AN EXPERT SYSTEM IN CHEMICAL SYNTHESIS WRITTEN IN APL2/PC

By Juan Rojas, Pilar Rodriguez and Manuel Alfonseca
IBM Madrid Scientific Center
Paseo de la Castellana, 4
28046 Madrid (SPAIN)

José Ignacio Burgos
Institut Quimic de Sarrià
08017 Barcelona (SPAIN)

INTRODUCTION

Chemical synthesis is a complicated process that uses a large set of rules, well known by the experts in the field. A correct synthesis is a sequence of chemical reactions, selected from those included in a catalog, that must be applied to certain "initial products" to obtain a given "goal product". The catalog must also contain information that will be used to deduce whether each reaction will be applicable or not in a particular situation.

The number of chemical species is huge, and each one can be obtained as the result of many reactions. Therefore, the number of possible synthesis paths of a single family of compounds is so large that a human being is easily overwhelmed. This is specially applicable to the synthesis of organic products.

Not all the synthesis paths are equivalent to each other. Industry wants to use, if possible, the cheapest procedure, or the easiest one under certain circumstances. The decision on which synthesis path to use depends on the available instrumentation and chemical reactors, the chemical products and reactants that can be bought in the market and their prices.

Unfortunately, the above mentioned conditions (simplicity and economy) are not always compatible. In that case, it is important to have criteria to choose the synthesis path most appropriate for each particular case. However, a global evaluation of the different possibilities requires a clear idea of the possible branching in the process of synthesis, for a single step in a procedure may make the whole impracticable. The automation of part or all of the design of chemical synthesis is thus recommendable to make sure that all possibilities are being considered, so that a good (but rare) synthesis path be not abandoned in favor of others, more typically used, but less efficient.

Since more than twenty years ago, different attempts have been made to use computers in Chemistry. Initially, chemical data bases were developed to help the chemist in the decision process, without any automation. However, these data bases grew in size and complexity to such an extent that other solutions had to be found.

Artificial intelligence was soon applied to Chemistry. In fact, the first expert system in history (DENDRAL, built by E. Feigenbaum and J. Lederberg, see reference 1) was a chemical system, although it has nothing to do with synthesis. DENDRAL is able to find the three-dimensional structure of organic molecules using data obtained from spectroscopic analysis (mass spectrograms and the like).

The first attempts to apply artificial intelligence to chemical synthesis were LHASA (Logic and Heuristic Applied to Synthesis Analysis), developed by J. E. Corey in the University of Harvard; SECS (Simulation and Evaluation of Chemical Synthesis), by W. T. Wipke in Princeton University; SYNCHEM (Synthesis Program) by H. Gelernter in Stony Brook; and PASCOP (Programme d'Aide a la Synthèse en Chimie Organique et Organo-Phosphorée) by C. Laurenço and G. Kaufman (see references 2 to 5). Some of these provide interactive graphic processors and include large knowledge bases about the most typical chemical reactions and their fields of application. The synthesis process itself is usually assisted, in the sense that it is the user who decides the synthesis path to follow among the different possibilities presented by the program.

AN EXPERT SYSTEM IN CHEMICAL SYNTHESIS

In 1986, the Scientific Center of IBM in Madrid started a joint project with the Institut Quimic de Sarrià. The objective of this project was the construction of an expert system that would be able to perform the completely automated design of the quasi-optimal synthesis paths to obtain certain concrete products. The system would be immediately applied to chemical education at the university level.

The expert system should comply with the following set of boundary conditions:

- Since the end users would be students and professional chemists, they could not be assumed to have any knowledge (or practice) of Computer Science.
- The result of the project will be used by students in a large number of workstations. In practice, this means that the system should be executable in personal computers.
- The exchange of information with the users should be done in an attractive way, similar to what is usually done in Chemistry. This means that the system should be provided with an interactive graphics editor, able to accept and view the graphic representation of organic molecules.
- Since artificial intelligence techniques seemed appropriate, the system should be able to represent and access a set of knowledge rules, and use them for automatic logic processing.
- The system should also have access to numeric computing techniques, to be able to apply optimization procedures based on economic data.
- Finally, and since chemical molecules have been traditionally represented in computers by means of a numeric matrix defining the connectivity between the different atoms or groups of atoms that make up the molecule, the system used should be able to manage this type of data structures with ease and speed.

