Ordinateur et synthèse organique
Fichiers de produits commerciaux

par A. La Tela *, R. Barone, M. Chanon et J. Metzger
(Centre de calcul et I.P.S.O.I., Université d'Aix-Marseille III, rue Henri-Poincaré, 13013 Marseille)*

Un précédent travail (1) nous a permis de résoudre quelques-uns des problèmes posés par l'utilisation de l'ordinateur en synthèse organique, à savoir : définition d'une stratégie type de synthèse; représentation des molécules; représentation des réactions.
La stratégie développée pour résoudre les problèmes de synthèse est celle indiquée par E. J. Corey (2) : partant du produit à synthétiser on remonte, étape par étape, jusqu'aux produits de départ, dans un sens inverse à celui de la synthèse telle qu'elle est menée au laboratoire. Le graphe de la figure 1 illustre ce processus.

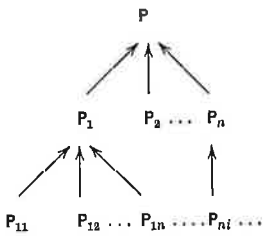
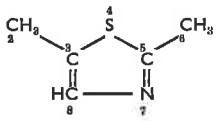


Figure 1. Stratégie développée. P est le produit à synthétiser, P_1, P_2, P_n, sont les précurseurs de P.

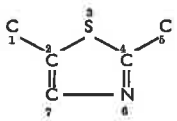
Le programme actuel est capable de présenter des solutions suivant ce schéma. Le problème est considéré comme résolu lorsqu'on aboutit à des produits commerciaux. Bien entendu l'organicien ne peut connaître tous ces produits, aussi nous paraît-il important d'introduire en mémoire d'ordinateur l'ensemble des produits commerciaux, et de pouvoir déterminer, parmi les intermédiaires proposés, ceux qui sont disponibles. Ceci permettra d'arrêter les chemins de synthèse correspondants. De plus, en introduisant le prix des produits, nous aurons une base d'estimation du coût de chaque synthèse. Cette estimation est essentielle pour doubler la discrimination purement chimique d'une discrimination économique.

La figure 2 montre une molécule codée dans le système utilisé. La mise en mémoire de tous les composés sous cette forme est onéreuse, car chaque molécule occupe une place importante. Un modèle plus compact s'imposait. Comme nous utilisons des tables de connectivité, la solution la plus simple à mettre en œuvre est celle décrite par H. L. Morgan (3). La figure 3 donne un exemple de cette représentation dans sa forme développée et dans sa forme linéaire telle qu'elle est enregistrée en mémoire d'ordinateur.
Des sous-programmes permettent le passage réciproque de la forme développée à la forme compacte, et de la forme compacte à la forme linéaire.
Cette représentation est fonction de la numérotation des atomes, aussi, est-il fondamental d'avoir une numérotation unique. Un sous-programme utilisant la technique proposée par Morgan (3) nous permet d'établir pour toute molécule une représentation unique.



N° *i*	Nature X	Lié aux atomes n° *j* *k* *l* *m*	Nature des liaisons *t* *u* *v* *w*
1	H		
2	C	3 1 1 1	1 1 1 1
3	C	2 4 8	1 1 2
4	S	3 5	1 1
5	C	4 6 7	1 1 2
6	C	5 1 1 1	1 1 1 1
7	N	5 8	2 1
8	C	3 7 1	2 1 1

Figure 2. Représentation d'une molécule dans le système utilisé. Commentaires : l'atome n° 5 (*i*) est un atome de carbone (X), il est lié à l'atome n° 4 (*j*) par une liaison simple (*t*), à l'atome n° 6 (*k*) par une liaison simple (*u*) et à l'atome n° 7 (*l*) par une double liaison (*v*). Tous les atomes d'hydrogène ont le numéro 1.



N°	Nature	Lié au n°	Liaison cyclique	Nature de la liaison
1	C	0		0
2	C	1		1
3	S	2		1
4	C	3		1
5	C	4		1
6	N	4		2
7	C	2		2
			6 7	1

Sous forme linéaire, cette molécule est représentée :
0 1 2 3 4 4 2 - 6 7 - 0 1 1 1 1 2 2 1 - C C S C C N C

Figure 3. Représentation de la molécule de la figure 2 dans le système compact de Morgan. Ici les hydrogènes ne sont pas notés.

Soit un produit P à synthétiser. L'ordinateur propose un schéma de synthèse : P peut être obtenu à partir de P*i*. Le problème est de déterminer si P*i* est commercial, c'est-à-dire de rechercher le produit dans une vaste collection de composés. Pour atteindre cet objectif deux problèmes étroitement liés sont à résoudre : tout d'abord un problème de stratégie : comment atteindre le plus rapidement possible tel produit dans la bibliothèque des produits commerciaux? Ce qui entraîne un second problème, de classement, d'organisation des données.

