REFERENCES AND NOTES

(1) "Thermophysical Properties Research Literature Retrieval Guide", Supplement I, Y. S. Touloukian, J. K. Gerritsen, and W. H. Shafer, Ed., IFI/Plenum, New York, N.Y., 1973.
(2) National Science Foundation, "Program Solicitation—Improved Dissemination and More Productive Use of Scientific and Technical Information" (NSF 75-23), Washington, D.C., 1975, pp 2–3.
(3) R. Bemer and A. Shriver, "Integrating Computer Text Processing with Photocomposition", *IEEE Trans. Prof. Commun.*, PC-16, 92-7 (1973) (Special Issue: Record of the Conference on the Future of Scientific and Technical Journals).
(4) J. Phillips, "Integrating Computer Processing with Photocomposition", *IEEE Trans. Prof. Commun.*, PC-16, 165 (1973) (Special Issue: Record of the Conference on the Future of Scientific and Technical Journals).
(5) R. Lerner, "The Use of the Computer in Converting Primary Information", *J. Chem. Doc.*, **14**, 112–4 (1974).
(6) Aspen Systems Corporation, Editorial Processing Centers: Feasibility and Promise (Report on NSF Contract C769), Aspen, Germantown, Md., 1975.
(7) S. Rhodes and H. Banford, "Editorial Processing Centers: a Progress Report", *Am. Sociologist*, **11**, 153–9 (1976).
(8) Chemical Abstracts Service, Features of the American Chemical Society's Primary Journal Composition System, American Chemical Society, Columbus, Ohio, 1976.
(9) R. May, Ed., "Automated Law Research", American Bar Association, Chicago, Ill., 1973, pp 40, 62, 104.
(10) "United States Air Force Judge Advocate General Law Review" (Special Issue on LITE), 1966.
(11) "United States Air Force Judge Advocate General Law Review" (Special Issue on LITE), 1972.
(12) J. O'Connor, "Retrieval of Answer-Sentences and Answer-Figures from Papers by Text Searching", *Inf. Process Manage.*, **11**, 155–64 (1975).
(13) F. Roehlich et al., "Surface Tension of Four Alkali Metals to 1000 °C", *J. Chem. Eng. Data*, **13**, 518–21 (1968).
(14) For compactness, TP also uses numerical codes for substance names. Thus TP entries *appear* somewhat different from the entry given here.
(15) Since the computer procedures studied were "manually" simulated, as already noted, the last three sentences of this paragraph are rather "mechanomorphic" in style, but that style permits quick understanding of the computer procedures involved and will be used throughout.
(16) D. Lapedes, Ed., "McGraw-Hill Dictionary of Scientific and Technical Terms", McGraw-Hill, New York, N.Y., 1974.
(17) H. Bennett, Ed., "Concise Chemical and Technical Dictionary", Chemical Publishing Co., New York, N.Y., 1974.
(18) G. Hawley, Ed., "Condensed Chemical Dictionary", Van Nostrand-Reinhold, New York, N.Y., 1971.
(19) H. Gray and A. Isaacs, Ed., "A New Dictionary of Physics", Longmans, London, 1975.
(20) J. Grant, Ed., "Hackh's Chemical Dictionary", McGraw-Hill, New York, N.Y., 1969.
(21) A. Merriam, Ed., "A Concise Encyclopedia of Metallurgy", American Elsevier, New York, N.Y., 1965.
(22) If this section seems too concise, re-reading section III.E will provide sufficient context.
(23) I thank Preston Marshall, Center for Information Science, Lehigh University, for the specific computer cost and performance information given here.
(24) Stop-words include both function words such as "of" and "the" and general words such as "described" and "details".

An Algorithmic Computer Graphics Program for Generating Chemical Structure Diagrams

PAUL G. DITTMAR,* JOSEPH MOCKUS, and KATHYRN M. COUVREUR

Chemical Abstracts Service, P.O. Box 3012, Columbus, Ohio 43210

Received March 2, 1977

An algorithmic computer graphics program has been developed at Chemical Abstracts Service (CAS) which has the capability of transforming connection tables into chemical structure diagrams of a quality suitable to meet editorial proofing requirements at CAS. The program, which runs on an IBM 370/168 computer, begins with a connection table record and a reference file of basic chemical ring shapes. Then, by using a set of defined rules, the order of processing and the preferred placement of the structural pieces are systematically determined and the diagram is constructed. An automatic adjustment capability is available to handle situations where one structural fragment overlaps another within a given diagram.

INTRODUCTION

The CAS Chemical Registry System grew out of CAS staff research in the early 1960's in which an algorithm developed by DuPont was perfected for generating a unique and unambiguous computer-language description of the two-dimensional structure and stereochemical details of a chemical substance.[1] The Registry System uses this algorithm as the foundation for identifying chemical substances on the basis of their composition and structure and for linking their structural descriptions to the various names by which they are identified in the literature through unique computer-checkable Registry Numbers. Registry III went into operation early in 1974 as an extension to this system.[2] This extension expanded the system's chemical substance naming capability and made possible, for the first time, the re-creation algorithmically of structure diagrams from computer structure records. In Registry III the parent ring systems of a structure are identified separately by a modified form of computer-readable structure notation. These parent ring systems can thus be retrieved from a computer file to provide a starting point for recreating the structure.

