HOUDINI: A New Approach to Computer-Based Structure Generation

A. Korytko, K.-P. Schulz, M. S. Madison, and M. E. Munk*

Department of Chemistry and Biochemistry, Arizona State University, Tempe, Arizona 85287

Received March 27, 2003

A new method of structure generation called *convergent structure generation* has been developed to address limitations of earlier methods. The features of the program (HOUDINI) based on this method include the following: a single integrated representation of the collective substructural information; the use of *parallel atom groups* for efficient processing of families of alternative substructural inferences; and a *managed* structure generation procedure designed to build required structural features early in the process.

INTRODUCTION

Most comprehensive, computer-based structure elucidation systems have as their design objective the reduction of the spectral properties of an organic compound of unknown structure *directly* to a single molecular structure at best but to no more than a small number of plausible alternative structures at worst. In the latter case, a final assignment of the correct structure is often readily achieved since experienced chemists are proficient at distinguishing between a small number of alternative structures. Such computer programs can substantially augment the productivity of the chemist and offer a degree of assurance that no equally compatible structure has been overlooked.

Current comprehensive systems of structure elucidation incorporate some method of structure generation and, at the minimum, some capability in spectrum interpretation. Some systems also include procedures for spectrum prediction and comparison. The SESAMI system, for which this paper describes a significant enhancement, places strong emphasis on spectrum interpretation. This initial step in the process is designed to produce a set of substructural inferences sufficiently rich in information content to dramatically limit the number of plausible candidates (preferably to one) produced by the structure generator in a second step. If the information content of the input is not sufficiently rich, too large a number of compatible structures to be immediately useful to the chemist may be produced. In such a case, it is possible that the most probable structure or structures can be identified based on the goodness of the fit between predicted and observed spectra. This approach is often more effective in narrowing a relatively large set of structures. When the output of structure generation is a small set of candidates, the structures very likely differ little, and distinguishing between them using computer-based spectrum prediction can be a challenging task. The experience and intuition of an experienced chemist is usually required in making the final assignment.

BACKGROUND

Exhaustive Structure Generation. Most structure generators that have been described are *exhaustive* in their execution, i.e., they produce *all* plausible molecular structures compatible with the input. Although many structure generators have been described, their underlying procedures generally fall into one of two classes: *structure assembly* or *structure reduction*. The former forms the basis of most structure generators and was the earliest method to be studied. Structure assembly, as illustrated by programs such as ASSEMBLE,[1] CONGEN,[2] and MOLGEN,[3] is an *exhaustive*, recursive bond-making process that resembles a tree search. Input consists of three components: the *molecular formula* of the unknown; substructures required to be present (i.e., disjoint fragments with one or more atoms possessing open bonding sites) that serve as the *structural building units* of structure generation; and any *constraints* to be applied. The root node is the initial problem state (the initial "partial structure") and every other node except the lowest ("leaves" representing a final state, i.e., complete molecules) is an expanded partial structure. Each edge of the tree represents bond formation between two atom sites possessing residual valence. Many of the constraints serve as tests of the validity of each expanded partial structure. Failure to pass a test results in termination of that branch of the tree (thereby reducing the search space), followed by backtracking to a previous valid node. The application of some final tests to the leaves gives rise to an exhaustive set of constitutional isomers compatible with the input. ASSEMBLE recognizes resonance structures, symmetry, and redundant structures (most of which are eliminated prospectively). That program also provides the user with an extensive array of constraints (e.g., designating forbidden substructures, number and kinds of rings, and multiple bonds).[4] Internally, ASSEMBLE represents atoms in the molecular formula that are unaccounted for by the entered structural building units as single atom "substructures" with all bonding sites open.

Structure assembly programs have limitations. First, the substructures used as structural building units must have no atoms in common, i.e., they must not overlap one another. This nonoverlap requirement is a problem for the user because the usual spectral-based procedures for inferring the presence of substructures in an unknown generally reveal no information regarding overlap. Where *potential* overlap exists between two required substructures, one of the two can be entered instead as a *constraint* (the "required

* Corresponding author phone: (602)965-4430; fax: (602)965-2747.

10.1021/ci034057r CCC: $25.00
Published on Web 08/15/2003

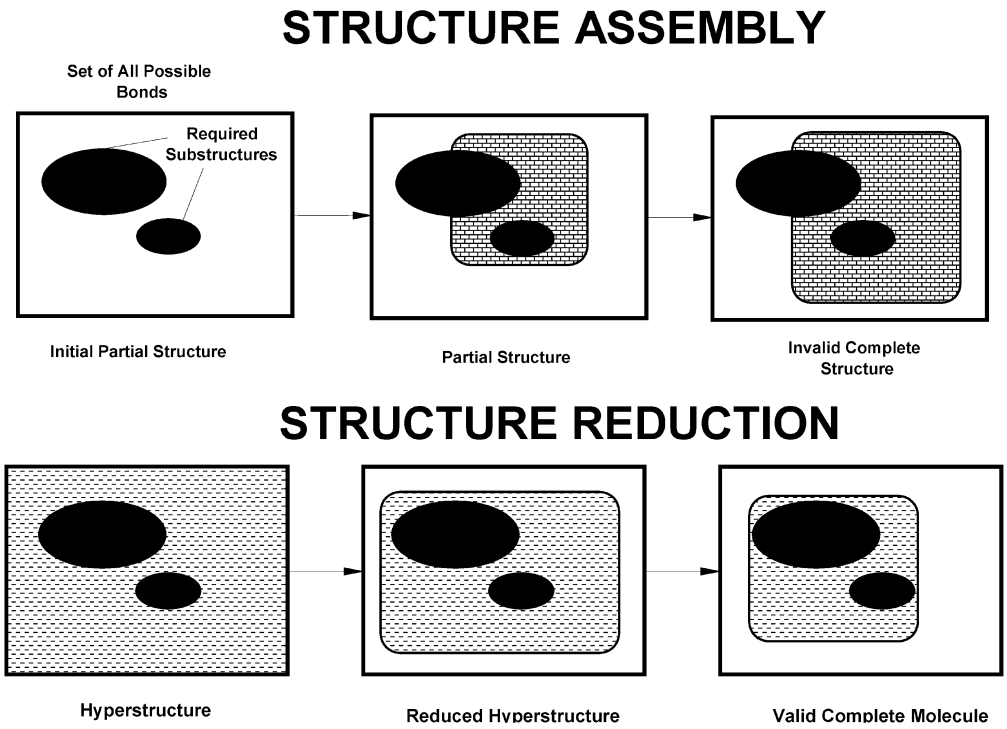

Figure 1. Venn diagrams illustrating structure generation: (a) structure assembly and (b) structure reduction.

substructure constraint"). Constraints are not directly involved in the bond-making process and are therefore not subject to the nonoverlap requirement. However, in structure assembly, required substructure information entered as constraints can only be used *retrospectively* to eliminate invalid molecular structures.

