of the closeness of relation between the observed and predicted values.

The inference mechanism of the DENDRAL system works encompasses three major stages (Fig. 3):

1) Planning – the substructure conditions are derived from available spectral and chemical data

2) Generation – a combinatorial algorithm generates candidate structures which are compatible with the substructure conditions

3) Testing – the candidate structures are revised by using heuristic criteria, and the structure which is best consistent with the data is selected.

DENDRAL was initially designed for mass spectra interpretation. Of all spectral techniques, mass spectrometry is least affected by stereochemical factors, and thus it is best applicable to structure analysis. However, the simultaneous development of STEREO program provided tools for representation and manipulation of configuration chemistry and offered the opportunity to define substructure characteristics involving

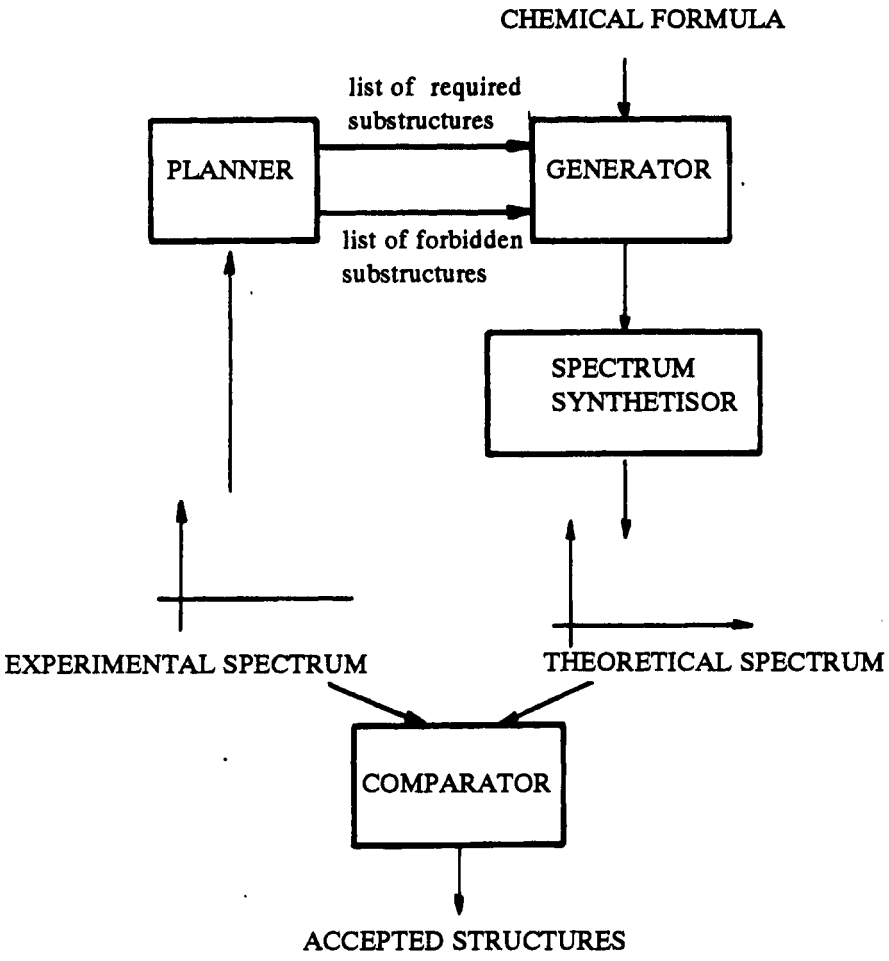


FIG. 3
Function chart of DENDRAL in structure elucidation

stereochemistry in ^{13}C and ^{1}H NMR spectra. To date, DENDRAL is the most widely used expert system with the help of which, thousands of chemical structure analyses have been performed[112 - 120,123 - 130]. The possibility of continual extension of its knowledge base is among the assets of this system.

In 1976, Buchanan and coworkers[116,131] set up the Meta-DENDRAL program which serves automated acquisition of knowledge from structure chemistry and its inclusion in the DENDRAL knowledge base. Using a large number of entered spectrogram-structure pairs, Meta-DENDRAL sets up, for a given class of molecules, rules concerning relations between types of spectrograms and types of structures. Again, the system works via the planning-generation-testing sequence. At present, Meta-DENDRAL enables automated creation of rules for mass spectra[131,132] and for ^{13}C NMR spectra[133].

The success of the DENDRAL project undoubtedly stimulated other research centres which resulted in additional expert systems, whose overview is presented in Table I. A brief description of interpretation of spectra by means of these systems can be found in various reviews[35,102,134,135].

Woodruff and coworkers[149 - 162] developed PAIRS system for the interpretation of spectra. PAIRS uses interpretation rules formulated in the special CONCISE language, which is a form of the binary decision tree. The rules are then compiled by a FORTRAN program to be used in the given interpretation program. At our laboratory we are dealing with a modification of a PAIRS system version designed for the interpretation of IR spectra of biologically active substances[199].

In conclusion, the overall structure elucidation strategy of the various expert systems given in Table I can vary somewhat; the largest differences are in the kind of the interpretation program, in the way of reducing the solution and also in the interaction of the user with the system. CRYSALIS[191], and expert system designed for the elucidation of the tertiary structure of proteins based on their X-ray diffraction crystallographic map, is an exception: graphic representation of the system was omitted in CRYSALIS because of the high complexity of protein molecules. CRYSALIS provides hypotheses of various kinds – atomic (atom positions), apical (position, composition and tilting of peptides), skeletal (secondary structure kind) and segmentary (secondary structure linking). The knowledge base includes the primary structures of proteins, X-ray diffraction maps, and chemical rules.

Heuristic rules are hierarchized into three levels[7]:

1) Strategic rules, determining which of a multitude of tasks is to be performed by the system

2) Tactical rules, determining how to perform the given task in the optimum way

3) Source rules, which actually comprise the expert knowledge of X-ray diffraction analysis.

The inference mechanism uses the agenda control. A bank of hypotheses which have different levels and weights is maintained continuously. The system responds to changes occurring in the agenda and does not proceed in an entirely targeted manner. The system stops running the moment a hypothesis with the desired degree of certainty is found.

3.1.3. *Structure Elucidation by Means of Neural Networks*

Application of neural networks to the interpretation of spectra offers new views upon the computerized chemical structure elucidation. Thomsen and Meyer[200] trained neural networks to the resolution of ^{1}H NMR spectra of alditols by a back-propagation algo-

TABLE I
Expert systems and kinds of spectral information handled by them

System	MS	IR	^{13}C NMR	^{1}H NMR	UV	Raman	Program language	References
DENDRAL	x		x	x			LISP	112 – 120, 123 – 130
Meta-DENDRAL	x		x	x			LISP	116, 131 – 133
CHEMICS		x	x	x			FORTRAN	136 – 142
CASE		x	x	x			FORTRAN	143 – 148
PAIRS		x					FORTRAN PROLOG	149 – 162
STREC	x	x	x	x	x	x	FORTRAN	163 – 168
SEAC		x		x	x		FORTRAN	4, 169, 170
ASSIGNER		x	x	x			FORTRAN	171 – 173
EXSPEC	x	x	x	x	x		PROLOG	174, 175
EXPERTISE		x						176
SICLOB		x	x	x	x		AIRELLE	59
STIRS	x						LISP	177
Curry's[a]	x	x					LISP	178, 179
Damo's	x							180
CRISE		x				x		181
FOSSIL		x						182
SCANSPEC		x	x				FORTRAN	183
IRIS		x						184
CTREZE			x					185 – 188
SISTEMAT			x					189, 190
CRYSALIS[b]							PROLOG	191
Others								192 – 199

[a] Interpretation of GC/IR/MS data. [b] Interpretation of X-ray data.

rithm. Kvasnička[201,202] used a three-layer neural network represented by oriented graphs in the classification of chemical shifts in the ^{13}C NMR spectra of acyclic alkanes. Extensive studies in spectroscopy have been undertaken by Curry and Rumelhart[203] and by Munk and coworkers[204 - 206], who employed more than 32 000 MS and about 6 700 IR spectra. The authors attempted to predict all the possible structure fragments relevant to the spectra. Bos and Weber[207] found that neural networks can be trained for fluorescence spectroscopy better by means of a back-propagation algorithm than by means of a genetic algorithm. Tušar and Zupan[208] applied Hopfield's[93,208] and Hamming's[209] networks to the interpretation of IR spectra. The analysis of the DNA structure is another field of application of neural networks. Examination of the sequence of nucleotide bases in DNA by means of neural networks, with the determination of regions which are coded for proteins, is done at the National Institute in Los Alamos[210].

More detailed reviews of applications of neural networks to structure elucidation can be found in papers and monographs by Zupan and Gasteiger[96,211], Wythoff and coworkers[212], and Otto and Hörchner[213].