The CAS Chemical Registry System utilizes a manual file of hand-drawn structural diagrams to support processing operations. [Note: Structures used throughout this paper are hand-drawn for illustrative purposes. Examples of actual chemical structures generated by the ASD program are given at the end of this paper (see Figure 19).] This file currently contains over 3.7 million entries. The ability to produce structural diagrams from the computer-readable Registry records minimizes the need to reference and maintain this extensive manual file.

When the system is fully operational, the ability to produce structure diagrams by computer will depend upon two complementary capabilities required to handle the full range of structures. Most structures will be automatically "constructed" by an Algorithmic Structure Display (ASD) program starting with information contained in the Unique Chemical Registry Records (UCRR's) of the CAS Registry File. For certain special classes of structures, this capability will be supplemented by a system which will cause structures to be "played back" from a file of display instructions recorded from input via an interactive graphics terminal.[3]

The purpose of this paper is to further describe the ASD program.

BASIC FEATURES OF ASD

The ASD program was designed with the intention of supporting the substance indexing operations within the CAS production system, primarily that of naming substances new to the CAS Chemical Registry System. A second objective

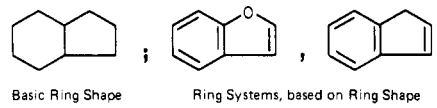


Figure 1. Basic ring shape.

of the program was to provide a base for identifying algorithmic display features that are unacceptable for use in printed publications and services.

The input to the ASD program consists of the connection table contained in the UCRR of the CAS Chemical Registry System. The ring systems contained in each structure are specifically identified by the Registry System within the UCRR. The ASD program uses a computer-readable authority file of basic ring shapes—the Ring Image File—in constructing the display. This ring image concept is a variation of the concept described by Zimmerman.[4] The ring images used by ASD are independent of explicit node and bond values. The Ring Image File contains basic geometrical data for each unique ring shape identified by the CAS Chemical Registry System for all ring systems contained in the Registry File (see Figure 1).

About 85% of the 3.5 million structures in the CAS Registry File contain one or more ring systems. Among these ring systems, there are approximately 17 000 unique ring shapes. By prestoring the coordinates for each unique ring shape, major problems usually encountered in determining coordinates for fused and bridged ring systems are eliminated, and standard shapes for all rings are achieved.

The ASD program uses the Graphical Data Structure (GDS) developed for CAS graphics processing. The GDS is a tree structure with a Cartesian coordinate system for graphical data.[5] Use of the GDS as the output form provides device independence in that ASD structure diagrams can be output on any device capable of general vector graphics (e.g., the Varian "dot" printer, the APS-4 photocomposer, or a graphics CRT terminal, but not a line printer); only small "driver" programs are device dependent.

The ASD program contains an overlap detection capability which determines if one part of a structure is drawn over another part[6] and an automatic adjustment capability to avoid such situations. A detailed discussion of this capability is provided below.

TECHNIQUES OF DISPLAY GENERATION

The ASD program consists of a number of procedures, each of which falls into one of five categories—Analysis, Ranking, Orientation, Drawing, or Positioning.

Analysis. Analysis consists of "unnormalizing" tautomer and alternating bonds, identifying shortcut groups, and identifying display components. The CAS Chemical Registry System normalizes tautomeric and alternating bond conditions so that the program can recognize when a structure has been input at different times with different structuring conventions. ASD regenerates single and double bonds from these normalized bonds according to the basic rules for CAS preferred nomenclature (see Figure 2).

The Registry System cites every nonhydrogen atom given in a structure in the UCRR record. Commonly recognized shortcut symbols are often used by chemists to highlight certain groups of atoms because of their frequency and because the atoms as a group play a role in the structure. ASD identifies many of these functional groups, alkyl groups, and in-line contractions and replaces the individual atoms with the commonly recognized shortcut symbols (see Figure 3). Replacing these atom groups by shortcut symbols eliminates many branching conditions from the display algorithm and makes the resulting display less complex.

The ASD program operates on small components of a

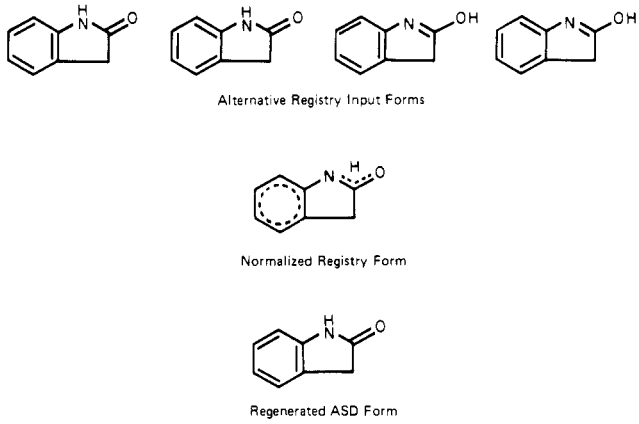


Figure 2. Example of normalized bonds.