This inherent inefficiency of structure assembly can be readily understood using the Venn diagram shown in Figure 1a. The area within the boundary represents the universal set of bonds, i.e., all of the bonds that could be considered in constructing a molecular structure from its component atoms. Subsets of the universal set describe structural building units (nonoverlapping required substructures), required substructure constraints, partial structures formed during structure assembly, and complete molecular structures. Structure assembly begins with the initial partial structure, i.e., those subsets of bonds representing the structural building units (dark shaded areas). The initial partial structure (light shaded area) expands as each new bond is made. Each expanded partial structure is tested for consistency with the constraints. If a forbidden substructure is constructed, the pathway is terminated and backtracking occurs. If a required substructure constraint is fulfilled, no further testing of that constraint is needed. However, it is possible that a required substructure entered as a constraint (e.g., because nonoverlapping with other required substructures could not be established) may not be constructed until the very last bond completing a molecule is made, or it may not be constructed at all. Thus, the absence of that required substructure cannot be established with certainty until that very last bond completing a molecule has been made. For this reason, molecular structures that do not meet a required substructure constraint can only be excluded retrospectively, after the computation time to build them has been expended fruitlessly.

ASSEMBLE includes a constraint—the "neighboring atom tag"—to partially overcome the nonoverlap requirement.[4] "Tagging" an atom in a structural building unit with a neighboring atom identifies the presence of an immediately contiguous atom, but that atom within the tag is *not* considered to be part of the structural building unit for purposes of structure generation. Therefore, whether that atom does or does not overlap an atom in another substructure is of no concern. However, the neighboring atom tag is of limited utility since it can only specify one-atom overlaps. A more satisfactory solution is found in GENOA,[5] an extension of CONGEN. GENOA requires no distinction between nonoverlapping and potentially overlapping structural building units. Therefore, no required substructure has to be designated by means of a constraint.

In an initial step, GENOA uses an interactive, stepwise procedure called "constructive substructure search" to generate a set of partial structures ("cases") from the original set of substructures. A "generated" partial structure may be a single substructure or consist of two or more disjoint substructures (i.e., nonoverlapping substructures). Each such partial structure will be consistent with the collective information in the original set of substructures and all imposed constraints. In a final step, the remaining residual valence in each generated partial structure is saturated in all possible ways in a random structure assembly process to generate an exhaustive set of molecular structures. Although GENOA addresses the overlap problem in principle, in practice, active user intervention is required in order to prevent combinatorial explosion in the number of "cases" generated.

A second limitation of structure assembly is the inefficient processing of alternative substructure inferences. This is a serious problem because to an increasing extent in the solution of real-world structure problems, the spectral data can give rise to many families of alternative interpretations.

This is especially true with the widespread application of the ambiguous correlations resulting from some two-dimensional NMR experiments (e.g., long-range carbon−hydrogen correlations that do not distinguish between two and three (and at times four) intervening bonds).

The concept of structure reduction, developed to overcome the shortcomings of structure assembly, is the basis of the structure generator COCOA.[6] In contrast to structure assembly, the initial state of a structure reduction problem is one or more *hyperstructures*, each of which is expressed as a bond adjacency matrix (BAM). Each row/column of the symmetrical matrix represents one bonding site on a non-hydrogen atom of specified hybridization with its attached hydrogens (e.g., a $-CH_2-$ is represented by two bonding sites in the matrix). Therefore, each element in the matrix represents a possible bond between two bonding sites. In those cases where the experimental data do not permit the assignment of exact hybridization states to all non-hydrogen atoms, multiple initial hyperstructures result. Each initial BAM represents *all* bonds that are possible between the hybridization-specific, non-hydrogen atoms in the molecular formula and therefore an extremely large family of isomeric molecular structures, each of which is a subset of the bonds of the that BAM.

COCOA is an exhaustive, *recursive bond-removal procedure* which systematically examines the search space for all valid subsets of bonds in each initial BAM that correspond to molecular structures compatible with the input. Conceptually, structure reduction offers a number of advantages over structure assembly. First, it has no requirement for nonoverlapping required substructures. Second, in contrast to structure assembly, all required substructures are used prospectively. This is explicable in terms of the Venn diagram (Figure 1b). In structure reduction, the initial state already contains all of the required substructures. Bonds are now removed from the initial BAM in a systematic, stepwise fashion, generally more than one at a time. At each step of the bond removal process, the resulting *reduced* hyperstructure is tested for consistency with the all constraints, including the presence of required substructures. Here, the absence of a required substructure can be detected as soon as the very first bond rupturing the required substructure is deleted. Structure generation along that pathway is terminated, leading to the elimination of an entire family of invalid molecular structures before they are generated. Since any number of required substructures can be tested for at each step, it does not matter whether they overlap or not. Forbidden substructures are likewise detected prospectively. Structure reduction also allows for the prospective utilization of families of alternative interpretations of data and symmetry information, the latter being derived in part from the required 1D ^{13}C NMR.

Most but not all workers developing comprehensive computer-based structure elucidation systems have required structure generators that are *exhaustive* in execution. Although the early structure assembly based programs, ASSEMBLE, CONGEN, MOLGEN, and GENOA, were developed as stand-alone structure generators with the substructural inference input to be provided by the user (A later version of ASSEMBLE also directly processes limited 2D NMR data.[7]), a number of comprehensive structure elucidation programs also incorporate such structure assembly based structure generators. These include CHEMICS,[8] StrucEluc,[9] COCON,[10] EPIOS,[11] and SpecSolv.[12] The first three of these programs are exhaustive in the sense that their output excludes no plausible structure. In contrast, the latter two systems are database dependent. Although their structure generators are exhaustive, they can only construct compounds from structural units present in their databases of assigned ^{13}C NMR spectra.

CHEMICS, one of the earliest of the comprehensive systems, has been the subject of an ongoing process of improvement. The program uses a library of predefined, nonoverlapping, small structural fragments, each of which has been assigned spectral properties. In an initial step, fragments whose assigned properties are not compatible with the observed spectral data are deleted. Surviving fragments, some of which are invalid, are arranged into sets of fragments each of which is consistent with the molecular formula of the unknown and any user-entered constraints. Each fragment set is treated as a separate structure assembly process. Structure generation can be constrained by user-entered substructures, inferences derived from 2D NMR data,[13] and carbon signal number considerations.[14]

StrucEluc[9] is an enhancement of an earlier expert system that makes extensive use of 2D NMR data. In an initial step, the program produces a set of homonuclear and heteronuclear atom−atom correlations which serves, along with other substructural inferences derived from the collective spectral data, as input to the structure generator. The structure generator is described by the authors as "classic" but enhanced to accommodate the 2D NMR-derived correlations. As in earlier versions of the program, a limited number of constraints, e.g., ring size, bond multiplicities, act to limit the number of plausible structures generated. Spectrum prediction software is used to select the most probable of the generated structures. The program has been applied to the structure elucidation of naturally occurring compounds.[15]

COCON is a recently developed program that relies heavily on atom−atom correlations derived from 2D NMR experiments, many of which are ambiguous. In a structure assembly based structure generation procedure, the exact hybridization and the number of attached hydrogens are required for each non-hydrogen atom. However, since the experimental data do not always permit the assignment of an exact hybridization state to each atom, COCON first generates all possible hybridization state combinations of the non-hydrogen atoms, each of which is then used to assemble final structures. Some very general ^{13}C NMR chemical shift information (e.g., carbon atoms connected to two oxygen atoms must have a chemical shift greater than 90 ppm) is also used to limit the number of candidates produced. COCON can produce a large number of candidates in the case of real-world structure elucidation problems.[16,17] Therefore, the ^{13}C NMR chemical shift prediction program SpecEdit[18] is used to retrospectively rank the generated structures in order of decreasing probability of being correct.