All these considerations took us to the following decisions: The system would run in IBM Personal Computers. The tool selected to build the system was APL2/PC (reference 6). The language is very powerful and is able to work easily with matrices, embodies many facilities for numeric computing, and incorporates an interactive graphic processor (AP206) as well as a logic auxiliary processor (AP998) that makes it possible to perform logic processing with rules written in quasi-natural language.

SELECTION OF THE FIELD OF APPLICATION

In a first phase, the project would be restricted to a relatively compact, sufficiently wide subset of the organic chemical species. In future phases, the system could be extended to other families of compounds. The subset chosen was the family of the derivatives of benzene.

Benzene is a simple organic molecule with a hexagonal structure. There is a carbon atom in each vertex of the hexagon, each of which has a remaining free link that can be connected to other atoms or groups of atoms. In benzene, all six links are linked to hydrogen atoms (see figure 1).

```
      C-H
     //  \
 H-C      C-H
   |      ||
 H-C      C-H
     \\  /
      C-H
```

The family of derivatives of benzene contains all those substances that may be obtained from benzene by substitution of one or more hydrogen atoms by other atoms or groups of atoms. For example, figure 2 represents the formula of acetil-salicylic acid, an analgesic used in several commercial products.

```
      C-H
     //  \
 H-C      C-COOH
   |      ||
 H-C      C-O-CO-CH3
     \\  /
      C-H
```

It can be seen that it differs from benzene because one of the hydrogens has been replaced by the group -COOH and another one by the group -O-CO-CH3.

The synthesis system we have built accepts nineteen different atoms or groups of atoms that may be connected to the six carbon links in all the possible ways. This means that the theoretical number of possible compounds is very large (several millions). Many of them have a considerable chemical importance. However, not all the possible formulae correspond to real compounds, for some chemical restrictions forbid their existence.

A derivative of benzene may be transformed into another by means of certain reactions that replace one atom or group of atoms by another. The "reaction catalog" included in the system contains information about twenty six different reactions.

SYSTEM STRUCTURE

The system is divided into two workspaces. The first one is the interactive graphics editor, which accepts the description of the desired goal molecule in graphic form and converts it into the internal representation appropriate for calling the second workspace (the synthesizer).

The graphic editor makes it possible to define molecules by drawing them on the screen in the same way as if they were being drawn on paper. The editor allows the user to move the cursor along the screen, to draw there a complex chemical structure by pressing a single key, to link different molecules or to replace atoms or groups of atoms in previously defined molecules. After the goal molecule has been introduced, the synthesizer may be invoked explicitly from the editor workspace.

In the case of the family of derivatives of benzene, all of which have a regular structure, the synthesis process can be invoked directly at the synthesizer workspace, without the graphics editor, using a reduced symbolic notation where just the six groups of atoms linked to the hexagonal structure are specified. The results can also be viewed using this notation.

The synthesizer reads the basic data from a file. This file can be prepared outside APL2 by means of any external editor. The first time it is used, the information is read into the workspace and a new file is created, that contains the same data in a compact APL2 structure. This makes subsequent reads of the basic data much faster. Different files of basic data may be used with the system. In this way, the user may decide to restrict a given synthesis to a certain set of basic products or to certain reactions.

The information in the file is divided into three sections:

1. A list of atoms or groups of atoms (radicals) that may be linked to the carbon atoms in the hexagonal ring of benzene. This information is read as a two column general matrix, where each row represents a radical, the first column contains their symbolic representation (e.g. CL for chlorine, NH2 for amine, etc), and the second column contains their corresponding chemical activities.
2. A list of reactions able to replace one radical linked to the benzene ring by a different one. This information is read as a six column general matrix, where each row represents a reaction, the first column contains the name of the reaction, the second column is the radical to be replaced, the third column is the new radical after the reaction has taken place, the fourth and fifth columns are numeric data that make it possible to estimate the cost of the reaction per mol of goal product. Finally, the sixth column contains some chemical restrictions that may be applied to each reaction. The number of these restrictions depends on the actual reaction, therefore each element in this column is an APL2 general array.
3. A list of basic products. This information is read as a two column general matrix, where each row represents a basic product, the first column contains the chemical description of the product, and the second column its price. Since a product can be completely defined by a simple enumeration of the six radicals connected to the benzene ring, in an appropriate order, the basic products, as well as the goal molecule or those appearing in the process of synthesis may be internally represented by a vector of six numbers, each number being the row index of the corresponding radical in the list of radicals. This simple representation is possible in this case, due to the extreme regularity of the structure of the family of compounds accepted by this version of the system. A possible extension to other families would probably require a more complicated internal representation (such as a connectivity matrix).