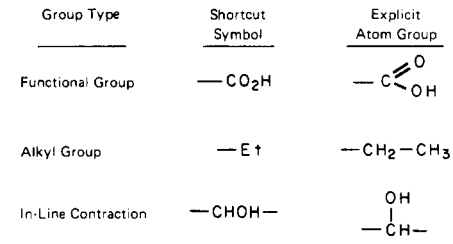


Figure 3. Shortcut examples.

structure and builds them into a display. To accomplish this, it is first necessary to identify the components: ring systems, acyclic center nodes, links, and chains. Rings systems are isolated by the Registry System and identified in the UCRR. The Ring Image File provides the necessary coordinate data, making any further analysis of ring systems unnecessary. [As new ring systems are identified during ongoing processing of the CAS Registration, the ring images are input and added to the Ring Image File through an interactive graphics terminal. Approximately 1200 such systems are identified each year.]

The acyclic portion of the structure does, however, require further analysis. This analysis consists of determining the linear strings of atoms to be treated as a unit by subsequent display operations. A linear string of acyclic atoms which contains a terminal atom (an atom with only one attachment) is called a chain; one with no terminal atom is called a link. A link or a chain may contain branch points (atoms with more than two attachments), but the link or chain containing the branch points may contain no more than two of these attachments. Each remaining attachment begins a new chain or link. Figure 4 illustrates these components.

Figure 5 identifies components of an acyclic structure, including the acyclic center node. This is the centermost node of the structure found by paring terminal nodes until only one (or two) nodes remain.

Every link and chain must be a substituent of another component. An order 1 link is a substituent of a ring system. An order n link, with $n > 1$, is a substituent of an order $n - 1$ link. An order 1 chain is a substituent of either a ring system or an acyclic center node. An order n chain, with $n > 1$, is a substituent of an order $n - 1$ link or chain. Determination of orders of links and chains is discussed in the next section.

Ranking. Ranking consists of determining a processing sequence for components. To accomplish this, a processing-order tree structure is constructed (see Figure 6). An order and a rank are determined for each component, where the order indicates the level in the tree structure and the rank indicates the sequence among components of equal order.

A subtree is constructed representing each ring system (or acyclic center node) where the root (order 0) of the subtree

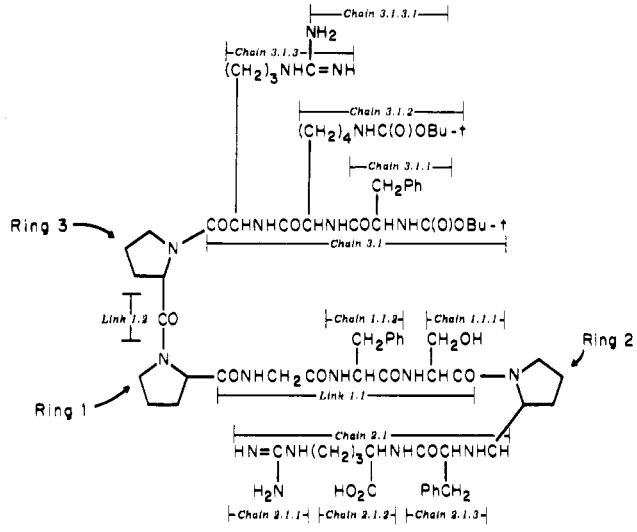


Figure 4. Components of a ring structure.

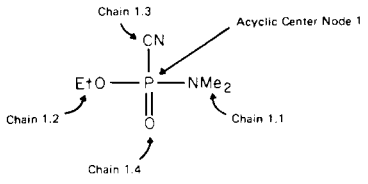


Figure 5. Components of an acyclic structure.

represents the ring system; order 1 branches represent link or chain substituents of the ring system; order 2 branches represent link or chain substituents of the order 1 link or chain; etc. Throughout the subtree, a link is considered a substituent of the lowest ranking ring system to which it is connected. (An order 1 link directly connects two ring systems, and an order n link connects a ring to an order $n-1$ link). Each ring system is a branch of the root of the processing order tree.

A ranking is assigned to the ring system subtrees such that the most central ring system is rank one; the next most central, rank two; etc. The most central ring system is identified by a method similar to selecting an acyclic center node—treating the ring systems as nodes and the links as bonds between the rings.

Ring system substituents are ranked such that links rank lower than chains. Each class is ranked further, based upon a sequence ordering through the quadrants of the ring system perimeter. Using the example in Figure 7, substituents are ranked beginning with those in the quadrant labeled 1 and continuing on around the perimeter in the sequence indicated by the arrows. Substituents of any internal ring nodes are ranked last.

The ranking of order n substituents begins with those furthest from the atom attached to the order $n-1$ link or chain and proceeds in sequence back to the attached atom (see Figure 8).