3.1.4. *Combination of Artificial Intelligence Methods with Theoretical Quantum Chemical Calculations*

In addition to artificial intelligence methods, quantum chemical calculations can also be used to establish some spectral properties of candidate structures[214 - 216]. The applicability of semiempirical methods (PPP, CNDO/2, INDO) and nonempirical methods (*ab initio*) is dubious or at least limited. The quantum chemical calculations of theoretical electronic spectra must include the configuration interaction (CI) approach[217]. In this, the so-called configuration interaction wave function is considered; this is a linear combination of Slater determinants corresponding to different (excited) electronic configurations:

$$\Psi = \sum_{k}^{M} A_k \Psi_k \qquad (1)$$

where M is the number of configurations. Electronic configurations Ψ_k (Slater determinants) represent the various ways of electron distribution over all orbitals. The configuration Ψ_0 corresponds to the ground state configuration, the other Ψ's are the configurations of excited states. The excitation energies can be found from the differences of the total energies of the excited and ground states[218]. In the Pariser–Parr limited configuration interaction method[219], only the determinants of monoexcited configurations are included in the wave function series. A modified CNDO/S method where the excited states are calculated by means of CI has been developed by Del Bene

and Jaffé[220]; this method is now most widely applied to the calculation of excitation energies.

Vibrational (IR) spectra are calculated theoretically by using *ab initio*[215] and semiempirical (CNDO/2, MINDO/3) methods. The problems solving then consists in obtaining a dependence of the total energy and dipole moments on the displacements from the equilibrium geometry along one, two or more coordinates. The dependences so obtained can be applied to any level of the vibrational spectra theory. The harmonic approximation is mostly used for polyatomic systems; in particular, the matrix ***GF*** method is employed where the harmonic vibrational wavenumbers are obtained by solving the equation[221]

$$| \boldsymbol{GF} - \boldsymbol{I}\lambda | = 0 , \qquad (2)$$

where $\boldsymbol{F}$ is the force constant matrix, $\boldsymbol{G}$ is the kinematic matrix and $\boldsymbol{I}$ is the unit matrix.

The task of the *ab initio* calculation then only consists in determining the matrix of quadratic force constants $\boldsymbol{F}$ and/or the equilibrium geometry requisite to calculate the elements of $\boldsymbol{G}$ matrix. An *ab initio* method for the calculation of force constants has been developed by Pulay[222,223] (TEXAS program). The *ab initio* calculation of the quadratic and cubic constants has advanced recently, and this might contribute to the solution of the highly intricate problem of anharmonicities of polyatomic molecules.

The NMR spectral parameters are also calculated by semiempirical methods – CNDO/2, MINDO/2, MINDO/3 – as well as by *ab initio* methods. In the theory of interactions between nuclear spins according to Ramsey[224], three basic types of interaction are considered. The most important of them is the Fermi contact interaction between the spins of the electron and the spins of the nucleus. This constant is calculated by the second order perturbation theory, by the incomplete perturbation theory and by the maximum overlap method[225]. A method for the calculation of nuclear interaction constants by the CNDO and INDO methods is described in the monograph by Pople and Beveridge[226]. The monograph[227] contains an extensive chapter by Segal on the application of approximate methods to the calculation of NMR spectral parameters. The BBC technique or the Lindeman–Adams method[228] can be employed in the NMR spectra prediction. Quantum chemical methods can also be applied to the interpretation of the chemical shifts[229,230].

Spectral data interpretation by means of artificial intelligence methods usually requires the processing of a vast quantity of generated candidate structures. This might be simplified considerably by suitably combining the artificial intelligence algorithms, which are based on the principle of structure assignment to spectral characteristics, and quantum chemical methods for the calculation of spectral characteristics of candidate structures. Quantum chemical methods can also find application when dealing with the effect of the remainder of the molecule, or the substructural environment, on the candi-

date structure fragment under examination (e.g., in NMR spectra, the atom studied is affected by the substructural environment as far as the fourth bond). Quantum chemical methods also enable the effect of medium (matrix) on various spectral characteristics to be modelled[231 - 238].

3.1.5. *Elucidation of Quantitative Structure–Biological Activity Relations (QSAR) and Quantitative Structure–Chromatographic Retention Relations (QSRR)*

Among the most difficult tasks of QSAR or QSRR studies is to find suitable molecular or submolecular characteristics of substances which determine their biological activity[239 - 242] or retention behaviour[241,243 - 245]. The application of artificial intelligence to this problem solving was the concern of Klopman[246], who developed a program which automatically identifies, tabulates and statistically evaluates substructures relevant to the given biological activity. Furthermore, Klopman and Henderson[247] set up and expert system for QSAR studies based on graph theory. Smith and Burr[248] developed the expert system CRIPES which is well suited to the prediction of retention data in reverse-phase high performance liquid chromatography, whereas Milne[249] developed the expert system CATHIE for the interpretation of GLC retention data.

Neural networks have also been used to study the structure–biological activity relation[250,251]. It turns out that neural networks[252], like rule-based expert systems[253], have a predictive ability which is better than that of the conventional methods for examining the structure–biological activity or structure–chromatographic retention relation (the multiparameter linear regression[254] or discriminant analysis[255]).

3.2. Synthesis Planning

Although organic synthesis, as one of the oldest fields of chemical sciences, was highly successful in complex syntheses of natural substances, it was not until the early seventies that, owing to the research efforts of E. J. Corey, the basic principles of systematic reasoning and retrosynthesis analysis began to be used in the planning of organic syntheses. Over the past twenty years, synthesis planning developed from purely inductive approaches to a highly sophisticated branch of science with various strategies of general synthesis problem solving. Equally important is the topical requirement of elaboration of a formal methodological approach which, in conjunction with the quantum chemical approach combined with statistical thermodynamics methods, would reflect the wealth of existing chemical knowledge in a unified and consistent manner. This formally methodological approach is based on tools of non-numerical mathematics (graph theory, topology, formal languages, etc.) and enables its notion and term apparatus to be rationalized, unified and formalized. The first major success of this mathematical chemistry was the solution of enumeration of chemical structures[125,256].

Enumeration of molecular graphs has found important applications in structure elucidation and algorithmization of generation of alternative structures which is based on data obtained by spectral methods (DENDRAL project).

The graph-theoretical approach[257 - 264] appears to be a suitable mathematical tool also in synthesis planning. This approach not only enables formalization of the description of molecules but also provides an illustrative formalism for the description of chemical reactions. Molecules are represented by a molecular graph. This is an unoriented pseudograph with multiple edges and loops, where the vertices are evaluated with alphanumeric strings containing letters (e.g. symbols of atoms) and numerals taken from a dictionary of symbols. Chemical reactions are expressed by means of reaction graphs involving the formation and decay of bonds (in ionic and radical reactions) and lone electron shifts during the chemical reactions.

However, the problem of chemical reactivity, which in the model means seeking for reasonable chemical transformations for the given molecular system, remains open. Within the formal method this problem is approached in various ways. Two major approaches include

– first generation algorithms

– second generation algorithms.

In first generation algorithms there is given a dictionary of reactions – transformations which accompany a segment of the molecule – synthon (subgraph of the molecular graph[265]) to the resulting product subgraph. The effect of environment for the given transformation is evaluated ("marked") by using heuristic rules. The molecular graph is analyzed with respect to the occurrence or nonoccurrence of a certain subgraph: if the subgraph exists, it is replaced with another subgraph by using a substitution rule, whereby the product molecular graph is obtained. This approach is feasible in the forward, synthetic direction, as well as in the backward, retrosynthetic direction. It is typical of this kind of algorithms that requirements put on the knowledge of the chemical reactivity are minimal and only occur within the heuristic rules marking the transformation with respect to the effect of environment.

When using second generation algorithms, the molecules are analyzed systematically with respect to the occurrence of so-called strategic bonds (double and triple bonds and bonds involving hetero atoms). Furthermore, the kinds of chemical reactions which are feasible at these bonds are regarded. Unlike first generation algorithms which handle immediately generated reactions, second generation algorithms use a more general viewpoint based on so-called type reaction mechanisms. The synthons of the molecule under study are not defined in advance; instead, they are generated based on potentially applicable reaction mechanisms. Reduction of the high number of chemical transformations derived[266,267] is achieved by means of an efficient estimate of reactivity of the molecule, viz. by using heuristic rules which correlate the substructures with the

reaction mechanism type. Hence, the choice of reactions can be limited by using tools obtained by artificial intelligence methods.

Algorithms of both generations have been implemented on computers and first and second generation programs have been so obtained.

3.2.1. *Computerized Synthesis Solving*

First generation programs have been used in research and in the industry since the late sixties (LHASA, used by Du Pont manufacturers, and SECS, used by Merck Sharp & Dohme manufacturers). Logically oriented second generation programs, formed based on mathematical models of logic structure (e.g. CAMEO, EROS), have been created in parallel. An overview of programs of the two generations is presented in Table II.