Since a given derivative of benzene may be described in several ways (up to twelve) depending on the starting position in the ring and the direction of displacement along the ring, the system includes a normalizing procedure that reduces all of them to one unique internal representation: the one that has in the first positions those radicals with the largest index to the list of radicals indicated above.

THE SYNTHESIS PROCESS

When the user wants to obtain the synthesis of a given product, the system determines the set of chemical reactions that produce it at the minimum cost, using the following information contained in a knowledge base:

- A list of basic products and their prices.
- A catalog of reactions, with information on starting and ending products, and the cost per mol.
- A set of restrictions or rules to decide when the reactions are not chemically possible.

Starting with this information, the program deduces the possible precursors to the goal product and estimates the prices it would have if it were obtained from each of them. In this way, the system is able to select the optimal path from the point of view of cost.

When the price of a precursor is unknown, the system takes this precursor as an intermediate goal and starts recursively the process of synthesis. The computation ends when a product of known price is found. This procedure can be considered as a "chemical backtracking", and gives rise to an exploration tree that can have a large number of branching nodes. The chemical restrictions, however, prevent the exploration of those branches in the tree that do not have a chemical meaning. An additional mechanism allows the system to detect and eliminate those paths that contain sequences of precursors that repeat in cycles.

In any case, the number of subsequent precursors to be analyzed is usually very large. The system performs a depth-first search, where the maximum depth of a given search is limited by a global variable that defines a maximum cost for the goal. This variable receives a small initial default value. If no answer could be found with a given value of the maximum cost, the system increases the limit (without exceeding a certain maximum value) and tries again. The user may change the default value of the initial maximum cost and the increment.

To increase the performance of the system, the program is able to learn. Every time it has computed an intermediate product, the corresponding information is added to the data base. In this way, future consultations that go through the same intermediate products will be much faster.

To increase the performance of the system, it has been designed in such a way that it is possible to obtain quickly a synthesis path (that may not be optimal) and, if a better result is desired, the user may tell the system to find other (better) alternatives. It is also possible to start an exhaustive search that tries to find an optimal solution.

The user may also specify additional restrictions, such as the exclusion of one or more products from the synthesis path, or the selection of a specific cost for one or more intermediate products. The combined and repeated application of these restrictions makes it possible to obtain many different synthesis paths for a single final product.

EXAMPLE

The following is an actual example of the use of the synthesizer. APL2/PC must have been invoked with the AP210 auxiliary processor. If the graphics editor will be used, AP206 and AP440 should also be active. The synthesizer is loaded in the usual way:

```
      )LOAD SINTESIS
```

We can now start a process of synthesis by invoking the synthesizer directly:

```
      OPTIMIZE 'OH COOR H H H H'
```

The goal molecule selected is used in a well-known liniment. After executing the preceding function, we will see the following in the screen:

```
OPTIMIZE 13

OPTIMIZATION. STEP  1
---------------------

OBTAIN 13

 13 = COOR  OH    H     H     H     H       UNKNOWN

METHOD 13

 13 = COOR  OH    H     H     H     H       UNKNOWN
```

In the first step, the system could not find any synthesis procedure that complies with the initial cost limit. Therefore, function OPTIMIZE increases automatically the cost limit to explore the synthesis tree with a greater depth and retries the search ...

```
OPTIMIZATION. STEP  2
---------------------

OBTAIN 13

 13 = COOR  OH    H     H     H     H       UNKNOWN

METHOD 13

 13 = COOR  OH    H     H     H     H       297.06
 14 = COOH  OH    H     H     H     H       230.05
 26 = COR   COOH  H     H     H     H       158.20
  6 = COOH  H     H     H     H     H       114.00
```

In the second try, the system has found the best synthesis path to produce the desired goal molecule. The starting basic product would be product number 6. The synthesis path would go through intermediate products 26 y 14. These products appear in reverse order, i.e. the first one in the list is the final goal, and the last one is the starting product. The numbers in the last column represent the prices corresponding to this synthesis path.