Orientation. Orientation is the process of determining the manner in which a component is to be drawn and subsequently positioned. Orientation depends upon how prior components in the subtree have been drawn. Ring system orientation is based upon the prestored basic ring shape. If alternative mappings are possible due to symmetry, then a mapping is selected such that (a) the node with the highest heteroatom (noncarbon) priority receives the lowest possible locant, (b) the set of hetero nodes receives the lowest possible set of locants, (c) the heteroatoms with the highest priority receive the lowest locants, and (d) the set of substituted nodes receives the lowest possible set of locants. The locants are manually assigned at the time the ring shapes are input and are stored with them in the Ring Image File. Figure 9 illustrates the selection of the mapping.

If the number one ranking ring system is monocyclic, the mapping will be rotated so that the largest, most troublesome substituent is in the right or upper right position (see Figure 10). For the number one ranking polycyclic ring system, no further orientation is required. For higher ranking ring systems, consideration must be made as to how the orientation is to be attached to a lower ranking ring system. The higher ranking ring system may be reflected about the X axis, the Y axis, or both in order to permit attaching it to the lower ranking ring (see Figure 11).

Link and chain orientation consists of determining how the link or chain is to be drawn relative to the component to which it is attached. The orientation specifies the bond angle from the component to the link or chain, the bond length, and the direction the link or chain is to be drawn away from the component.

The orientation data for a link are:

bond angle—for an order 1 link, the horizontal or vertical angle closest to the angle bisecting the ring system perimeter; for an order n link ($n > 1$), the horizontal or vertical angle perpendicular to the order $n-1$ link and away from the ring.

bond length—standard (the standard bond length is defined as a multiple of the character size).

drawing direction—whichever direction is away from the ring: up or down for a vertical bond angle, or right or left for a horizontal bond angle.

The orientation data for a chain are:

bond angle—for an order 1 chain, the angle bisecting the ring perimeter; for an order n chain ($n > 1$), the angle perpendicular to the order $n-1$ link or chain and away

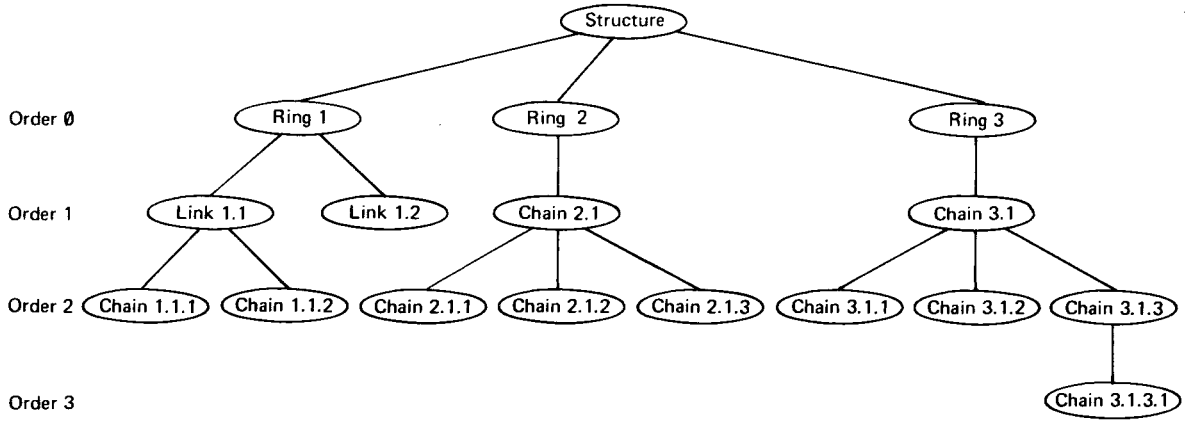


Figure 6. Processing order tree for the structure in Figure 4.

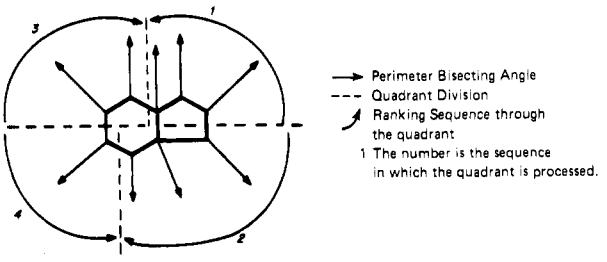


Figure 7. Ring substituent ranking.

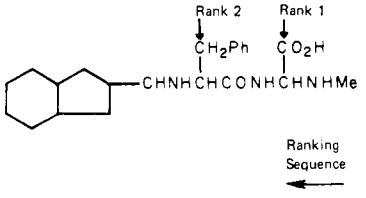


Figure 8. Ranking of (sub)chains or (sub)links.

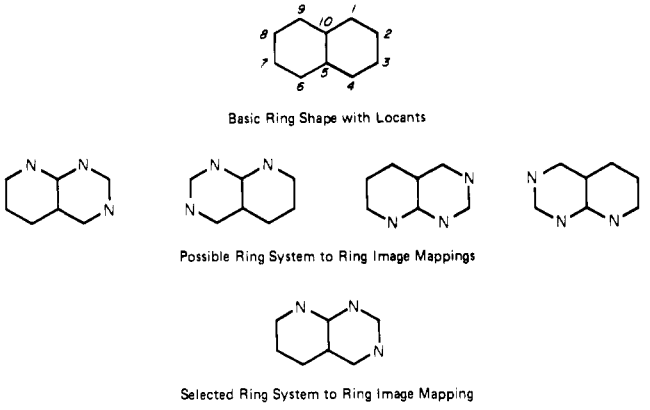


Figure 9. Ring system to ring shape mapping.