In the database-dependent systems EPIOS and SpecSolv, a large database of assigned ^{13}C NMR spectra is the source of a library of concentrically layered, carbon-centered substructures each carbon atom of which is assigned a chemical shift range and signal multiplicity. In an initial step, each of these systems identifies those substructures in the library which, within a specified tolerance, are consistent

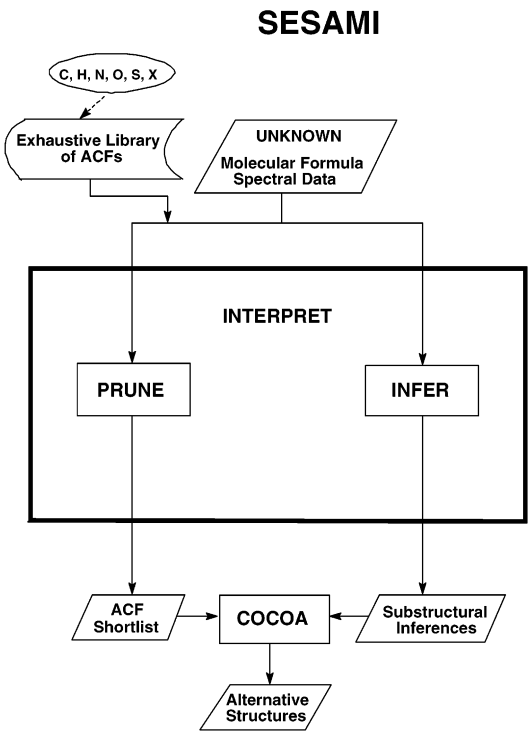

Figure 2. Information flow in SESAMI.

with the unknown's observed spectral data. The retrieved substructures—some of which are invalid, as expected—serve as the structural building units in a stepwise process in which fragment overlap forms the basis for structure generation. The process is constrained by predicting chemical shifts for each carbon atom in a partially assembled structure and comparing them with observed spectral data. Partially assembled structures with chemical shift deviations that exceed an established value are discarded or weighted less. Spectrum prediction is used again to retrospectively rank the final set of generated molecular structures. A notable advantage of SpecSolv is that it does not require a molecular formula.

The first structure reduction-based structure generator, COCOA, was designed to seamlessly link to a spectrum interpreter (INTERPRET). Together they form the basis of the current version of the SESAMI system.[19] The information flow in SESAMI[20] is shown in Figure 2. INTERPRET is a two-track spectrum interpretation program that currently uses 1D and 2D ^{13}C and ^{1}H NMR spectral data as its main source of information. On one track (Program PRUNE), the *ACF Shortlist* is generated which, based on the collective spectral data, consists of a family of likely alternative atom-centered fragments (ACFs) for each non-hydrogen atom in the unknown. An ACF describes the immediate chemical environment about a non-hydrogen atom. It is an explicitly defined structural unit consisting of a valence-satisfied central atom, with its attached hydrogens, if any, and one concentric layer of nearest neighboring non-hydrogen atoms—also with its attached hydrogens, if any—at least one of which possesses residual valence. The bonds by which neighboring atoms join to the central atom and the bonds by which the first layer atoms can join to other atoms—"half-bonds"—are specified ($-CH_2-CH_2-O-$ is an example of a methylene-centered ACF).

PRUNE acts on a computer-generated, exhaustive library of ACFs each of which has been assigned a set of properties. In particular, the central carbon atom of each carbon-centered ACF is assigned a chemical shift range (derived largely from a library of assigned ^{13}C NMR spectra) and a signal multiplicity based on the number of attached hydrogen atoms. Hydrogens attached to the central carbon atom are also assigned a chemical shift range. In generating the ACF Shortlist, PRUNE deletes those ACFs from the exhaustive list that are incompatible with the collective spectral properties of the unknown. Thus, each carbon-centered ACF family in the ACF Shortlist contains only those carbon-centered ACFs whose assigned central carbon chemical shift range and signal multiplicity (and assigned hydrogen chemical shift range if there are attached hydrogens) are consistent with a given observed carbon (and hydrogen) signal. Heteroatom-centered ACF families are produced in a similar fashion. The number of alternative ACFs in an ACF family can be quite large—in some cases exceeding 100—since the ACF is too small a fragment to permit a distinction between each ACF on the basis of spectral properties.

On the second track of INTERPRET (Program INFER) the program uses data from 2D NMR experiments (HMQC, COSY, HMBC, 2D INADEQUATE) to produce a set of correlations, each of which represents a pair of carbon atoms (labeled with their respective chemical shifts) separated either by an exact or variable number of intervening bonds (usually between one and four bonds). The number of such correlations produced from the experimental 2D NMR data from a complex compound can be quite large (often exceeding 100). SESAMI can also utilize user-defined or other computer-generated substructures. The combined output of the two tracks of INTERPRET, and any other entered information, is handed to the structure generator COCOA with or without prior editing by the user.

Recall that the initial problem state in COCOA is one or more initial hyperstructures (the initial BAMs). COCOA *exhaustively* processes each of these in a two-stage process. The first, *ACF Selection*, utilizes the information in the ACF Shortlist to guide bond removal in the BAM in a process that is further constrained by the 2D NMR-derived atom−atom correlations. Although bond reduction is extensive in this first stage, a complete one-to-one mapping of bonds (a molecular structure) sometimes requires a second stage, *Bond Selection*, a depth-first search in which the BAM is tested at each bond making step for consistency with all available information. This recursive two-stage procedure, which combines structure reduction and structure assembly, continues until all compatible, one-to-one mappings between bonds in each of the hyperstructures have been identified.

Other structure generators utilize variations of the concept of structure reduction first applied in COCOA. GEN is a standalone structure generator in which the initial problem state is also a hyperstructure.[21] In the first step of the process, information contained in user-entered substructural inferences—substructures that must be present and those that must be absent—is used to remove bonds. Although there are no substructure size restrictions, substructures that must be present in the unknown must not overlap one another (a disadvantage shared with ASSEMBLE). In a second step, a "controlled" bond selection procedure leads to one-to-one mapping of bonds.