Program LHASA[268 - 270] was preceded by program OCSS[271], which was developed at Harvard University under the leadership of E. J. Corey. The major part of the system has been set up in FORTRAN whereas information concerning organic reaction is in a special language CHMTRN. LHASA system, and also SECS[272 - 274] and CASP[268] systems, employ large organic reaction databases and the CAS Online search system. The aim of the systems is to find the "optimum" synthesis route to the target molecule by retrosynthesis analysis. This is retrosynthesis analysis from the target molecule structure to the starting materials. Another program, CAMEO[275 - 277] developed by W. L. Jörgensen at Purdue University, works in the synthesis direction. This reaction-simu-

TABLE II
Programs for computerized synthesis solving

Program	References	Program	References
LHASA	268 – 270, 284	IGOR	296
OCSS	271	IGOR 2	297, 298
SECS	272 – 274	RAIN	299, 300
CASP	268	PASCOP	301, 302
CAMEO	275 – 277, 286	LILITH	303
EROS	278 – 280	KASP	304
CICLOPS	281	MASSO	305
SYNGEN	285	MICROMASSO	306
AHMOS	287, 288	SOS	307
ASSOR	289	AIPHOS	308
SYNCHEM	290, 291	PSYCHO	309
SYNSUP-MB	292, 293	MAROCO	60
CARSA	242	MAPOS	310
RETROSYN	294, 295	DARC-SYNOPSIS	311, 312

lating program is a tool which conveniently supplements the above-mentioned LHASA, SECS and CASP programs.

A next program, EROS[278 - 280], can serve synthesis, analysis as well as reaction simulation. An example of a complex use of LHASA, SECS, CASP, EROS and CAMEO programs is the work of Nevalainen[281], who studied the synthesis of antifugal 5,6-dihydro-1,4-oxathiines and 5,6-dihydro-1,4-dithiines. Brief characteristics of LHASA, SECS, EROS, CICLOPS[282] and other programs can be found in refs[60,283].

While LHASA[284] and SECS are knowledge-based programs, EROS, Hendrickson's SYNGEN[285], CAMEO[286], AHMOS[287,288] and ASSOR[289] are logically oriented programs. Another system, SYNCHEM by Gelernter and coworkers[290,291], is a large knowledge-based program which solves problems heuristically by using inductive and deductive machine learning programs (Isolde and Tristan). The approach to proposing synthesis procedures by means of the noninteractive program SYNSUP-MB by Bersohn and coworkers[292,293] is also interesting.

At present there exist new logically oriented models which are used in conjunction with quantum chemical calculations, whose algorithms are based on potential energy hypersurfaces. Very likely, these models will form a basis for third generation programs[313,314].

3.2.2. *Synthesis Planning by Using Neural Networks*

Neural networks can serve as a supplement to expert systems, particularly where rules describing chemical reactivity are not easy to formulate.

Luce and Govind[315] developed a hybrid system which enables planning of syntheses of new molecules with retrosynthesis analysis. Elrod and coworkers[316] published an interesting paper concerning chemical reactivity in relation to the distribution of *meta* products of nitration of a series of monosubstituted benzenes. Input information into the neural networks was encoded in the form of the ***BE*** matrix by Dugundji and Ugi[317], which is an analogue of adjacency matrices in graph theory. Zou and coworkers[259] attempted to set up a model for the reactions of nitration of aromatics by using graph-theoretical transformations. A similar model has been set up by the authors[318], who represented neural networks by means of acyclically oriented graphs. Using this model, the authors were able to predict the yields of *meta* products of nitration of a series of monosubstituted benzenes.

3.3. Experiment Planning

Experiment planning has been receiving interest only lately. In analytical chemistry, attention is paid to the planning of simple experimental procedures[35,319 - 321] such as the choice of conditions for mixture separations by HPLC[322]. Successful are the expert system SPINPRO implemented in LISP language and serving the optimization of

centrifugation[323], and the expert system implemented in PROLOG language which serves in the planning of HPLC separation of various steroids[324]. Varian Associates manufacturers developed ECAT system for solving various HPLC problems[325]. An overview of additional expert systems in chromatography can be found in refs[326 - 332].

Relevant to chemical technology are problems associated with the planning of chemical reactions in the direction of catalytic synthesis[333] of the desired products, as well as problems related to the development of algorithms for technical diagnostics[334].

For instance, SCOPE system[320] for automated synthesis has been applied to the depropanization of the product from the production of ethylene (with a simultaneous separation of hydrocarbon fractions lower than C_3 and also higher than C_4).

Expert systems applied in chemical engineering also attract interest. CONPHYDE[335] assists in the choice of a suitable method for liquid–vapour calculations. SCCES enables the calculation of the corrosion risk of stainless steel reactors during cracking, whereas FALCON is capable of online analysis of a large volume of data in the planning of chemical processes[35]. CAMBIO[336] has been developed for the simulation of bioprocesses. Neural networks have also been applied with success to chemical process control[337 - 342].

3.4. Intelligent Instrumentation and Robotics

Combination of artificial intelligence methods with process control and optimization methods[343 - 345] has been leading to the development of intelligence-controlled processes and intelligent instrumentation.

Intelligence-controlled processes have found application e.g. in electrochemistry[346] and in the control of chemical processes in general[347]. A rapid development of intelligent instrumentation can be expected in the nearest future particularly in analytical chemistry[348,349]. This technique exhibits features such as automatic error detection, calibration, temperature compensation, sample composition interpretation, etc. This, however, places demands on the development of new chemical sensors or biosensors. A block diagram of instrumentation with intelligent control[350] is shown in Fig. 4.

The use of artificial intelligence methods in intelligent chromatographic instrumentation has been discussed in refs[351,352].

An expert system which selectively acquires, optimizes and interprets large volumes of data in real time[119] has been developed for analysis by triple quadrupole mass spectrometry.

The combination of expert systems with robotics is another field of active research in artificial intelligence[353]. Robot systems in conjunction with expert systems are used, for instance, in chelatometric analysis[354] and in chromatography[355]. Data on various types of automated laboratory equipment and automated systems with robotic techniques can be found in papers and monographs by Walters[348,349], Dessy[1,2], Kramer and Fuchs[356], Barker[210,357], Newman[358] and Insenhour and coworkers[359,360]. For example, PASS

system[358] serves to optimize the synthesis and analysis of organic substances. Another automated system, APOCALYPSE[210], is designed for crystal growth for X-ray diffraction analysis. GASP system[210,361] is used to optimize sample preparation for analysis.

3.5. Chemical Databases and Retrieval Systems

The importance of databases (sophisticated information banks) and search systems is obvious also in chemistry. The use of artificial intelligence methods in chemical informatics for the analysis as well as classification of chemical reactions will also be potentially important in the building and updating of databases and search systems. Another field of application of artificial intelligence is in its implementation as a tool for work with files. The user thus can, through questions and answers, define more specifically his or her demands during the retrieval of information, e.g., concerning a particular reaction. Artificial intelligence-based programs can also play an important part in the protection of databases against computer viruses.

Chemical reaction databases can be classed as general, subject-oriented and selective bases[362]. The largest general database is CASREACT (Chemical Abstracts Service, Columbus, U.S.A.) which contains more than a million reactions taken from 70 000 documents. CASREACT can be employed in conjunction with the CAS ONLINE FILE. This is now the best information system over the world. Another large database is VINITI (Moscow), which is now available to Western users as well. The best known search systems include REACCS with seven databases, ORAC, OSAC, and SYNLIB[363]. DARC system[364] for searching chemical substructures deserves particular attention.

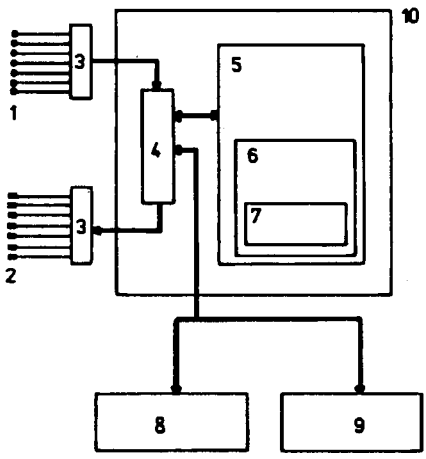


FIG. 4
Block diagram of intelligent instrumentation. 1 Sensors, 2 driving mechanism, 3 input and output interfaces, 4 microprocessor, 5 memory, 6 expert system shell, 7 knowledge base, 8 external expert, 9 peripherals (monitor, printer, mouse, etc.), 10 microcomputer

3.6. Intelligent Tutoring Systems

The increasing interest in applications of artificial intelligence methods in chemistry is also mirrored in the number of papers dealing with tutoring problems[349]. These include expert systems such as DENDRAL[116,365,366], LHASA[367], CAMM[368], and IDM[369], which can be used as an interactive teaching system for students. Other expert systems are designed for tutoring on PROLOG language[52,53] or for organizing work in analytical laboratories[349,370].

3.7. Processing of Sensoric Data and Natural Language

Although this problem may seem futurological, it has found application in chemistry. As early as 1979, an American company achieved A/D conversion of an IR spectrum by photographing the spectral image with a camera interfaced to a computer which accomplished the conversion by means of an artificial intelligence program.

At present there exist several commercial models of complex systems (Parallel Reader, Calera CDP 9000, Kurzweil k 5200) for text and image processing, which are equipped with dedicated microcomputers with a high capacity of the process memory[371]. These include high-performance sensors for bulk data processing making it possible to handle various letter types including hand-writing.

Other development trends are aimed at improving the resolving power of sensors. State-of-the-art computerized reading systems attain a rate of success of 90 to 96% in the identification of hand-written digits. A neural computer exhibiting a rate of success of 99.8 to 99.9% has been developed recently at Toshiba laboratories, while Hitachi manufacturers developed a system for speech recognition which is also based on the neural architecture; the device understands 3 000 words at a 95% rate of success.