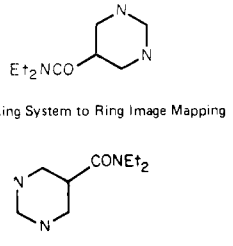


Figure 10. Ring mapping rotation.

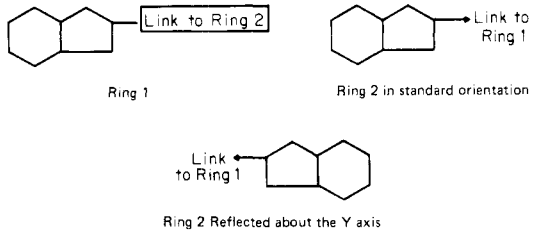


Figure 11. Ring shape reflection.

from the ring (or acyclic center node).

bond length—standard.

drawing direction—whichever direction is away from the ring (or acyclic center node): right or left.

Link and chain orientations are illustrated in Figure 12.

Drawing. Drawing a component consists of constructing that portion of the GDS which represents the component and determining coordinates for each atom and bond of the

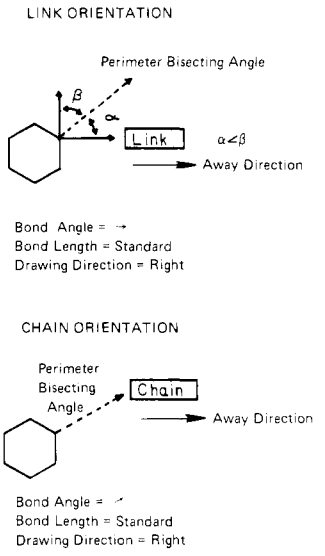


Figure 12. Link and chain orientation.

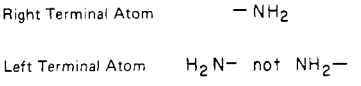


Figure 13. Inverted character symbols for terminal atoms where drawing direction equals left.

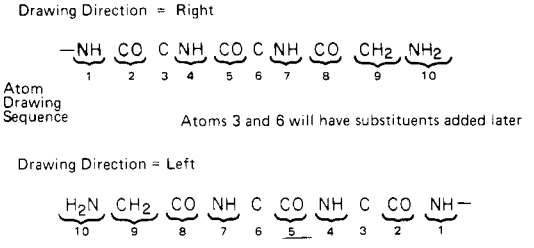


Figure 14. Drawing sequence for atoms of a link or chain.

component relative to its node block. Determining coordinates of the component node block relative to higher level (lower order) components is performed during the positioning phase.

Essentially, the GDS is a sequence of blocks linked together to form a tree. Each block may be one of three types. A node block serves to structure a picture. A branch block serves to connect a node to a node or a leaf to a node and, using a relative coordinate system, to specify the coordinates of the lower level node or leaf relative to the higher level node. The third type of block, the leaf block, contains the graphical data necessary to draw the picture on an output device. For ASD, each atom and each bond are represented by a set of character symbols or line vectors in one or more leaf blocks. Each leaf block is connected through a branch block to a node block which identifies an atom or bond. These atom and bond node blocks are again connected through branch blocks to a higher level node block which identifies the component (ring system, link, or chain). The component node blocks are similarly connected to higher level node blocks such that they form the same logical structure as the processing order tree described earlier.

For ring systems, the drawing consists of calculating atom characters and bond vectors for each atom and bond within the ring system. Coordinates are assigned to each based upon the coordinate from the basic ring shape (reflected as necessary). For links and chains, drawing begins with the base node (atom attached to the ring or the lower order link or chain). The next atom is drawn above, below, to the right, or to the left, as specified by the drawing direction, and the bond is drawn between the two atoms. Horizontal single bonds are implicit and are not drawn; all others are drawn explicitly.

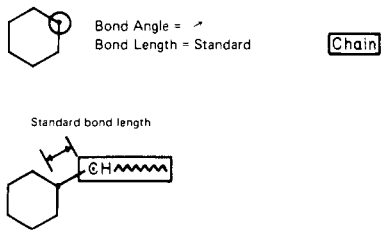


Figure 15. Positioning of a ring substituent.

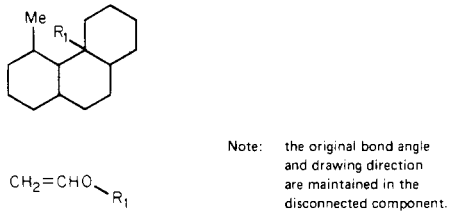


Figure 16. Example of overlap resolved by connectors.