Next, function OPTIMIZE calls function REACTIONS to produce a more complete listing of the reactions that would be used in the synthesis path just found.

```
REACTIONS 13

 13 = COOR  OH    H     H     H     H       297.06
COOH IN POSITION 1 STERIFIC    REACTION  OBTAINED FROM 14

 14 = COOH  OH    H     H     H     H       230.05
COR  IN POSITION 2 BAEYER_VILL REACTION  OBTAINED FROM 26

 26 = COR   COOH  H     H     H     H       158.20
H    IN POSITION 1 F_C_ACIL    REACTION  OBTAINED FROM 6

  6 = COOH  H     H     H     H     H       114.00
BASIC PRODUCT.
```

After this execution has finished, we may want to save the experience gained by the system (including data about products encountered in the synthesis procedure that never came to be part of the final synthesis goal) so as to be able to use that experience in future syntheses. This is done in the following way:

```
      SAVE EXPERIENCE
```

Let us assume that we don't want the synthesis path to go through product COR COOH H H H H (number 26 in the preceding sequence). In that case, we will write:

```
      EXCLUDE 'COR COOH H H H H'
```

We can now invoke again function OPTIMIZE to obtain a synthesis path that does not use the indicated product:

```
      OPTIMIZE 'OH COOR H H H H'
OPTIMIZE 13

OPTIMIZATION. STEP  1
---------------------

OBTAIN 13

 13 = COOR  OH    H     H     H     H       UNKNOWN

METHOD 13

 13 = COOR  OH    H     H     H     H       UNKNOWN

OPTIMIZATION. STEP  2
---------------------

OBTAIN 13

 13 = COOR  OH    H     H     H     H       UNKNOWN

METHOD 13

 13 = COOR  OH    H     H     H     H       UNKNOWN

OPTIMIZATION. STEP  3
---------------------

OBTAIN 13

 13 = COOR  OH    H     H     H     H       UNKNOWN

METHOD 13

 13 = COOR  OH    H     H     H     H      414.23
 15 = COOR  NH2   H     H     H     H      346.48
 37 = NO2   COOR  H     H     H     H      286.17
 74 = NO2   COOH  H     H     H     H      220.58
  6 = COOH  H     H     H     H     H      114.00
```

We have thus obtained a different (longer) synthesis path for the same product. Function REACTIONS would give us further information about the procedure. The excluded product can again be included in the following way, to ' restore the initial situation:

```
      INCLUDE 'COR COOH H H H H'
```

CONCLUSION

The expert system in chemical synthesis has been programmed successfully, using APL2. This first version of the system did not require, after all, the logic auxiliary processor, and could make use of a simplified internal representation of molecules. The synthesizer can be extended in the future to other families of chemical products without a great difficulty. In particular, the graphics editor already supports a very general set of products (much larger than the synthesizer) and would not need to be extended.

REFERENCES

1. E.A. Feigenbaum, B. Buchanan and J. Lederberg, *On Generality and Problem Solving: A Case Study Using the DENDRAL Program*, **Machine Intelligence**, 6:165, 1971.
2. E.J.Corey and W.T.Wipke, *Computer Assisted Design of Complex Organic Synthesis*, **Science**, 5:351, 1969.
3. W.T.Wipke, H.Braun, G.Smith, F.Choplin and W.Sieber, *SECS - Simulation and Evaluation of Chemical Synthesis: Strategy and Planning*, In **Computer-assisted Organic Synthesis**, edited by W.T.Wipke and W.J.House, American Chemical Society, Washington D.C., p. 97, 1977.
4. H.L.Gelernter, A.F.Sanders, D.L.Larsen, K.K.Agarival, R.H.Boivie, G.A.Spritzer and J.E.Scarleman, *Empirical explorations of SYNCHEM*, **Science**, 197:1041, 1977.
5. G. Kaufman y C. Laurenço, *La sintesis quimica por ordenador*, **Mundo Cientifico**, 6:653, 1981.
6. *APL2 for IBM Personal Computer*, IBM Corp., Program number 5799-PGG, PRPQ RJB411, Part No. 6242036.