Voice recognition, particular in continuous speech, starts to be employed commercially. For instance, Verbex Voice System manufacturers supply these systems for postal services in the U.S.A.

There exist now several computer programs for chemical text reading, such as SAM system developed at Yale University[372]. It must be stressed, however, that chemical text processing is a special problem because it involves three different kinds of information, viz. numerical, conceptual and structural, the complexity increasing in that order.

The listening and intelligently talking computer is still a vision and will remain so for some time. The major problem is the computer's ability to process speech, i.e. not only to decode the acoustic input but to understand the meaning of the sentences.

4. CONCLUSIONS

Artificial intelligence methods can find wide applications in chemistry but they require computer techniques starting from the conventional systems based on von Neumann architecture to systems with combined architecture, which is obtained by crossing with neural computers, to neural computers themselves.

Neural networks will be an important tool in the implementation of intelligent systems for the classification of chemical databases and prediction of properties of molecules. Advances in the research into artificial neural networks are expected in the nearest years to come; this can result in intelligent neural computers with millions of neurons (a new generation of computers) which will be able, after a training, to solve problems. Further progress should also concern the visual capability of robots, recognition of patterns including handwritten text, transformation of written text into speech, and speech processing, which also includes understanding the meaning of the spoken sentences.

REFERENCES

1. Dessy R.: Anal. Chem. *56*, 1200A (1984).
2. Dessy R.: Anal. Chem. *56*, 1313A (1984).
3. Francett B.: Computer World *51–52*, 20 (1991).
4. Hippe Z.: Anal. Chim. Acta *150*, 11 (1983).
5. Szewczyk A.: Informatyka *14*, 20 (1979).
6. Kelemen J.: Informační Systémy *4*, 305 (1980).
7. Gosman S. in: *Umělá inteligence a expertní systémy*. Obor. inf. stred., Praha 1990.
8. Lam T. F., Wilkins C. L., Brunner T. R., Soltzberg L.J., Kaberline S. L.: Anal. Chem. *48*, 1768 (1976).
9. Arkadjew A. G., Bravermann E. M. in: *Zeichenerkennung und Maschinelles Lernen*. Oldenburg Verlag, Munich 1966.
10. Jurs P. C., Kowalski B. R., Isenhour T. L.: Anal. Chem. *41*, 21 (1969).
11. Wold S.: Pattern Recognition *8*, 127 (1976).
12. Kaufman L., Rousseeuw P. J. in: *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York 1989.
13. Jurs P. C., Isenhour T. L. in: *Chemical Applications of Pattern Recognition*. Wiley, New York 1975.
14. Varmuza K. in: *Pattern Recognition in Chemistry*. Springer Verlag, Berlin 1980.
15. Bawden D., Ioffe I. I. in: *Application of Pattern Recognition to Catalytic Research*, Chemometrics Series, Vol. 14, p. 185. Research Studies Press, Letchworth 1988.
16. Patrick E. A., Fattu J. M. in: *Artificial Intelligence with Statistical Pattern Recognition*. Prentice Hall, Englewood Cliffs 1986.
17. Detaevernier M., Michotte Y., Buydens L., Derde M. P., De Smet M., Kaufman L., Musch G., Smeyers-Verbeke J., Thielemans A., Dryon L., Massart D. L.: J. Pharm. Biomed. Anal. *4*, 297 (1986).
18. Strouf O. in: *Chemical Pattern Recognition*. Wiley, New York 1986.
19. Brown S. D., Barker T. Q., Larivee R. J., Monfre S. L., Wilk H. R.: Anal. Chem. *60*, 252R (1988).
20. Brown S: D.: Anal. Chem. *62*, 84R (1990).
21. Musil S., Mocák J., Leško J.: Chem. Listy *86*, 14 (1992).
22. Winograd T.: Artif. Intell. *25*, 375 (1985).

23. Nilsson N. J. in: *Problem Solving Methods in Artificial Intelligence.* McGraw–Hill, New York 1971.
24. Zlatuška J.: Informačné Systémy *6*, 611 (1990).
25. Elcock E. W.: Computer *16*, 114 (1983).
26. Kelemen J. in: *Umelá inteligencia – programovacie prostriedky.* Comenius University, Bratislava 1985.
27. Bobrow D. G., Raphael B.: Comput. Surv. *6*, 153 (1974).
28. Hewitt C. in: *Description and Theoretical Analysis (Using Schemata) of Planner: A Language for Proving Theorema and Manipulating Models in a Robot.* MIT, Cambridge 1972.
29. Schubert L. K.: Artif. Intell. *7*, 163 (1976).
30. Ritchie G. D., Hanna F. K.: Inform. Technol. Res. Dev. *2*, 187 (1983).
31. Minsky M. in: *The Psychology of Computer Vision* (P. Winston, Ed.). McGraw–Hill, New York 1975.
32. Davis R., King J. in: *Machine Intelligence* (L. Elcoock and D. Michie, Eds). Ellis Horwood, Chichester 1977.
33. Nilsson N. J. in: *Principles of Artificial Intelligence.* Tioga, Palo Alto 1980.
34. Renc Z. in: *Úvod do umělé inteligence.* Charles University, Praha 1986.
35. Klaessens J., Kateman G.: Fresenius Z. Anal. Chem. *326*, 203 (1987).
36. Kleywegt G. J.: Lab. Microcomputer *6*, 74 (1987).
37. Štěpánek P. in: *Matematická logika.* Státní pedagogické nakladatelství, Praha 1982.
38. Chang C. L., Lee R. C. T. in: *Symbolic Logic and Mechanical Theorem Proving.* Academic Press, New York 1969.
39. Hayes-Roth F.: Computer *17*, 11 (1984).
40. Gliviak F., Kubiš J.: Informační Systémy *14*, 185 (1985).
41. Buchanan B. G., Duda R. O. in: *Advances in Computers* (M. Yovits, Ed.), Vol. 22, p. 163. Academic Press, New York 1983.
42. Hayes-Roth F.: Computer *17*, 263 (1984).
43. Clare J., Ostler N.: Data Processing *26*, 40 (1984).
44. Bartoš P.: Informační Systémy *15*, 269 (1986).
45. Iványek J., Švenda J., Ferjenčík J.: Informační Systémy *5*, 481 (1988).
46. Harmon P., King D. in: *Expert Systems – Artificial Intelligence in Business*, p. 283. Wiley, New York 1985.
47. Bartoš P., Kalaš I.: Informačné Systémy *18*, 285 (1989)
48. Winston P. H., Horn B. K. P. in: *LISP.* Addison–Wesley, Reading 1981.
49. Williamson M. in: *Artificial Intelligence for Microcomputers – The Guide for Business Decisionmakers.* Simon and Schuster, New York 1986.
50. Kelemen J. in: *Úvod do umelej inteligencie.* Comenius University, Bratislava 1984.
51. Bratko I. in: *PROLOG Programming for Artificial Intelligence.* Addison–Wesley, Wokingham 1986.
52. Kleywegt G. J., Luinge H. J., Schuman B. J. P.: Chemometr. Intell. Lab. Syst. *4*, 273 (1988).
53. Kleywegt G. J., Luinge H.J., Schuman B. J. P.: Chemometr. Intell. Lab. Syst. *5*, 117 (1989).
54. Clocksin W. F., Mellish C. S. in: *Programming in PROLOG.* Springer, Berlin 1984.
55. Clark K. L., McCabe F. G. in: *Micro-PROLOG: Programming in Logic.* Prentice Hall, Englewood Cliffs 1984.
56. Brodie L. in: *Starting FORTH.* Prentice Hall, Englewood Clifs 1981.
57. Golberg A., Ribson D. in: *Smalltalk-80, the Language and its Implementation.* Addison–Wesley, Reading 1984.
58. Adam-Nicolle A. in: *Revue d'intelligence artificielle*, Vol. 1. Hermés, Paris 1990.
59. Marie S., Adam-Nicolle A.: New J. Chem. *14*, 933 (1990).
60. Marie S.: *Thesis.* University of Caen, Caen 1991.
61. Adam-Nicolle A., Proquet Ch.: New J. Chem. *14*, 905 (1990).