CISOC-SES,[22] like SESAMI, processes 1D and 2D NMR data. During structure generation, the program initially creates hyperstructures similar to those of COCOA but with atom hybridization unspecified. Next, bonds in the initial hyperstructure are removed using carbon−carbon connectivities inferred from the 2D NMR to force connections between atoms. The resulting reduced hyperstructure is then processed by a procedure similar to the Bond Selection step of COCOA (i.e., "fixing" bonds in a depth-first manner) to produce a subset of bonds representing a molecular structure. However, in contrast to COCOA's Bond Selection, the most *probable* bonds are selected first based on long-range carbon−hydrogen correlations and limited use of ^{13}C NMR chemical shift/substructure (one-sphere, carbon-centered fragments) correlations. The recursive procedure produces a ranked list of candidates. As in COCOA, required substructure information is utilized prospectively.

Stochastic Structure Generation. Current, computer-based structure elucidation programs employing exhaustive structure generation are not routinely successful in solving the structures of higher molecular weight compounds, in particular, complex naturally occurring compounds. The problem relates to the rapidly increasing size of the problem search space with increasing numbers of atoms. One consequence of this is that program execution may not be completed on a practical time scale. To obtain information useful to the chemist in such cases, several investigators have turned to stochastic methods. Although such methods lend themselves to efficient execution, they only provide optimum solutions; possibly, but not necessarily including the correct solution. The problem search space can be thought of having several "minima", only one of which corresponds to the correct structure or a small set of structures including the correct one. A stochastic method will find a minimum but not necessarily the correct one. However, at the very least, it should provide some insight into the nature of the structure.

Faulon[23] was among the first to develop a stochastic structure generator based on simulated annealing. SENECA,[24] a recent and more comprehensive program, also uses simulated annealing. The program begins by generating a random structure based on the molecular formula and a set of constraints derived from spectral data, NMR in particular. An "energy" is calculated which is an evaluation of the fit of structure to its spectral properties. That structure is then randomly transformed to a new structure. If its energy has decreased, the structure is "accepted", and the process continues until an optimum energy is achieved. The method requires very little information prior to commencing structure generation, and the range of structures produced is not dependent on structures present in a database.

A similar stochastic approach is based on a genetic algorithm.[25] Starting with a set of randomly generated structures based only on the molecular formula of the unknown, succeeding generations of structures—"offspring"—are produced by combinations of the "fittest" members of the set—"parents"—as judged by a fitness function. In this case, the fitness function compares the predicted 1D ^{13}C NMR spectrum for each structure with the observed spectrum. The process continues until convergence is attained. Additional information in the form of present or absent substructures is not a requirement. However, required substructures can be used in the generation of the initial random set and retained during further processing. Forbidden substructures can be used to exclude offspring from consideration.

Exhaustive Versus Stochastic Methods. The needs of chemists engaged in structure elucidation can best be served by the availability of an armamentarium of diverse computer-based tools to address the broad range of structure problems encountered in the laboratory. The SESAMI system, which is based on exhaustive structure generation, was designed for those occasions when the chemist needs the maximum assurance that no plausible alternative has been overlooked in assigning the structure of the unknown. Stochastic systems can provide useful information about the kinds of structures compatible with the spectral input in cases where systems based on exhaustive structure generation fail to produce any useful information.

Structure Generation: A Constraint Satisfaction Problem. Structure generation can be viewed as a constraint satisfaction problem since its function is to constrain the set of structural isomers corresponding to a given molecular formula to the subset compatible with a set of substructural inferences. The general class of constraint satisfaction problems has been studied by computational mathematicians who developed a constraint satisfaction problem (CSP) paradigm, a model for resolving conflicts by removing or reconciling inconsistent values in a constraint network that represents the structure of the problem.[26] An examination of GENOA's "constructive substructure search" (see above) suggests that this procedure, although described prior to the formulation of the CSP paradigm, resembles that approach. GENOA tries to match all of the atoms contained in the substructural inferences (by "removing or reconciling inconsistent values") with the atoms of the molecular formula. However, that process is slowed by the need to consider the substructural atoms of one substructure at a time.

A recently described structure generator, Program LSD,[27] which is based on the CSP paradigm, resolves that shortcoming by mapping all of the substructural atoms in an arbitrary order. The process is similar to but more efficient than that of GENOA, because LSD uses required substructures in a more concerted way. LSD's input is largely limited to constraints derived from several 2D NMR experiments, the ambiguous ones of which are expressed as sets of alternative substructures. However, the alternative substructures within a set are tested one at a time, with the order in which they are selected being completely arbitrary. The output of LSD is similar to GENOA's final "cases" (i.e., partial structures), each of which is then expanded to a set of complete molecules. In the structure problems studied,[27] a large number of complete molecular structures, including some highly strained, were obtained.

The Shortcomings of Current Exhaustive Structure Generation Methods. Although the COCOA-based program SESAMI and some other comprehensive structure elucidation systems have demonstrated considerable problem-solving capabilities,[28,29] experience with higher molecular weight compounds of complex structure revealed the need for enhancements, in particular, in the efficiency of structure generation. A useful first step in the design of a new structure generator is the identification of the sources of inefficiencies of current approaches.

Formulation of the structure generation problem in terms of the formal CSP paradigm provides a useful means to classify and compare the methods used in the various structure generators. In structure assembly, new bonds are added to a partial molecular structure until either a forbidden substructure or a molecular structure is constructed. Using a backtracking procedure, all possible solutions are found. In terms of the CSP paradigm, this method resembles *chronological* backtracking, a searching algorithm. Structure reduction, on the other hand, removes those bonds from the hyperstructure which do not satisfy at least one of the substructural inferences. This method of checking consistency is analogous to the CSP's *forward checking* consistency algorithm.

Although chronological backtracking should always give an exhaustive set of plausible molecular structures, it is one of the most inefficient search methods for this purpose because many of the dead-end branches of the search tree which do not lead to a valid solution are being fully visited. The number of such dead-end branches in a typical real-world structure generation problem can be enormous. The forward checking in structure reduction, on the other hand, enhances the "vision" of the program allowing it to "see" many of those dead-end branches and to reject them prospectively. However, in the case of large, complex structure problems, structure reduction run times may still be excessive for a number of reasons. First, even with prospective utilization of both unambiguous and ambiguous inferences, the applied consistency checks based on this information are made sequentially in an order of limited flexibility. Thus, the "prospectiveness" of the process may not be optimized, and, for a given problem, the order imposed may result in too much time being spent "reducing" the least promising parts of the universal set of bonds. Second, with sequential consistency tests, inferences are not being used in concert, and, as a result, useful information may go unused. For example, the information in two or more ambiguous, long-range carbon−hydrogen correlations derived from the 2D NMR HMBC experiment may unambiguously define a specific carbon−carbon bond. Third, the individual inferences within a given family of alternative inferences (e.g., a large family of alternative ACFs assigned to a particular non-hydrogen atom) are tested sequentially, with little control on the order, and therefore inefficiently. Finally, as noted earlier, if the assignment of a specific hybridization to all atoms is not possible, more that one initial hyperstructure may be needed to represent the necessary search space, each of which must be treated as a separate structure generation problem.

CONVERGENT STRUCTURE GENERATION

General Principles. To address the shortcomings of earlier methods, a new approach to structure generation, *convergent structure generation*, has been devised which integrates elements from both structure assembly and structure reduction. Development of the method was facilitated by a concept called *constraint propagation* which has proved useful in solving a variety of constraint satisfaction problems.[26] It involves making certain assumptions about how the solution to a problem should look.