The next atom in the link or chain is placed similarily. Character symbols for terminal atoms drawn to the left are inverted if necessary. Figure 13 illustrates the terminal node inversion. Figure 14 illustrates the atom drawing sequence for links and chains.

Positioning. Positioning consists, first, of determining coordinates for a subtree relative to the next higher level (father) of the subtree and, second, of detecting and resolving any overlap. A link or chain is positioned relative to its father using the bond angle and bond length specified at the time it was oriented. The connecting bond is then drawn (see Figure 15).

A rank 1 ring system is positioned at the coordinate system origin. A higher ranking ring system is positioned relative to the link of the lower ranking ring system to which it is attached.

Whenever positioning takes place, some portion of the subtree being positioned may overlap with previously positioned portions of the father's subtree. Positioning therefore checks for overlap using the detection algorithm discussed below in the next two paragraphs. If no overlap is detected, positioning is complete. If, on the other hand, overlap is detected, the subtree is repositioned by altering the bond angle and/or bond length specified at orientation time. If overlap still exists after a predefined sequence of bond angle and length changes, the bond is physically broken and shown via logical connector symbols R_i, $i = 1,2,\ldots$ (see Figure 16). The subtree is then positioned independently of its father at the origin of a separate coordinate system. This results in a series of "frames" or "windows" which can be positioned in a subsequent formatting stage.

The tree data structure representing a substance has hierarchical organization determined by properties of the chemical structure. The overlap detection algorithm is illustrated in Figure 17. To determine whether overlap exists between two components of a substance, rectangles are circumscribed about each component along with its subcomponents as represented by the subtree. The coordinates of the rectangles are then compared for overlap (intersection). If the rectangles overlap, the subtrees are then further examined, working down each subtree until no overlap of rectangles is found, in which case no overlap exists, or until the lowest level of both subtrees is reached, indicating that overlap does exist.

In order to determine whether overlap exists between the two ring substituents in Frame 1 (Figure 17), the coordinates of the rectangles around the substituents are compared (Frame 2). Since overlap exists at this level, the next level of the subtree for the first substituent is tried. Rectangle 1.1 does not overlap rectangle 2 (Frame 3) but rectangle 1.2 does

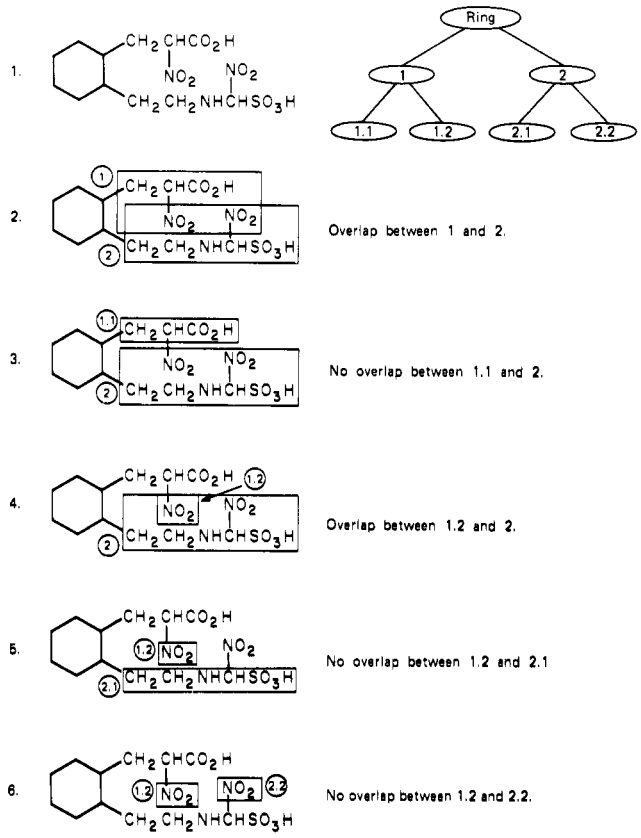


Figure 17. Overlap detection.

Step #	
1	Analyze Structure
2	Rank Rings*
	Do Until all Rings Processed
3	Select Unprocessed Ring
4	Orient Ring
5	Draw Ring
6	Rank Order 1 Substituents
	Do Until all Order 1 Substituents Processed
7	Select Unprocessed Order 1 Substituent
8	Orient Link/Chain
9	Draw Link/Chain
10	Rank Order 2 Substituents
	Do Until all Order n Substituents (n >1) Processed
11	Select Unprocessed Order n Substituent
12	Orient Link/Chain
13	Draw Link/Chain
14	Rank Order n + 1 Substituents
	End Do
	Do Until all Order n Substituents Positioned
15	Select Unpositioned Order n Substituent
16	Position Order n Substituent
	End Do
17	Position Order 1 Substituent
	End Do
18	Position Ring
	End Do
	End

*An acyclic center node is treated as a ring

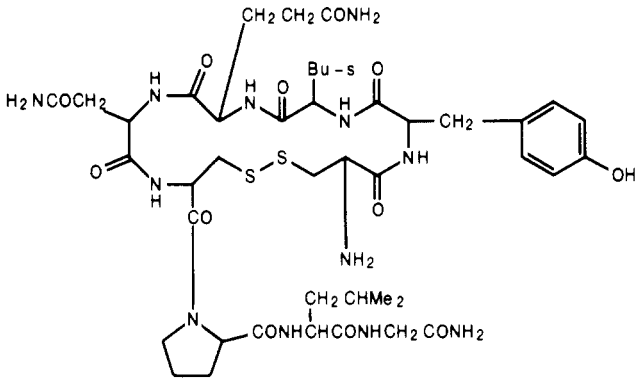
Figure 18. Algorithm structure.