62. Forgy C. L. in: *The OPS 5 User's Manual. Technical Report CMU-CS-81-135.* Carnegie Mellon University, Pittsburgh 1981.
63. McDermott J.: Artif. Intell. *19*, 39 (1982).
64. Fain J., Gorlin D., Hayes-Roth F., Rosenschein S. J., Sowizral H., Waterman D. in: *The ROSIE Language Reference Manual. Technical Report N-1647-ARPA.* Rand Corporation, Santa Monica 1981.
65. Bahill A. T., Ferell W. R.: IEEE Expert *1986*, 59.
66. Sloman A. in: *POPLOG – A Multi-User Program Development Environment.* University of Sussex, Brighton 1982.
67. Barrett R., Ramsey A., Sloman A. in: *POP-11. Practical Language for Artificial Intelligence.* Halsted Press, Ellis Horwood, Chichester 1985.
68. Reboh R. in: *Knowledge Engineering Techniques and Tools in the Prospector Environment.* Technical Note 243. p. 149. SRI International, Menlo Park 1981.
69. Weiss S. M., Kulikowski C. A. in: *Proceedings of the 6th International Joint Conference on Artificial Intelligence*, Tokyo 1979, p. 942. Tokyo University Press, Tokyo.
70. Goss S.: Chemistry in Australia *1990*, 420.
71. Erman L. D., London P. E., Fickas S. F. in: *Proccedings of the 7th International Joint Conference on Artificial Intelligence, Vancouver 1981*, p. 409. British Columbia University Press, Vancouver 1981.
72. Davis R., Lenat D. B. in: *Knowledge-Based Systems in Artificial Inteliigence.* McGraw–Hill, New York 1982.
73. Van Melle W. in: *EMYCIN: A Domain Independent Systems That AIDS in Constructing Knowledge-Based Consultation Programs. HPP-Memmo-81-27.* Stanford University, Stanford 1981.
74. Mařík V., Zdráhal Z. (Eds) in: *Metody umělé inteligence a expertní systémy II*, p. 175. Czech Technical University, Praha 1985.
75. Duda R. O., Gaschnig J. G.: Byte *6*, 238 (1981).
76. Duda R. O., Hart P. E., Nilsson N. J., Sutherland G. in: *Pattern Directed Inference Systems* (D. A. Waterman and F. Hayes-Roth, Eds). Academic Press, New York 1978.
77. Duda R. O., Hart P. E., Barrett P., Gaschnig J., Konolige K., Reboh R., Slocum J. in: *Expert Systems in the Micro-Electronic Age* (D. Michie, Ed.). Edinburgh University Press, Edinburgh 1979.
78. Weiss S. M., Safir A.: Comput. Biol Med. *8*, 25 (1978).
79. Weiss S. M., Kulikowski C. A., Amarel S., Safir A.: Artif. Intell. *11*, 145 (1978).
80. Erman L. D., Hayes-Roth F., Lesser V., Reddy D.: ACM Comput. Surv. *12*, 213 (1980).
81. Shortliffe E. H. in: *Computer Based Medical Consultation. MYCIN.* Elsevier, Oxford–New York–Amsterdam 1976.
82. Buchanan B. G., Shortliffe E. H. in: *Rule Based Expert Systems: The Mycin Experiments of the Stanford Heuristic Programming Project.* Addison–Wesley, Reading 1984.
83. Hayes-Roth F., Waterman D. A., Lenat D. B. in: *Building Expert System.* Addison–Wesley, Reading 1983.
84. Gevarter W. B.: Computer *20*, 24 (1987).
85. Waterman D. A. in: *A Guide to Expert Systems.* Addison–Wesley, Reading 1986.
86. Harmon P., Maus R., Morrisey W. in: *Expert Systems: Tools and Applications.* Wiley, New York 1988.
87. Holme B. A., Pierce T. H. in: *Expert System Applications in Chemistry* (B. A. Holme and T. H. Pierce, Eds), ACS Symp. Ser. 408. American Chemical Society, Washington 1989.
88. Wade A. P., Crouch S. R., Betteridge D.: Trends Anal. Chem. *7*, 358 (1988).
89. Turing A. M. in: *Computers and Thought* (A. A. Feigenbaum and J. Feldman, Eds). McGraw–Hill, New York 1963.
90. McCulloch W. S., Pitts W.: Bull. Math. Biophys. *5*, 115 (1943).
91. Rosenblatt F. in: *Principles of Neurodynamics.* Spartan Books, Washington 1962.

92. Minsky M. L., Papert S. A. in: *Perceptrons*. MIT Press, Cambridge 1969.
93. Hopfield J. J.: Proc. Natl. Acad. Sci. U.S.A. *79*, 2554 (1982).
94. Rumelhart D. E., Hinton G. E., Williams R. J. in: *Microstructure of Cognition* (D. E. Rumelhart and J. L. McClelland, Eds), Vol. 1, p. 318. MIT Press, Cambridge 1986.
95. Freman J. A. in: *Neural Networks: Algorithms, Applications and Programming Techniques*. Addison–Wesley, Reading 1991.
96. Zupan J., Gasteiger J.: Anal. Chim. Acta *248*, 1 (1991).
97. Jansson P. A.: Anal. Chem. *63*, 357A (1991).
98. Winograd T.: Cognitive Sci. *4*, 209 (1980).
99. Cigánik J.: Informačné Systémy *1*, 87 (1989).
100. Charniak E.: Cognitive Sci. *7*, 171 (1983).
101. Brachman R. J. in: *Associate Networks* (N. V. Findler, Ed.), p. 3. Academic Press, London 1970.
102. Luinge H. J.: Vibr. Spectrosc. *1*, 3 (1990).
103. Ramos S. L.: Anal. Chem. *58*, 294R (1986).
104. Delaney M. F.: Anal. Chem. *56*, 261R (1984).
105. Zupan J., Munk M. E.: Anal. Chem. *58*, 3219 (1986).
106. Jalsovszky G., Holly S.: J. Mol. Struct. *175*, 263 (1988).
107. Domokos L., Frank I., Matolcsy G., Jalsovszky G.: Anal. Chim. Acta *154*, 181 (1984).
108. Smith D. H., Gray N. A. B., Nourse C. W.: Anal. Chim. Acta *133*, 471 (1981).
109. Yamasaki T., Abe H., Kudo Y., Sasaki S. in: *Computer Assisted Structure Elucidation*, ACS Symp. Ser. 54, p. 108. American Chemical Society, Washington 1981.
110. Hippe Z., Hippe R., Duliban J., Debska B. in: *Techn. Univ. Res. Project*. Techn. University Publ., Rzeszow 1981.
111. Hippe Z., Hippe R., Duliban J.: Fresenius Z. Anal. Chem. *311*, 440 (1982).
112. Carhart R. E., Smith D. H., Gray N. A. B., Nourse J. G., Djerassi C.: J. Am. Chem. Soc. *46*, 1708 (1981).
113. Djerassi C., Smith D. H., Crandell C. W.: Pure Appl. Chem. *54*, 2425 (1982).
114. Lederberg J., Sutherland G. L., Buchanan B. G., Feigenbaum E. A., Robertson A. U., Duffield A. M., Djerassi C.: J. Am. Chem. Soc. *91*, 2973 (1969).
115. Feigenbaum E. A., Buchanan B. G., Lederberg J. in: *Machine Intelligence 6*. University Press, Edinburgh 1971.
116. Gray N. A. B.: Chemometr. Intell. Lab. Syst. *5*, 11 (1988).
117. Gray N. A. B. in: *Computer Assisted Structure Elucidation*. Wiley–Interscience, New York 1986.
118. Egli H., Smith D. H., Djerassi C.: Helv. Chim. Acta *65*, 1898 (1982).
119. Enke G. Ch., Wade A. P., Palmer P. T., Hart K. J.: Anal. Chem. *59*, 136A (1987).
120. Lindsay R. K. Buchanan B. G., Feigenbaum E. A., Lederberg J. in: *Applications of Artificial Intelligence for Organic Chemistry. The DENDRAL Project*. McGraw–Hill, New York 1980.
121. Gray N. A. B., Crandell C. W., Nourse J. G., Smith D. H., Dageforde M. L., Djerassi C.: J. Org. Chem. *46*, 703 (1981).
122. Gray N. A. B.: Artif. Intell. *22*, 1 (1984).
123. Kvasnička V., Pospíchal J.: Collect. Czech. Chem. Commun. *56*, 1777 (1991).
124. Kvasnička V., Pospíchal J., Koča J., Hladká E., Matyska L.: Chem. Listy *85*, 449 (1990).
125. Baláž V., Kvasnička V., Pospíchal J.: Pokroky matematiky, fyziky a chémie *36*, 1 (1991).
126. Buchs A., Delfino A. B., Duffield A. M., Djerassi C., Buchanan B. G., Feigenbaum E. A., Lederberg J.: Helv. Chim. Acta *53*, 1394 (1970).
127. Carhart R. E., Smith D. E., Brown H., Djerassi C.: J. Am. Chem. Soc. *97*, 5755 (1975).
128. Crandell C. W., Gray N. A. B., Smith D. H.: J. Chem. Inf. Comput. Sci. *22*, 48 (1982).
129. Nourse J. G., Smith D. H., Carhart D. E., Djerassi C.: J. Am. Chem. Soc. *102*, 6289 (1980).