For example, by connecting some pairs of atoms, i.e., by assembling some substructures, generation can be directed to the more promising branches of the search tree. Program HOUDINI, an initial implementation of the new method, incorporates a number of important new features that make implementation of this concept practical. First, although the input to the new program remains the same as that to COCOA—the families of alternative ACFs for each non-hydrogen atom, the unambiguous and ambiguous atom−atom correlations derived from 2D NMR experiments and any other computer-generated or user-defined substructures—HOUDINI does not use the inferences separately as does COCOA. In an initial step, a *single, integrated substructural representation* is created which incorporates *all* of the information: explicitly defined substructures, ambiguously defined substructures, and the families of alternative substructures. The result is that the whole is greater than the sum of the parts; the integrated representation is substantially richer in information content since it reveals structural relationships between the individual substructures.

Second, HOUDINI uses the integrated structural representation to construct valid structural features and, at the same time, to reduce the search space of the problem. The two processes converge in the generation of a valid molecular structure. Operationally, the order of information utilization is *managed* so as to maximize the efficiency of convergence. Third, HOUDINI incorporates a newly developed procedure—*parallel alternative substructure processing*—which allows alternative inferences to be processed simultaneously and at the same time as other inferences.

Computer Representation. There are two major components in the computer representation of a structure generation problem in HOUDINI, the *hyperstructure* and the *substructure representation*. Each has its own data structure, the contents of which at any stage of structure generation reveal the status of the process. The data structure of the hyperstructure is a two-dimensional, symmetric, square matrix in which the labels of the rows and columns are the individual *actual atoms* of the molecular formula. The matrix describes all possible bonding relationships. Hybridization of the atoms can but need not be specified. Hydrogen atoms can be included in the matrix as separate actual atoms or attached initially to non-hydrogen atoms to the extent that information is known. In the latter case, hydrogen atoms not initially accounted for will be connected to residual valence sites after the completion of molecular skeleton. It should be noted that in contrast to COCOA, only a *single* matrix is needed to represent a structure generation problem in HOUDINI.

In the hyperstructure, every bonding relationship in the matrix is either a *fixed bond* or a *possible bond*. A fixed bond is one that has been assigned as a single, double, triple, or "no" (i.e., forbidden) bond between two atoms. Except for backtracking, the multiplicity of a fixed bond does not change during structure generation. A possible bond is one that has not been assigned to one of the four above classes. Any possible bond may be transformed into a fixed bond as structure generation progresses toward a solution. This transformation is reversed during backtracking.

The initial state (*initial* hyperstructure) for a real-world problem usually consists of possible bonds with a small subset of fixed bonds representing known required and forbidden substructures (computer and/or user-derived). A final state of the problem, representing *one* plausible solution,

a. **Initial State**

	1. CH_3	2. CH_2	3. CH_2	4. OH
1. CH_3	Fixed (0)	Possible	Possible	Possible
2. CH_2	Possible	Fixed (0)	Possible	Possible
3. CH_2	Possible	Possible	Fixed (0)	Possible
4. OH	Possible	Possible	Possible	Fixed (0)

b. **Final State**

	1. CH_3	2. CH_2	3. CH_2	4. OH
1. CH_3	Fixed (0)	Fixed (1)	Fixed (0)	Fixed (0)
2. CH_2	Fixed (1)	Fixed (0)	Fixed (1)	Fixed (0)
3. CH_2	Fixed (0)	Fixed (1)	Fixed (0)	Fixed (1)
4. OH	Fixed (0)	Fixed (0)	Fixed (1)	Fixed (0)

Figure 3. States of the propanol problem: (a) initial state (hyperstructure) and (b) "final" state (a one-to-one mapping of bonding sites). A zero designates "no bond."

contains no possible bonds. At any intermediate state, a *reduced* hyperstructure describes the status of the problem. Using an example for purposes of illustration, the initial and final states of a simple problem (C_3H_8O) are shown in Figure 3. Note that in this example, hydrogens are not treated as separate atoms; non-hydrogen atoms with their attached hydrogens are used. In the initial state, only the diagonal bonds are fixed and set to zero (i.e., no bond) since atoms do not bond to themselves. The final state—all fixed bonds—is a one-to-one mapping of bonds. (There is only one valid final state with the distribution of hydrogens shown: CH_3-CH_2CH_2OH.)

The second component of the computer representation of a structure generation problem, the substructure representation (SR), also has multiple states. In the *initial* state, the data structure describes the *collective information* in all of the required substructural inferences (forbidden substructures are treated separately). The SR takes the form of an *integrated network* of substructure atoms each of which is *generic* (A1, A2,...An) and has a *domain*, defined as that subset of the *actual atoms* to which it can be mapped. As structure generation proceeds, the size of a domain can change. In a final state, each generic atom is mapped to a single actual atom.

Bonds between substructure atoms in the network can be specified as single, double, triple, or undefined. In the initial state, bonds types will be undefined if the hybridization of some (many) substructure atoms is not specified. In contrast to a set of individual substructure inferences, the SR network can reveal the relationship of one substructure atom to another substructure atom, e.g., whether the two different substructure atoms can be mapped to the same actual atom, i.e., superimposed on one another.

The properties of an actual atom in a substructure atom domain—in particular, hybridization and the number of attached hydrogens—may but need not be explicitly defined. During structure generation, the properties of the actual atoms become more explicitly defined, thereby limiting the choice. Mapping a substructure atom with explicitly defined properties to an actual atom precludes any change in those properties as structure generation proceeds in the forward direction.

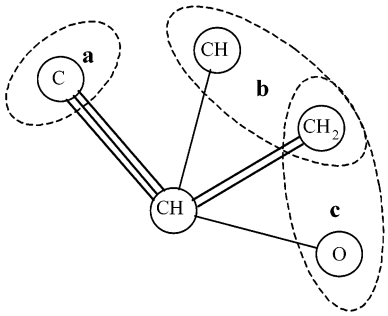

Figure 4. A single "network" representing a family of three alternative substructures (ACFs) with a common central CH group. Atom groups a, b, and c are the nearest neighbors of the central CH group in the three ACFs.

The SR is designed to permit a description of a family of alternative substructures (e.g., a family of alternative ACFs), only one of which is required to be present in a final molecular structure. Members of the family may have some atoms in common which can significantly reduce the number of substructure atoms needed to represent the family. The concept is illustrated in Figure 4. For simplicity, specific rather than generic atoms are used. The single "network" shown represents a family of three ACFs (a, b, and c), each of which has the same central atom: CH. In ACF a, the central CH is joined to a C by a triple bond; in ACF b, it is joined to another CH by a single bond and to a CH_2 by a double bond; and in ACF c, it is joined to an O by a single bond and to a CH_2 by a double bond. Note that ACFs b and c have a CH_2 in common. That CH_2 will be mapped to an actual atom if either ACF b or c is to be satisfied. Each of the three groups of atoms in a broken-line ellipse is an *alternative neighborhood* (AN). The set of alternative neighborhoods associated with one substructure atom establishes a *parallel atom group* (PAG) that forms the basis of *parallel alternative substructure processing* (see **Parallel Alternative Substructure Processing** below).