(Frame 4). The next level of the second substituent is tried. Rectangle 1.2 does not overlap rectangle 2.1 (Frame 5) nor rectangle 2.2 (Frame 6). Therefore, there is no overlap between the two substituents. If overlap had been detected at this lowest level, corrective action would be necessary.

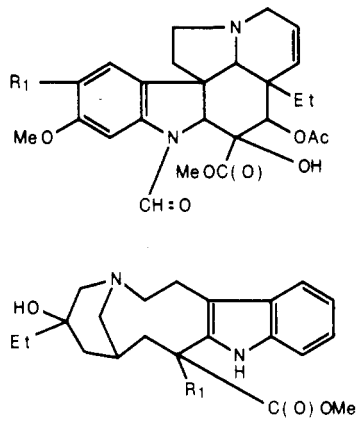
This overlap detection technique, which minimizes the number of items to be examined for overlap, is based upon general geometric principles and does not depend on any specific characteristics of chemical structure diagrams.

ALGORITHM STRUCTURE

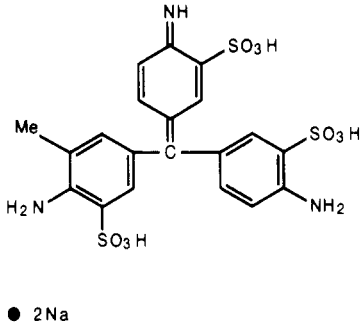
A preorder processing sequence[7] is used to scan the pro-



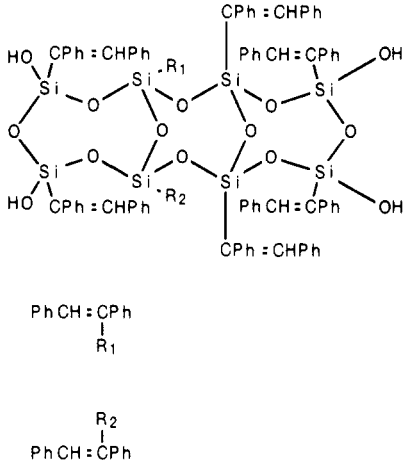
REG NO 50-56-6



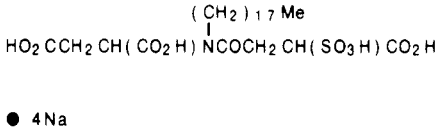
REG NO 57-22-7



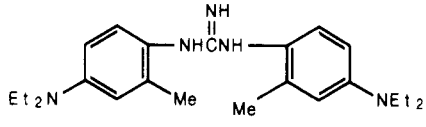
REG NO 3244-88-0



REG NO 57539-57-8



REG NO 38916-42-6



REG NO 50854-48-3

Figure 19. Examples of ASD output.

cessing order tree structure (shown in Figure 6) constructed as a result of the analysis step. [Preordering, as described by Knuth, is a means of traversing a tree by visiting each node of the tree exactly once. The order involved is (1) visit the root, (2) traverse the left subtree, and (3) traverse the right subtree.] Components are oriented and drawn while going down the tree and are positioned when coming back up. This means that a component is oriented and drawn before any of its substituents, but positioned after its substituents have been oriented, drawn, and positioned relative to it.

The structure of the algorithm is shown in Figure 18.

RESULTS

As a preliminary step in obtaining "live" pilot experience, the ASD program has been incorporated into the CAS Chemical Registry System for monitoring the contents of requested structure records. This monitoring was previously performed by obtaining connection table printouts of these records and then manually reconstructing the structure diagrams. This monitoring activity amounts to approximately 100 requests per day; therefore, the program has resulted in saving the intellectual and manual effort required for reconstructing the diagrams.

Limitations. This algorithm produces highly acceptable results for most types of compounds (for examples of ASD output see Figure 19). However, in the initial design and implementation of the algorithm, special considerations required for indicating stereo detail and for handling coordination compounds, polymers, and incompletely defined structures were deferred. Images for many of these structures can be generated but the acceptability of these images is entirely dependent upon their use. Therefore, the acceptability of these images cannot be predicted.

Table I. Summary of Operational Performance

Sample Size		
CAS File Size	3.5×10^6	
Sample Size	1581	.05% of CAS File
Samples Drawn	1362	86.1% of Sample
Samples Rejected by program	219	13.9% of Sample
Generation Timings	Time*	
Sample File	218 Sec.	
Per Structure	.138	

*IBM 370/168, Raw CPU. These timings represent only the algorithm and do not include the input or registration of the structures or formatting of the output.