Pour résoudre le premier problème nous disposons de la table de connectivité, mais l'utilisation exclusive de cette table prendrait trop de temps, car il faudrait comparer la table décrivant le produit avec toutes les tables. Comme il est possible, à partir de la table de connectivité sous sa forme développée, de calculer la formule brute, nous nous servirons de celle-ci pour effectuer un premier tri parmi les produits commerciaux. La recherche d'un produit se fera donc en deux étapes :
1. Recherche sur les formules brutes.
2. Recherche sur les tables de connectivité.
En définitive nous coderons les éléments suivants :
a. La formule brute et la table de connectivité pour la recherche des produits.
b. Le nom du produit, une référence et le prix moyen qui seront imprimés.

Le classement à mettre en œuvre doit être simple et doit permettre une recherche rapide. Nous avions deux possibilités :
1. Enregistrer les produits par ordre croissant de formule brute, et calculer la zone à tester pour chaque formule brute. Cela correspond à une méthode de classement utilisée dans certains catalogues de produits commerciaux. Cette méthode est intéressante du point de vue interrogation, mais présente un inconvénient pour la création des fichiers. En effet, comme nous sommes amenés à introduire régulièrement de nouveaux composés, il fallait prévoir des programmes permettant un interclassement automatique des produits, ce qui aurait entraîné une perte de temps importante.

2. Pour la facilité de création des fichiers il nous a paru préférable de pouvoir introduire les produits dans un ordre quelconque, tout en conservant une interrogation partielle et rapide.
Pour cela, nous avons créé trois fichiers principaux :
Le fichier 1 renferme les tables de connectivité,
Le fichier 2, les formules brutes,
Le fichier 3, les noms, les références et les prix.

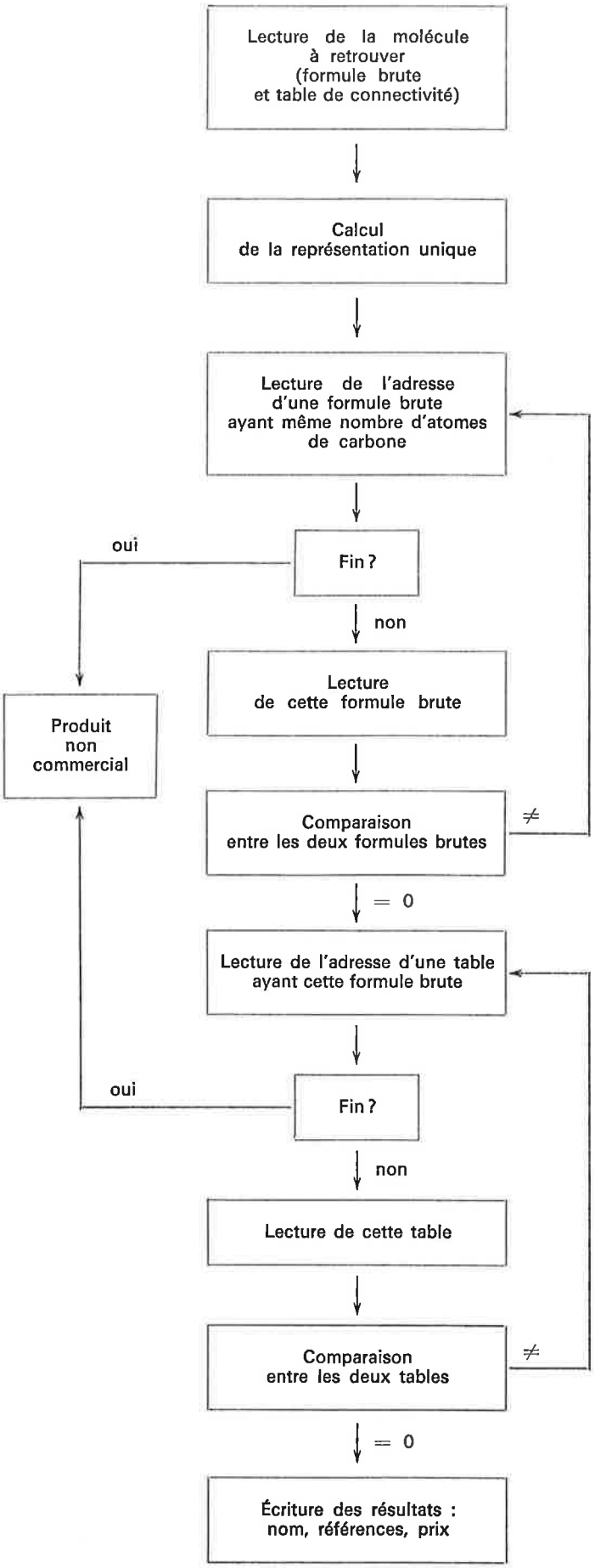


Figure 4. Organigramme général du programme.