An integrated network of connected ACF families can be created from the atom−atom correlations derived from 2D NMR experiments. For example, the central atoms of two carbon-centered ACF families (e.g., atoms A1 and A2) can be bonded directly together based on a particular three-bond hydrogen−hydrogen COSY correlation (in conjunction with one-bond carbon−hydrogen HMQC correlations), or they can be ambiguously joined in the case of a particular long-range carbon−hydrogen HMBC correlation (again in conjunction with HMQC). As an example of the latter, consider an unknown containing only carbon and hydrogen which gives rise to an observed HMBC correlation that relates central carbon atoms A1 (with domain {C1}) and A2 (with domain {C2}). Although the HMBC correlation may not distinguish between one, two, and three intervening bonds between those carbon atoms, the ambiguous "connection" between atoms A1 and A2 can be represented as a *single*, bonded array of atoms [A1-A3-A4-A2], where the domains of A3 and A4 are {C1, C3, C4...Cn} and {C2, C3, C4...Cn}, respectively. If A3 is mapped to C1 and A4 is mapped to C2, the representation collapses to C1−C2, i.e., one bond between A1 and A2. If A3 is mapped to C1, but A4 is not mapped to C2 *or* if A4 is mapped to C2, but A3 is not mapped to C1, two bonds separate C1 and C2, i.e., there is one intervening carbon atom between them. If A3 is not mapped to C1 and

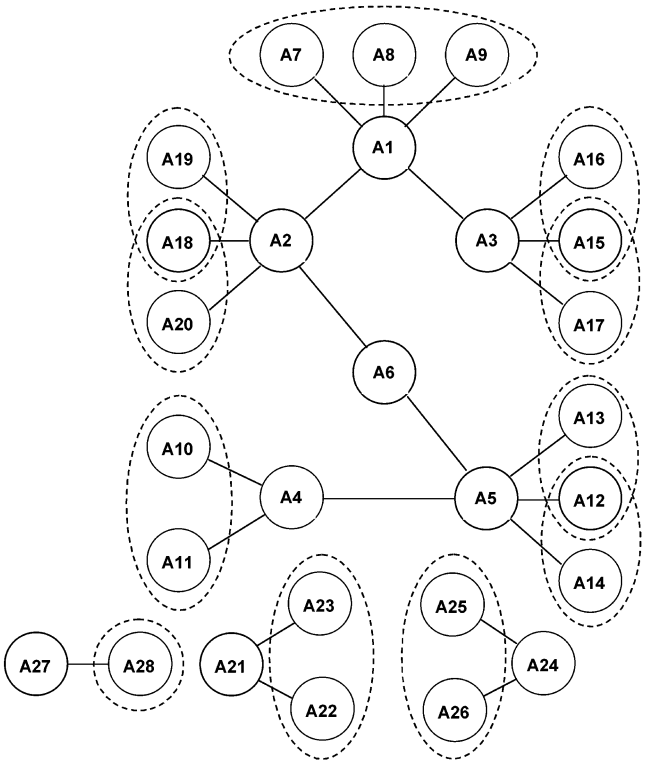

Figure 5. A substructural representation (SR). Broken-line ellipses designate alternative neighborhoods.

A4 is not mapped to C2, and if A3 and A4 are not mapped to the same actual atom, there are three bonds between C1 and C2.

Figure 5 describes a simple, but complete, *initial* SR created by HOUDINI using the dual output of INTERPRET. The molecular formula of the unknown, C_5H_9BrOS, and one- and two-dimensional NMR spectral data served as the input to INTERPRET. The initial SR of the unknown consists of a network of 28 generic atoms (A1-A28) each of whose domain is listed in Table 1. In this example, the actual atoms in the domain of each generic atom in the SR are expressed as *element groups* (ELGs). An ELG is a non-hydrogen atom with its attached hydrogens (if any) and each of its remaining residual bonding sites, differentiated by bond order (e. g., $-CH_2-$; $-CH=$). Although HOUDINI does not require attached hydrogens—they can be treated as any other atom—element groups are often used in solving structure problems since such information is readily available for carbon from ^{13}C NMR data.

The broken-line ellipses in Figure 5 describe alternative neighborhoods, a means of representing families of ACFs for efficient processing of alternative substructural inferences (see **Parallel Alternative Substructure Processing** below). As revealed in Table 1, the domains of generic atoms A1, A2, A3, A4, and A5 each map to a *single actual atom*; the carbon atom with the specified chemical shift label, e.g., the domain of A1 is the CH (sp^3) group whose carbon chemical shift is 83.1 ppm. Each of the *unlabeled* methylene groups—A7, A8, A10, A12, A22, A25, A26, and A28—can be mapped to any one of the four actual methylene groups (38.8, 36.9, 70.9, and 36.0 ppm) in the molecule. Since there is only one of each of the heteroatoms in the molecule (O, S, Br), the domains of the corresponding substructure atoms can contain only one actual atom.

Table 1. Domains of the Substructure Atoms of Figure 5

atom	ELG	shift
A1	CH (sp^3)	83.1 ppm
A2	$-CH_2-$	38.8 ppm
A3	$-CH_2-$	36.9 ppm
A4	$-CH_2-$	70.9 ppm
A5	$-CH_2-$	36.0 ppm
A6	any	
A7	$-CH_2-$	
A8	$-CH_2-$	
A9	$-O-$	
A10	$-CH_2-$	
A11	$-O-$	
A12	$-CH_2-$	
A13	$-Br$	
A14	$-S-$	
A15	CH (sp^3)	
A16	$-Br$	
A17	$-S-$	
A18	CH (sp^3)	
A19	$-Br$	
A20	$-S-$	
A21	$-O-$	
A22	$-CH_2-$	
A23	CH (sp^3)	
A24	$-S-$	
A25	$-CH_2-$	
A26	$-CH_2-$	
A27	$-Br$	
A28	$-CH_2-$	

The chemical environment of generic atoms A1, A2, A3, A4, and A5 derives from two sources, the ACF Shortlist and the 2D NMR data. A single ACF survives in carbon chemical shift family C83.1 and that ACF describes the chemical environment of atom A1. Two different ACFs represent atoms A2, A3, and A5. (In most complex structure problems, the number of alternative neighborhoods is usually much greater than two.) The connections between A1 and A2, A1 and A3, and A4 and A5 were revealed by the COSY experiment. Based on an HMBC experiment, atoms A2 and A5 are joined through one *or* two intervening bonds, i.e., initially the domain of A6 contains *every* atom including those of the domains of both A2 and A5. Atoms A21, A24, and A28 are heteroatoms for which no connectivity information is available. In this example, each of these heteroatoms is represented by a single ACF. The changing nature of the SR during structure generation is discussed below (see **Parallel Alternative Substructure Processing** below).