Additionally, the procedures for normalizing tautomers, resolving overlap, and for handling substituents of nodes forming concavities within the perimeters of ring shapes have been defined to handle the more commonly occurring conditions and may fail in less common, more complex situations.

Operational Performance. Prior to the installation of the program a review was performed based upon both routine Registry input and structures specially selected to determine the algorithm utility. Table I summarizes the operational performance.

Qualitative results are less easily determined as they are a function of the structure diagrams' intended use. The file monitoring activity is much less demanding than the index nomenclature generation activity. This activity is again less demanding than index publication requirements. An analysis of these requirements is, however, beyond the scope of this paper.

PLANS FOR FUTURE WORK

The algorithm has been designed to handle general classes of structures with little consideration given to special classes. While this produces images suitable for most structures, images for certain classes of structures are clearly inadequate for much of the chemical community. Classes of structures being investigated as potential candidates for extensions to the algorithm include considerations for carbohydrates, steroids, polymers, incompletely defined structures, coordination compounds, and delocalized charges. In addition to algorithm extensions for handling special classes of structures, consideration is also being given to enhancing the esthetics of those structures currently handled.

ACKNOWLEDGMENT

CAS is pleased to acknowledge the funding of this work by the National Science Foundation (Contract C656).

REFERENCES AND NOTES

(1) D. P. Leiter, H. L. Morgan, and R. E. Stobaugh, "Installation and Operation of a Registry for Chemical Compounds", *J. Chem. Doc.*, **5**, 238 (1965).
(2) P. G. Dittmar, R. E. Stobaugh, and C. E. Watson, "The Chemical Abstracts Service Chemical Registry System. I. General Design", *J. Chem. Inf. Comput. Sci.*, **16**, 111 (1976).
(3) J. E. Blake, N. A. Farmer, and R. C. Haines, "An Interactive Computer Graphics System for Processing Chemical Structure Diagrams", presented to the Division of Computers in Chemistry at the 169th National Meeting of the American Chemical Society, Philadelphia, Pa., April 1975; *J. Chem. Inf. Comput. Sci.*, in press.
(4) B. L. Zimmerman, "Computer-Generated Chemical Structural Formulas with Standard Ring Orientations", Ph.D. Dissertation, University of Pennsylvania, 1971.
(5) N. A. Farmer and J. C. Schehr, "A Computer-Based System for Input, Storage, and Photocomposition of Graphical Data", *Proc. Assoc. Comput. Mach.*, **2**, 563–570 (1974).
(6) N. A. Farmer, "An Overlap Detection Algorithm", CAS working paper, Oct 1972.
(7) D. E. Knuth, "The Art of Computer Programming", Vol. 1, "Fundamental Algorithms", Addison-Wesley, Reading, Mass., 1975, p 316 ff.

LETTERS TO THE EDITOR

KEYWORDS VS. INDEX TERMS

Dear Sir:

Chemical Abstracts Service (CAS) welcomes user evaluations and letters regarding our publications and services. It is only from such user feedback that we can work effectively to improve our publications and services. We, therefore, were pleased when Ms. Barbara Charton wrote to us in December 1975 regarding her search in the *Chemical Abstracts* (CA) Keyword Indexes for applications of the Hammett equation and other linear free energy relationships. We replied to Ms. Charton's letter by suggesting a set of keywords that we felt would be useful for searching the Keyword Index where the index terminology is not controlled and another set of index terms for searching the CA six-month Volume Indexes where the index heading terminology is controlled.

In reading Ms. Charton's recent paper,[1] we noted that our reply was referenced. From the way Ms. Charton has described our reply, however, it is apparent that we must have somehow failed in our letter to distinguish clearly between the two types of CA subject indexes—the weekly issue Keyword Indexes and the six-month Volume Subject Indexes. For this, we apologize, and would like to offer some further clarification.

In her paper, Ms. Charton presents in part B of Table III a list titled "Suggested Keywords for the Cumulative Index". CAS uses the term "keyword" to describe only the content of the Keyword Index that appears in the weekly issues of CA. (This content also is included in certain CAS computer-readable files, e.g., *CA Condensates*.) The terms listed in part B of Table III are the keyword terms that we suggested to Ms. Charton as being useful in searching the weekly issue Keyword Index for references on the subjects of interest to her. There is no cumulative index of keywords.

The four phrases indentified as "index terms" in part A of Table III also are Keyword Index phrases. These are the phrases that appeared in the weekly CA issue Keyword Index for the review article on drug design that Ms. Charton identifies in her paper and for which she was unable to find a Keyword Index citation. It was this particular paper and its keyword indexing that prompted us to suggest to Ms. Charton that she add the terms "structure reactivity" and "structure relation" to her list of search terms along with "structure activity", which she had identified previously (Table I).

Ms. Charton asks, "why cannot the suggested keywords be used in the index which abstracts the paper?" We presume she means in the index of the weekly CA issue in which the abstract of the paper appears. We can only reply that the suggested keywords are those that we believe are most likely to appear in the CA issue Keyword Indexes for papers on the subjects of interest to Ms. Charton. However, the terminology