130. Fella A. L., Nourse J. G., Smith D. H.: J. Chem. Inf. Comput. Sci. *23*, 43 (1983).
131. Buchanan B. G., Smith D. H., White W. C., Gritter R. J., Feigenbaum E. A., Lederberg J., Djerassi C.: J. Am. Chem. Soc. *98*, 6168 (1976).
132. Buchanan B. G., Feigenbaum E. A.: Artif. Intell. *11*, 5 (1978).
133. Mitchell T. M., Schwenzer G. M.: Org. Magn. Reson. *11*, 378 (1978).
134. Gribov L. A.: Anal. Chim. Acta *122*, 249 (1980).
135. Luinge H. J., van der Mass J. H.: Chemometr. Intell. Lab. Syst. *8*, 157 (1990).
136. Sasaki S.: Kobutsugaku Zasshi *17*, 289 (1986).
137. Sasaki S., Kudo Y., Ochiai S., Abe H.: Microchim. Acta *1971*, 726.
138. Sasaki S., Abe H., Hirota Y., Ishida Y., Kudo S., Ochiai S., Sato K., Yamasaki T.: J. Chem. Inf. Comput. Sci. *18*, 211 (1978).
139. Sasaki S., Fujiwara I., Abe H., Yamasaki T.: Anal. Chim. Acta *122*, 87 (1980).
140. Oshima T., Ishida K., Saito K., Sasaki S.: Anal. Chim. Acta *122*, 95 (1980).
141. Funatsu K., Miyabayashi N., Sasaki J.: J. Chem. Inf. Comput. Sci. *28*, 18 (1988).
142. Funatsu K., Del Carpio C. A., Sasaki S.: Fresenius Z. Anal. Chem. *324*, 750 (1986).
143. Shelley C. A., Munk M. E.: Anal. Chim. Acta *133*, 507 (1981).
144. Trulson M. O., Munk M. E.: Anal. Chem. *55*, 2137 (1983).
145. Christie B. D., Munk M. E.: J. Chem. Inf. Comput. Sci. *28*, 87 (1988).
146. Shelley C. A., Munk M. E., Roman R. V.: Anal. Chim. Acta *103*, 121 (1978).
147. Lipkus A. H., Munk M. E.: J. Chem. Inf. Comput. Sci. *28*, 9 (1988).
148. Munk C. A., Shelley C. A., Woodruff H. B., Trulson M. O.: Fresenius Z. Anal. Chem. *313*, 473 (1982).
149. Woodruff H. B.: Trends Anal. Chem. *3*, 72 (1984).
150. Woodruff H. B., Tomellini S. A., Smith G. M. in: *Artificial Intelligence Applications in Chemistry* (T. H. Pierce and B. A. Hohne, Eds), ACS Symp. Ser. 306, p. 312. American Chemical Society, Washington 1986.
151. Woodruff H. B., Smith G. M.: Anal. Chem. *52*, 2321 (1980).
152. Woodruff H. B., Smith G. M.: Anal. Chim. Acta *133*, 545 (1981).
153. Tomellini S. A., Stevenson J. M., Woodruff H. B.: Anal. Chem. *56*, 67 (1984).
154. Tomellini S. A., Hartwick R. A., Stevenson J. M., Woodruff H. B.: Anal. Chim. Acta *162*, 227 (1984).
155. Tomellini S. A., Hartwick R. A., Woodruff H. B.: Appl. Spectrosc. *39*, 331 (1985).
156. Wythoff B. J., Buck C. F., Tomellini S. A.: Anal. Chim. Acta *217*, 203 (1989).
157. Puskar M. A., Levine S. P., Lowry S. R.: Anal. Chem. *58*, 1156 (1986).
158. Puskar M. A., Levine S. P., Lowry S. R.: Anal. Chem. *58*, 1981 (1986).
159. Ying L., Levine S. P., Tomellini S. A., Lowry S. R.: Anal. Chim. Acta *227*, 343 (1989).
160. Ying L., Levine S. P., Tomellini S. A., Lowry S. R.: Anal. Chem. *59*, 2197 (1987).
161. Wythoff B. J., Tomellini S. A.: Anal. Chim. Acta *227*, 359 (1989).
162. Hong-kui X., Levine S. P., D'Arcy J. B.: Anal. Chem. *61*, 2709 (1989).
163. Gribov L. A., Elyashberg M. E., Koldashov V. N., Pletnev I. V.: Anal. Chim. Acta *148*, 159 (1983).
164. Gribov L. A., Elyashberg M. E.: J. Mol. Struct. *5*, 179 (1970).
165. Gribov L. A., Elyashberg M. E., Moscovkina L. A.: J. Mol. Struct. *9*, 357 (1971).
166. Elyashberg M. E., Serov V. V., Martirosyan E. R., Zlatina L. A., Karasev Yu. Z., Koldashov V. V., Yampolskii Yu. Yu.: Theochem *76*, 191 (1991).
167. Gribov L. A., Elyashberg M. E., Raikhstat M. M.: J. Mol. Struct. *53*, 81 (1979).
168. Gribov L. A., Dementev V. A., Todorovsky A. T.: J. Mol. Struct. *50*, 389 (1978).
169. Debska B., Duliban J., Guzowska-Swider B., Hippe Z.: Anal. Chim. Acta *133*, 303 (1981).
170. Hippe Z.: Trends Anal. Chem. *2*, 240 (1983).

171. Szalontai G., Simon Z., Csapo Z., Farkas M., Pfeifer G.: Anal. Chim. Acta *133*, 31 (1981).
172. Farkas M., Markos J., Szepesvary P., Bartha I., Szalontai G., Simon Z.: Anal. Chim. Acta *133*, 19 (1981).
173. Szalontai G., Recsey Z., Csapo Z.: Anal. Chim. Acta *140*, 309 (1982).
174. Luinge H. J.: Anal. Proc. *27*, 266 (1990).
175. Luinge H. J.: Trends Anal. Chem. *9*, 66 (1990).
176. Blaffert T.: Anal. Chim. Acta *191*, 161 (1986).
177. Haraki K. S., Venkataraghavan R., McLafferty F. W.: Anal. Chem. *53*, 386 (1981).
178. Curry Bo in: *Computer Enhanced Analytical Spectroscopy* (H. L. C. Meuzelaar, Ed.), Vol. 2, p. 183. Plenum Press, New York 1990.
179. Curry Bo in: *Artificial Intelligence Applications in Chemistry* (T. H. Pierce and B. A. Hohne, Eds), ACS Symp. Ser. 306, p. 350. American Chemical Society, Washington 1986.
180. Damo C., Dachun C., Teshu K., Shaoyu C.: Anal. Chim. Acta *133*, 575 (1981).
181. Visser T., van der Maas J. H.: Anal. Chim. Acta *122*, 363 (1980).
182. Ayscough P. B., Chinnick S. J., Oynowski R., Edwards P.: Chem. Ind. (London) *1987*, 515.
183. Hippe Z.: J. Chem. Inf. Comput. Sci. *25*, 344 (1985).
184. Guonan S., Xun C., Weidong H., Faxiao S., Chuan W.: Kexue Tongbao *32*, 960 (1987).
185. Gastmans J. P., De Paula V. E., Furlan M.: Comput. Chem. *12*, 285 (1988).
186. Gastmans J. P., Lopes M. N., Furlan M., Emerenciano V. de P.: Comput. Chem. *14*, 75 (1990).
187. Gastmans J. P., Zurita J. C., Zurita J. S., Emerenciano V. de P.: Anal. Chim. Acta *217*, 85 (1989).
188. Gastmans J. P., Furlan M.: Quimica Nova *12*, 25 (1989).
189. Gastmans J. P., Furlan M., Lopes M. N., Borges J. H. G., Emerenciano V. de P.: Quimica Nova *13*, 2 (1990).
190. Gastmans J. P., Furlan M., Lopes M. N., Borges J. H. G., Emerenciano V. de P.: Quimica Nova *13*, 75 (1990).
191. Craig I. D. in : *The Cassandra Architecture: Distributed Control in a Blackboard System.* Ellis Horwood, Chichester 1989.
192. Dubois J. E., Carabedian M., Dagane I.: Anal. Chim. Acta *158*, 217 (1984).
193. Dolata D. P., Carter R. E.: J. Chem. Inf. Comput. Sci. *27*, 36 (1987).
194. Cheng H. N., Bennett M. A.: Anal. Chim. Acta *242*, 43 (1991).
195. Hong H. X., Xin X.: J. Chem. Inf. Comput. Sci. *30*, 203 (1990).
196. Hippe Z. in: *Artificial Intelligence in Chemistry: Structure Elucidation and Simulation of Organic Reactions.* Elsevier, Amsterdam 1991.
197. Perkins J. H., Hasenoehrl E: J., Griffits P. R.: Anal. Chem. *63*, 1738 (1991).
198. Pomeroy R. S., Kolczynski J. D., Denton M. B.: Appl. Spectrosc. *45*, 1111 (1991).
199. Jakuš, V., Woodruff H. B., Miertuš S., Májek P.: Unpublished results.
200. Thomsen J. U., Meyer B.: J. Magn. Reson. *84*, 212 (1989).
201. Kvasnička V.: J. Math. Chem. *6*, 63 (1991).
202. Kvasnička V.: Chem. Papers *44*, 775 (1990).
203. Curry Bo, Rumelhart D. E.: Tetrahedron Comput. Methodol. *3*, 213 (1990).
204. Robb E. W., Munk M. E.: Mikrochim. Acta *I*, 131 (1991).
205. Munk M. E., Madison M. S., Robb E. W.: Mikrochim. Acta *II*, 505 (1991).
206. Munk M. E., Christie B. D.: Anal. Chim. Acta *216*, 57 (1989).
207. Bos M. T., Weber H. T.: Anal. Chim. Acta *247*, 97 (1991).
208. Tušar M., Zupan J. in: *Software Development in Chemistry 4* (J. Gasteiger, Ed.). Springer, Berlin 1990.
209. Lippmann R. P.: IEEE ASSP Magazine April 4, 1987.
210. Barker P. G.: Anal. Proc. (London) *28*, 110 (1991).