The SR network is advantageous for two reasons: it is no longer necessary to break structure generation into several steps as in COCOA,[6] and, relative to the listing of separate substructural inferences used by COCOA, HOUDINI's substructure representation is substantially richer in information content since it reveals *structural interrelations* between the substructures.

Main Algorithm. The function of HOUDINI is to determine all possible ways in which the atoms of the substructural inferences can be mapped to the actual atoms of the molecular formula such that the integrity of all of the information content is preserved. The program initially constructs the hyperstructure and the substructure representation from the entered data. The information in the latter is then used to *fix* possible bonds in the hyperstructure—comparable to a structure assembly procedure—and, at the same time, to *reduce* the set of remaining possible bonds—comparable to a structure reduction procedure.

The overall structure generation process is controlled by *constraint agents* of which there are three types: *managing*, *checking*, and *mixed*. These constraint agents are the key to efficient information processing. They are responsible for building the required substructural features in the hyperstructure as early as possible in the structure generation process. Managing constraint agents fix bonds between actual atoms of the hyperstructure. Every managing constraint agent keeps track of the possible bonds it fixes in order to ensure the generation of all possible solutions. Checking constraint agents function as consistency checking routines which are applied after every new bond is fixed. Depending on the *consistency status* returned by such routines, another bond will be fixed or backtracking to the previous level of the "tree search" will occur. Another important function of checking constraint agents is structure reduction, i.e., reducing the search space comprised of remaining possible bonds, which results in simplification of both the hyperstructure and substructure representation. Constraint agents that act as both managing and checking constraint agents are referred to as "mixed."

Only one managing constraint agent can be active at a time. A priority queue based on preassigned *managing priority values* controls the order in which they are called. Initially, that managing constraint agent with the highest priority value selects possible bonds to fix. Each managing constraint agent has a specific goal while it is active; as soon as it is fulfilled, control is transferred to the managing constraint agent with the next highest priority value. In the event of backtracking to a state where transfer of control occurred, the former managing constraint once again assumes control.

Checking constraint agents are organized in a stack which HOUDINI uses in its entirety each time a bond is fixed. However, checking constraints agents that are known in advance to be satisfied when starting from a particular search tree level are excluded from the stack until the program backtracks to a preceding search tree level.

The main constraint agent, the *substructure representation (SR) agent*, is mixed since it has properties of both a managing constraint agent and a checking constraint agent. The goal of the SR agent is satisfaction of all of the information embedded in the substructure representation. This agent manages the process of fixing bonds by *labeling*, i.e., selectively choosing single actual atoms from the domains of the substructure atoms. An actual atom chosen as a *label* of some substructure atom must be connected to all already labeled neighbors of the substructure atom.

The order in which substructure atoms are labeled is controlled by heuristics based on various factors, e.g., the size of the substructure atom domain, the number of alternative substructures containing the substructure atom, and the number of free valences of the actual atoms in the substructure atom's domain. The ordering corresponds to the CSP's *variable ordering*.[26] It is known that computation times can vary significantly with the variable ordering heuristics used. After choosing the substructure atom to be labeled, the SR agent then chooses its label from among the actual atoms of the domain. This decision is also controlled by heuristics, e.g., the probable distance of the chosen atom to other actual atoms, information which is available from the 2D NMR data. The order of choosing the actual atoms within a domain corresponds to CSP's *value ordering*.[26] Value ordering influences the order in which solutions are produced but generally not computation time.

The procedure of labeling substructure atoms by the SR agent resembles a typical substructure search algorithm. That algorithm tries to match substructure atoms with the atoms of the complete structure in terms of atom attributes, e.g., element and specific connectivities. However, during structure generation in HOUDINI, the actual atoms may have some but not all of their connectivities specified. Thus, HOUDINI's substructure atoms are labeled with actual atoms based on the *possible future* match with the substructure atoms' attributes. Choosing an actual atom as a label of a certain substructure atom will, in effect, force further definition of the actual atom's attributes, including its connectivity, i.e., the actual atom will have to be connected (unless it already is) with all other actual atoms that serve as labels to the substructure atom's neighbors.

Labeling one substructure atom can invalidate some possible labels (domain members) of other substructure atoms. In its role as a checking constraint agent, the SR agent conducts consistency checks after each new bond is fixed, which in turn can lead to a reduction in the domain sizes of other substructure atoms. These reductions are based on several criteria. First, every actual atom in the domain of a substructure atom should be assignable as a label without violating its valence, connectivity or other current attributes. Second, this same rule is applicable to the *group* of substructure atoms forming the outer layer of an ACF. All actual atom labels which fail these tests are excluded from their respective substructure atoms' domains. Except for atoms of parallel atom groups (which express alternative substructures and are treated differently; see **Parallel Alternative Substructure Processing** below), as soon as the domain of one substructure atom is emptied, the SR agent signals HOUDINI to backtrack.

In addition to satisfying the collective substructure constraints contained within the substructure representation, HOUDINI includes checking constraint agents that search the hyperstructure for forbidden substructures, e.g., a library of chemically strained structural features. If such a feature is detected, backtracking results along the current tree branch to the nearest previous valid state. Checking constraint agents that maintain a record of assembled cycles facilitates the process.

Although the utilization of forbidden substructures is important, it should be noted that implementation is less efficient than the building of required substructures in the hyperstructure. The latter process has a higher degree of prospectiveness since it reduces the total problem search space in a forward-looking fashion; i.e., as bonds are fixed, the possible bond space and the domains of substructure atoms are concurrently reduced. In contrast, the most efficient use of a forbidden substructure requires less prospectiveness; eliminating potential bonds which would complete a forbidden substructure. Moreover, every forbidden substructure is checked individually, while the required substructure information, integrated in a single representation, is used concertly. Therefore, in simple cases, it could be beneficial to express a forbidden substructure as a set of alternative required substructures that excludes construction of the forbidden substructure, i.e., taking the "inversion" of the

forbidden substructure. Forbidden substructures with only a few atoms can have an inverted set with a small number of alternatives. For example, consider forbidden substructure $A_1{=}A_2$, where A_1 is mapped to a single actual atom, CH, and A_2 is any other actual atom capable of double bonding with A_1. This forbidden substructure can be replaced with a set of required substructures in which A_1 is either alkanyl or alkynyl but not alkenyl as in the forbidden substructure. In this case, that set includes two alternative required substructures: $A_1(\text{-}A_3)(\text{-}A_4)\text{-}A_5$ and $A_1{\equiv}A_6$ where A_3, A_4, A_5, and A_6 are any other actual atoms with appropriate bonding sites. Since structure generation is guided by the required substructures, substructure $A_1{=}A_2$ will never be constructed.

COCOA requires an ACF Shortlist, i.e., an ACF family (containing at least one ACF) for each non-hydrogen atom in the molecular formula. This requirement limits its general applicability as a structure generator. HOUDINI is far more versatile. It efficiently utilizes ACF information in limiting the search space of a problem but does not require it, or any other particular kind of information, except for the molecular formula. In the absence of one or more ACF families (whose central atom represents one of the non-hydrogen atoms in the molecule), the substructure representation in HOUDINI may not account for all of the actual atoms. In this case, the SR agent completes the construction of the required substructures in the hyperstructure without building a complete molecular structure. At that stage, control of structure generation is handed to another managing constraint agent, the *bonding agent*. Its function is to fix those remaining possible bonds in the search space of the hyperstructure that lead to all molecular structures compatible with the constraints.