et des fichiers secondaires qui établissent des relations entre ces trois fichiers.
La séparation des informations en trois fichiers permet un gain de place en évitant la répétition des formules brutes, elle permet aussi certaines facilités pour l'interrogation et l'édition des résultats.
Du fait des isoméries il n'y a pas une correspondance directe entre les enregistrements du fichier 2 et des fichiers 1 et 3. En effet, une même formule brute n'apparaît qu'une seule fois dans le fichier 2. Lors de l'enregistrement, les adresses (du fichier 1) des produits ayant même formule brute sont relevées et inscrites dans les fichiers secondaires 6 et 7. Le fichier 6 donne la première adresse, et le fichier 7 la suite des adresses.
La recherche d'un produit débute par la recherche de la formule brute. Le nombre des formules brutes est élevé, aussi une fragmentation s'impose afin de permettre une recherche partielle. Le fichier 2 sera découpé en fonction du nombre d'atomes de carbone. Lors de l'enregistrement, les adresses (du fichier 2) des formules brutes ayant même nombre de carbone sont relevées et inscrites dans les fichiers secondaires 4 et 5. Le fichier 4 donne la première adresse, et le fichier 5 la suite des adresses.

Montrons sur l'exemple suivant les correspondances qui existent entre ces différents fichiers. Soit à enregistrer les produits :

1. Acétamide ..... C_2H_5NO ... Table de Connectivité 1
2. Imidazole ..... $C_3H_4N_2$ ... TC 2
3. Oxyde d'éthylène ..... C_2H_4O ... TC 3
4. N-méthyl formamide ... C_2H_5NO ... TC 4
5. Acide acétique ..... $C_2H_4O_2$ ... TC 5
6. Acétone ..... C_3H_6O ... TC 6
7. Éthanal ..... C_2H_4O ... TC 7
8. Alcool allylique ..... C_3H_6O ... TC 8
9. Propanal ..... C_3H_6O ... TC 9
10. Formiate de méthyle ... $C_2H_4O_2$ ... TC 10
11. Oxyde de propylène ... C_3H_6O ... TC 11
12. Pyrazole ..... $C_3H_4N_2$ ... TC 12

Les fichiers se présentent ainsi :

F 1
1. TC 1
2. TC 2
3. TC 3
4. TC 4
5. TC 5
6. TC 6
7. TC 7
8. TC 8
9. TC 9
10. TC 10
11. TC 11
12. TC 12

F 2
1. C_2H_5NO
2. $C_3H_4N_2$
3. C_2H_4O
4. $C_2H_4O_2$
5. C_3H_6O

F 3
1. Acétamide
2. Imidazole
3. Oxyde d'éthylène
4. N-méthyl formamide
5. Acide acétique
6. Acétone
7. Éthanal
8. Alcool allylique
9. Propanal
10. Formiate de méthyle
11. Oxyde de propylène
12. Pyrazole

F 4
1.
2. 1
3. 2

F 5
1. 3
2. 5
3. 4
4. 4
5. 5

F 6
1. 1
2. 2
3. 3
4. 5
5. 6

F 7
1. 4
2. 12
3. 7
4. 4
5. 10
6. 8
7. 7
8. 9
9. 11
10. 10
11. 11
12. 12

Les chiffres de gauche représentent les numéros d'enregistrement, et les chiffres de droite la valeur de ces enregistrements.

Soit à rechercher un produit de formule brute C_3H_6O.
La formule brute est de 3 atomes de carbone, dans F 4 le troisième enregistrement donne le numéro 2, la machine se reporte au deuxième enregistrement de F 5 qui renvoie au numéro 5. C'est-à-dire que les formules brutes de 3 atomes de carbone sont enregistrées dans le fichier 2 aux numéros 2 et 5. Les comparaisons sont effectuées. La formule C_3H_6O existe, c'est la cinquième. La machine se reporte au cinquième enregistrement de F 6, qui donne le numéro 6. Reporté au fichier 7 le sixième enregistrement renvoie aux numéros 8, 9, 11. Dans le fichier 1, les molécules de formule brute C_3H_6O sont enregistrées aux numéros 6, 8, 9, 11.

Comme on peut le constater, cet ensemble de fichiers en cascade permet d'atteindre très rapidement les composés ayant même formule brute que le produit recherché. Ces fichiers secondaires, qui ne comportent qu'un mot par enregistrement, sont peu coûteux du point de vue place et sont simples à générer.
L'organigramme général de ce programme est présenté à la figure 4. Le programme a été écrit en FORTRAN IV pour une IBM 1130 de 16 Kmots, nous pouvons coder environ 25 000 composés sur deux unités de disque de 512 000 mots chacune. Le temps mis pour retrouver une molécule en triant 100 formules brutes et 25 tables de connectivité, est de 10 secondes.
Ce programme est à inclure dans notre programme de synthèse, cependant sa conception est tout à fait générale, et il peut donc être utilisé pour tout problème bibliographique faisant intervenir des composés chimiques.

Bibliographie

(1) R. Barone, M. Chanon et J. Metzger, *Rev. Inst. Fr. Pétrole*, 1973, **5**, 771.
(2) E. J. Corey et W. T. Wipke, *Science*, 1969, **166**, 178.
(3) H. L. Morgan, *J. Chem. Doc.*, 1965, **5**, 107.