211. Zupan J.: Anal. Chim. Acta *235*, 53 (1990).
212. Wythoff B. J., Levine S. P., Tomellini S. A.: Anal. Chem. *62*, 2702 (1990).
213. Otto M., Hörchner U. in: *Software Development in Chemistry 4* (J. Gasteiger, Ed.), p. 377. Springer, Berlin 1990.
214. Rivail J. F. in: *Computational Advances in Organic Chemistry: Molecular Structure and Reactivity* (C. Ögretir and I. G. Csizmadia, Eds), p. 229. Kluwer, Dordrecht 1991.
215. Čársky P., Zahradník R., Hess A. B., Schaad L. J.: Chem. Listy *79*, 337 (1984).
216. Adler B., Binshuba A., Herman A., Lenski U., Scheubeck H., Weissbarth H.: Z. Chem. *26*, 157 (1986).
217. Polák R., Zahradník R. in: *Kvantová chemie. Základy teorie a aplikace.* SNTL, Praha 1985.
218. Meissner L., Bartlett R. J.: J. Chem. Phys. *94*, 6670 (1991).
219. Pariser R., Parr R. G.: J. Chem. Phys. *21*, 466 (1953).
220. Del Bene J., Jaffé H. H.: J. Chem. Phys. *49*, 1221 (1968).
221. Horák M., Papoušek D. in: *Infračervená spektra a struktura molekul.* Academia, Praha 1976.
222. Pulay P.: Mol. Phys. *17*, 197 (1969).
223. Pulay P.: Theor. Chim. Acta *50*, 299 (1979).
224. Ramsey N. F.: Phys. Rev. *91*, 203 (1955).
225. Miertuš S., Boča R., Pelikán P., Valko L.: Chem. Phys. *11*, 237 (1976).
226. Pople J. A., Beveridge L.: *Approximate Molecular Orbital Theory.* Mc Graw–Hill, New York 1970.
227. Segal A. in: *Semiempirical Methods of Electronic Structure Calculation*, Part B, Vol. 8. Plenum Press, New York 1977.
228. Lindeman L. P., Adams J. Q.: Anal. Chem. *43*, 1245 (1971).
229. Riera J. M., Mariné J.: Int. J. Quant. Chem. *25*, 715 (1984).
230. Craik J. D., Brownlee R. T.C.: Ann. Rep. NMR Spectrosc. *21*, 2 (1989).
231. Pytela O.: Collect. Czech. Chem. Commun. *53*, 1333 (1988).
232. Rivail J. L., Rinaldi D., Ruiz-Lopez M. F. in: *Theoretical and Computational Models for Organic Chemistry* (S. J. Formosinho, Ed.), p. 79. Kluwer, Dordrecht 1991.
233. Bonnacorsi R., Floris F., Tomasi J.: J. Mol. Liq. *47*, 25 (1990).
234. Luzhkov V., Warshel A.: J. Comput. Chem. *13*, 199 (1992).
235. Perrot G., Cheng B., Gibson K. D., Vila J., Palmer K. A., Nayeem A., Maigret B., Scheraga H. A.: J. Comput. Chem. *13*, 1 (1992).
236. Tapia O.: J. Mol. Struct. (Theochem) *226*, 59 (1991).
237. Teleman O., Svensson Bo, Jönsson Bo: Comput. Phys. Commun. *62*, 307 (1991).
238. Montagnani R., Tomasi J.: Int. J. Quant. Chem. *39*, 851 (1991).
239. Takahashi Y., Hosokawa K., Yoshida F., Ozaki M., Sasaki S.: Anal. Chim. Acta *217*, 61 (1989).
240. Štrouf O., Mařík V.: Collect. Czech. Chem. Commun. *55*, 317 (1990).
241. Jakuš V., Miertuš S.: Chem. Listy *84*, 468 (1990).
242. Moll R.: Anal. Chim. Acta *235*, 189 (1990).
243. Miertuš S., Jakuš V., Matisová E.: Chromatographia *30*, 144 (1990).
244. Jakuš V., Miertuš S.: Collect Czech. Chem. Commun. *56*, 2833 (1991).
245. Rittich B., Rudolfová Š., Jandera P.: Collect. Czech. Chem. Commun. *55*, 2138 (1990).
246. Klopman G.: J. Am. Chem. Soc. *106*, 7315 (1984).
247. Klopman G., Henderson V.: J. Math. Chem. *7*, 187 (1991).
248. Smith R. M., Burr C.M.: J. Chromatogr. *485*, 325 (1989).
249. Milne R.: J. Chromatogr. *485*, 341 (1989).
250. Aoyama T., Suzuki T., Ichikawa H.: J. Med. Chem. *33*, 905 (1990).
251. Aoyama T., Suzuki T., Ichikawa H.: J. Med. Chem. *33*, 2583 (1990).
252. Gemperline P. J., Long J. R., Gregoriou V. G.: Anal. Chem. *63*, 2313 (1991).

253. Derde M. P., Buydens L., Guns C., Massart D. L., Hopke P. K.: Anal. Chem. *59*, 1868 (1987).
254. Moriguchi I., Komatsu K., Matsushita Y.: J. Med. Chem. *23*, 20 (1979).
255. Solberg H. E.: Crit. Rev. Clin. Lab. Sci. *9*, 209 (1978).
256. Bangov I. P.: J. Inf. Comput. Sci. *30*, 277 (1990).
257. Johnson M., Tsai Ch., Nicholson V.: J. Math. Chem. *7*, 3 (1991).
258. Kvasnička V., Pospíchal J., Baláž V.: Theor. Chim. Acta *79*, 65 (1991).
259. Zou Y., Johnson M. A., Tsai C. C.: J. Chem. Inf. Comput. Sci. *30*, 442 (1990).
260. Herges R.: J. Chem. Inf. Comput. Sci. *30*, 377 (1990).
261. Rücker G., Rücker C.: J. Chem. Inf. Comput. Sci. *31*, 422 (1991).
262. Rücker G., Rücker C.: J. Chem. Inf. Comput. Sci. *30*, 187 (1990).
263. Trinajstic N. in: *Mathematics and Computational Concepts in Chemistry*. Ellis Horwood, Chichester 1986.
264. Navech J., Pelletier G., Despax C.: New J. Chem. *15*, 527 (1991).
265. Corey E. J.: Pure Appl. Chem. *14*, 19 (1967).
266. Corey E. J. in: *The Logic of Chemical Syntheses*. Wiley, New York 1989.
267. Seebach D.: Angew. Chem., Int. Ed. Engl. *29*, 1320 (1990).
268. Long A. K., Rubenstein S. D., Joncas L. J.: Chem. Eng. News *61*, 23 (1983).
269. Corey E. J., Long A. K., Rubenstein S. D.: Science *228*, 408 (1985).
270. Corey E. J., Howe W. J., Orf H. W., Pensak D. A., Peterson G.: J. Am. Chem. Soc. *97*, 6116 (1975).
271. Corey E. J., Wipke W. T.: Science *166*, 178 (1969).
272. Wipke W. T., Dyott T. M.: J. Am. Chem. Soc. *96*, 4825 (1984).
273. Wipke W. T., Gund P.: J. Am. Chem. Soc. *100*, 9107 (1978).
274. Wipke W. T., Ouchi G. I., Krishnan S.: Artif. Intell. *11*, 173 (1978).
275. Metivier P., Gushurst A. J., Jörgensen W. L.: J. Org. Chem. *52*, 3724 (1987).
276. Jörgensen W. L., Salatin T. D., McLaughli D.: J. Org. Chem. *46*, 5284 (1981).
277. Burnier J. S., Jörgensen W. L.: J. Org. Chem. *49*, 3001 (1984).
278. Gasteiger J.: Top. Curr. Chem. *137*, 19 (1987).
279. Gasteiger J., Jochum C.: Top. Curr. Chem. *74*, 93 (1978).
280. Gasteiger J., Ihlenfeldt W. D., Röse P., Wanke R.: Anal. Chim. Acta *235*, 65 (1990).
281. Nevalainen V.: *Thesis*. University Joensuu, Helsinki 1988.
282. Blair J., Gasteiger J., Gillespie P. D., Gillespie C., Ugi I. in: *Computer Representation and Manipulation of Chemical Structure Information*. Wiley, New York 1974.
283. Haggin J.: Chem. Eng. News *61*, 7 (1983).
284. Lee T.V.: Chemometr. Intell. Lab. Syst. *31*, 497 (1991).
285. Hendrickson J. B.: Angew. Chem., Int. Ed. Engl. *29*, 1286 (1990).
286. Peishoff C. E., Jörgensen W. L.: J. Org. Chem. *48*, 1970 (1983).
287. Weise A.: Wiss. Forsch. *30*, 269 (1980).
288. Weise A.: J. Prakt. Chem. *322*, 761 (1980).
289. Schubert W.: Match *6*, 213 (1979).
290. Gelernter H., Royce-Rose J., Chyouhwa Ch.: J. Chem. Inf. Comput. Sci. *30*, 492 (1990).
291. Gelernter H. L., Sridharan N. S., Hart A. J., Yen S. C., Fowler F. W., Shue H. J.: Top. Curr. Chem. *137*, 19 (1987).
292. Takahashi M., Dogane I., Yoshida M., Yamachika H., Takabatake T., Bersohn M.: J. Chem. Inf. Comput. Sci. *30*, 436 (1990).
293. Takabatake T., Bersohn M.: Pure Appl. Chem. *62*, 1977 (1990).
294. Blurock E. S.: J. Chem. Inf. Comput. Sci. *30*, 507 (1990).
295. Blurock E. S.: Tetrahedron Comput. Methodol. *2*, 207 (1989).
296. Bauer J., Herges R., Fontain E., Ugi I.: Chimia *39*, 43 (1985).

297. Bauer J.: Tetrahedron Comput. Methodol. *2*, 269 (1989).
298. Bauer J., Fontain E., Forstmeyer D., Ugi I.: Tetrahedron Comput. Methodol. *1*, 129 (1988).
299. Ugi I., Bauer I., Baumgartner R., Fontain E., Forstmeyer G., Lohberger S.: Pure Appl. Chem. *60*, 1573 (1988).
300. Ugi I., Bauer J., Bley K., Dengler A., Fontain E., Knauer M., Lohberger S.: Theochem, J. Mol. Struct. *230*, 73 (1991).
301. Laurenco C.: *Thesis*. Pasteur University, Strasbourg 1985.
302. Laurenco C., Py M., Napoli A., Quinqueton J., Castro B.: New J. Chem. *14*, 921 (1990).
303. Baumer L., Sala G., Sello G.: Anal. Chim. Acta *235*, 209 (1990).
304. Nakayama T.: J. Chem. Inf. Comput. Sci. *31*, 497 (1991).
305. Pollet P.: New J. Chem. *14*, 957 (1990).
306. Moreau G.: Nouv. J. Chim. *2*, 187 (1978).
307. Barone R., Chanon M., Cende J. M., Cadiot P.: Bull. Soc. Chim. Belg. *91*, 333 (1982).
308. Funatsu K., Sasaki S. I.: Tetrahedron Comput. Methodol. *1*, 27 (1988).
309. Jauffrey P.: *Thesis*. University of Caen, Caen 1987.
310. Matyska L., Koča L.: J. Chem. Inf. Comput. Sci. *31*, 380 (1991).
311. Dubois J. E. in: *Description des Structures Chimiques, Acquisition Restitution et Conception*. Paris Univesity, Paris 1964.
312. Dubois J. E. in: *Computer Representation and Manipulation of Chemical Information*, p. 239. Wiley–Interscience, New York 1974.
313. Slanina Z.: J. Mol. Struct. *185*, 217 (1989).
314. Koča J.: Theor. Chim. Acta *80*, 29 (1991).
315. Luce H. H., Govind R.: Tetrahedron Comput. Methodol. *3*, 143 (1990).
316. Elrod D. W., Maggiora G. M., Trenary R. G.: J. Chem. Inf. Comput. Sci. *30*, 477 (1990).
317. Dugundji J., Ugi I.: Top. Curr. Chem. *39*, 19 (1973).
318. Kvasnička V., Pospíchal J.: Theochem, J. Mol. Struct. *235*, 227 (1991).
319. Walters J. P.: Anal. Chem. *63*, 977A (1991).
320. Hwang K. S., Tomita S., O'Shima E.: Int. Chem. Eng. *31*, 134 (1991).
321. Tomita S., Hwang K. S., O'Shima E., McGreavy C.: J. Chem. Eng. Jpn. *22*, 364 (1989).
322. Peichang Lu, Hongxin H.: J. Chromatogr. Sci. *27*, 690 (1989).
323. Martz P. R., Heffron M., Griffith O. M. in: *Artificial Intelligence Applications in Chemistry* (T. H. Pierce and B. A. Hohne, Eds), ACS Symp. Ser. 306, p. 297. American Chemical Society, Washington 1986.
324. Gunasingham H., Srivasan B., Ananda L.: Anal. Chim. Acta *182*, 193 (1986).
325. Bach R., Karnicky J., Abbott S. in: *Artificial Intelligence Application in Chemistry* (T. H. Pierce and B. A. Hohne, Eds), ACS Symp. Ser. 306, p. 278. American Chemical Society, Washington 1986.
326. Berry V.: J. Liq. Chromatogr. *12*, 1090 (1985).
327. Glajch J. L.: LC-GC *6*, 30 (1988).
328. Bridge T. P., Williams M. H., Fell A. F.: Anal. Proc. (London) *25*, 43 (1988).
329. Borman S. A.: Anal. Chem. *58*, 1192A (1986).
330. Jinno K. in: *A Computer-Assisted Chromatography System*. Hüthig, Heidelberg 1990.
331. Dolan J. W.: Am. Lab. (Fairfield, Conn.) *12*, 52 (1990).
332. Reich G.: Anal. Chim. Acta *201*, 171 (1987).
333. Hattori T., Kito S., Murakami Y.: Chem. Lett. *1988*, 1269.
334. Kafarov V. V., Meshalkin V. P.: Dokl. Akad. Nauk SSSR *306*, 409 (1989).
335. Banares-Alcantara R., Westemberge A. W.: Comput. Chem. Eng. *9*, 127 (1985).
336. Chéruy A. in: *Résultats et perspectives en automatique* (J. M. Dion and I. D. Landau, Eds), p. 235. Hermés, Paris 1988.

337. Bhagat P.: Chem. Eng. Process. *86*, 55 (1990).
338. Hoskins J. C., Himmelblau D. M.: Comput. Chem. Eng. *12*, 881 (1988).
339. Watanabe K., Matsura I., Abe M., Kubota M., Himmelblau D. M.: AIChE J. *35*, 1803 (1989).
340. Bhat N., McAvoy T. J.: Comput. Chem. Eng. *14*, 573 (1990).
341. Venkatasubramanian V., Chan K.: AIChE J. *35*, 1993 (1989).
342. Psichogios D. C., Ungar L. H.: Ind. Eng. Chem. Res. *30*, 2564 (1991).
343. Wuensch G., Schummig K.: Fresenius J. Anal. Chem. *339*, 800 (1991).
344. Brunovská A. in: *Malá optimalizácia* (E. Známová, Ed.). Alfa, Bratislava 1988.
345. Rutan S. C.: Anal. Chem. *63*, 1109A (1991).
346. He P., Avery J. P., Faulkner L. R.: Anal. Chem. *54*, 1313A (1982).
347. Azhogin V. V., Zgurovskii M. Z., Rudenki S. S.: Khim. Tekhnol. *1982*, 31.
348. Walters J. P.: Anal. Chem. *63*, 1077A (1991).
349. Walters J. P.: Anal. Chem. *63*, 1179A (1991).
350. Diamond D.: Trends Anal. Chem. *8*, 58 (1989).
351. Liebman S. A., Levy E. J., Lurcott S., O'Neill S., Guthrie J., Ryan T., Yocklovich S.: J. Chromatogr. Sci. *27*, 118 (1989).
352. S. A. B.: Anal. Chem. *58*, 1312A (1984).
353. Salin E. D., Winston P. H.: Anal. Chem. *64*, 49A (1992).
354. Schlieper W. A., Isenhour T. I., Marshall J. C.: Anal. Chem. *60*, 1142 (1988).
355. Kingston H. M., Ruegg F., De Voe J., Clements C.: Adv. Lab. Autom. Rob. *5*, 27 (1989).
356. Kramer G. W., Fuchs P. L.: Adv. Lab. Autom. Rob. *1987*, 339.
357. Barker P. G. in: *Computers in Analytical Chemistry*. Pergamon Press, Oxford 1983.
358. Newman A. R.: Anal. Chem. *62*, 29A (1990).
359. Isenhour T. L., Marshall J. C.: J. Res. Natl. Bur. Stand. (U.S.) *93*, 209 (1988).
360. Eckert-Tilotta S. E., Isenhour T. L.: Anal. Chim. Acta *254*, 215 (1991).
361. Barker P. G.: Irish Chem. News *1989*, 28.
362. Judson P. N. in: *Chemical Information 2* (H. R. Colier, Ed.), p. 13. Springer, Berlin 1991.
363. Langstaff E. M., Sobala B. K., Zahm B. C. in: *Chemical Information 2* (H. R. Collier, Ed.), p. 17. Springer, Berlin 1991.
364. Dubois J. E., Mathieu G., Pegnet P., Panayes A., Doucet J.: J. Chem. Inf. Comput. Sci. *30*, 290 (1990).
365. Buchanan B. G., Feigenbaum E. A., Lederberg J.: Chemometr. Intell. Lab. Syst. *5*, 33 (1988).
366. Gray N. A. B.: Chemometr. Intell. Lab. Syst. *5*, 37 (1988).
367. Stolow R. D., Joncas L. J.: J. Chem. Educ. *57*, 868 (1988).
368. Box V. G. S.: J. Chem. Educ. *68*, 662 (1991).
369. Smith H. R., Fink P. K., Lusth J. C. in: *Expert Systems in Government Symposium*. IEEE, Washington 1985.
370. Vandeginste B. G. M., Klaessens J. W. A., Kateman G.: Trends Anal. Chem. *6*, 252 (1987).
371. Mark F.: Computer World *5*, 15 (1992).
372. Gray N. A. B.: Anal. Chim. Acta *210*, 8 (1988).

Translated by P. Adámek.