In HOUDINI, the process of substructure atom labeling, bond fixing, and solution space reduction continues until a molecular structure compatible with the required substructures, any forbidden substructures and strained structural features, and any other applied constraints, is generated. In fact, the process makes use of the entire range of constraints originally developed for ASSEMBLE. Convergent structure generation is a recursive process and *exhaustively* leads to all compatible structures.

Parallel Alternative Substructure Processing. The presence of a large number of families of alternative required substructures has become increasingly common in the structure elucidation of compounds of complex structure. Each such family represents ambiguous information. Two-dimensional NMR experiments in particular can generate large numbers of atom−atom correlations each of which may be consistent with two or more different bonding arrangements. Additionally, as indicated earlier, it is not uncommon for a single family of alternative ACFs in the ACF Shortlist to contain a hundred or more ACFs.

Efficient processing of a large number of families of alternative inferences is central to a *practical*, computer-based structure elucidation system. However, a common feature of structure generators which have been described is the sequential processing of alternative required substructures; at its best, an inherently inefficient process. The method is comparable to creating an unambiguous set of substructures in an initial step by selecting a single substructure from each family of alternative substructures, followed by a structure generation procedure which uses the substructures in the unambiguous set sequentially in some order (Figure 6). The process resembles a tree search using "try-and-fail" strategy to find all possible combinations of alternative substructures satisfying all of the constraints. However, the total search space can become enormous as the number and size of families of alternative substructures are increased. Furthermore, the heuristics used to control the order of selection of substructures within a family, and then the order in which the substructures in the resulting unambiguous set are sequentially applied, are generally not optimized to maximize the prospectiveness of the search.

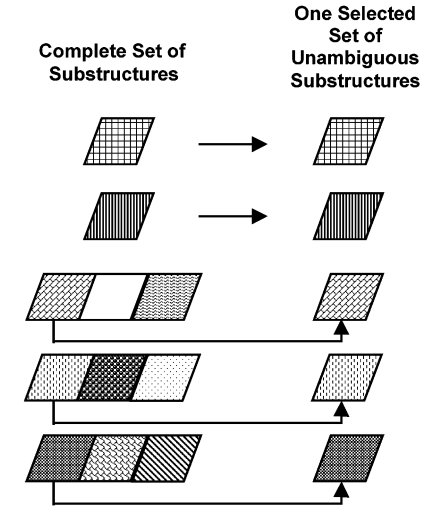

Figure 6. Conventional processing alternative substructural inferences.

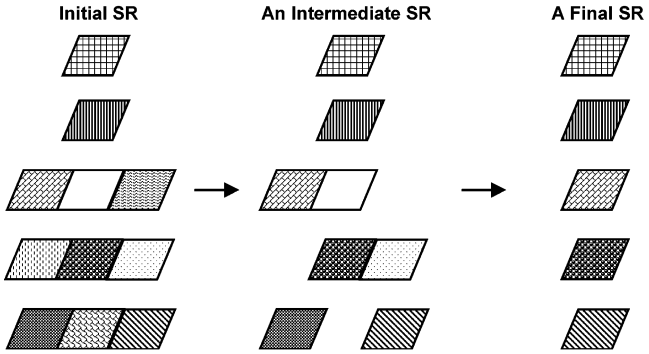

Figure 7. Processing of alternative substructural inferences in HOUDINI.

HOUDINI incorporates a new method called *parallel alternative substructure processing* (PASP). In contrast to earlier methods, the alternative required substructures within a family—which in HOUDINI are all integrated into the single SR—are used directly in structure generation without substructure selection. The process is illustrated in Figure 7. At various steps in the structure generation procedure, alternative substructures are "dropped out" in a manner analogous to structure reduction. There is no explicit "try-and-fail" step in PASP; alternative substructures within a family work in concert with other substructures to generate complete molecules. The end result is the same; each plausible molecular structure generated includes a single substructure from each family of alternatives, but it is achieved by a more efficient, prospective processing of the information.

The parallel atom group (PAG) is the structural representation of alternative substructural inferences used in HOUDINI. Its application to a family of alternative ACFs is illustrated by means of Figure 5/Table 1. The chemical environment of carbon atom ($-CH_2-$) with chemical shift 38.8 ppm (atom A2) is described by a family of two ACFs: A18-A2-A19 and A18-A2-A20, i.e., C38.8 joins to a CH group (A18) and either bromine (A19) or divalent sulfur (A20). Each of the two sets of first-layer atoms, (A18/A19) and (A18/A20), is an alternative neighborhood (AN), and together they comprise a PAG.

The treatment of substructure atoms belonging to a PAG differs only in a few aspects from the treatment of normal substructure atoms. Some of the heuristics influencing the choice of the substructure atom to be labeled depend on information related to the nature of the PAG, e.g., the number of alternative substructures sharing a substructure atom. Aside from these heuristics, the selection of the SR agent does not discriminate between normal substructure atoms and those belonging to a PAG. Thus, all substructure atoms compete directly with one another in being labeled. This competition occurs uniformly throughout structure generation without any restrictions. Moreover, all substructures, including alternative substructures, compete among each other. An alternative substructure may temporarily "lose" to others by having fewer of its atoms singly labeled but eventually "win" by outliving all others in the "race" to a final solution.

The strategy of HOUDINI's PASP offers two major advantages compared to the predefined order methods. First, the hyperstructure will reflect the common substructures of the remaining ANs in a PAG as soon as possible, thus speeding up structure generation. For example, in the case of an ACF family consisting of several different ester and amide functions, the common carbonyl substructure of that family leads to a fixed bond at the start of structure generation, and, as soon as one of the two subsets is excluded, e.g., the amide, another fixed bond (the ester linkage) appears in the hyperstructure. Second, if an attempt to label a substructure atom belonging to an ACF empties the domain of this substructure atom, all ACFs which contain this substructure atom are removed from the family. Therefore, PASP excludes many of the irrelevant alternative substructures earlier in the structure generation than methods relying on a predefined order. For example, COCOA alternates between selecting an ACF and reducing the hyperstructure based on the remaining constraints. Other systems, e.g., LSD, treat alternative required substructures sequentially.

In implementing PASP, an additional value—the *null* value—is initially placed in the domain of each substructure atom in the SR that is part of a PAG. Removal of all of the actual atoms in the domain of a substructure atom results in assignment of the null label. This in turn results in the removal of that substructure atom and those ANs in the PAG in which it is present. Those substructures are no longer considered in the ongoing structure generation process until the null value is reset, i.e., until backtracking to a previous problem state occurs. Removing a particular alternative substructure does not necessarily require the removal of all of its substructure atoms from the SR. That same substructure atom may be present in one or more ANs of other PAGs. For every atom which is part of an alternative substructure,

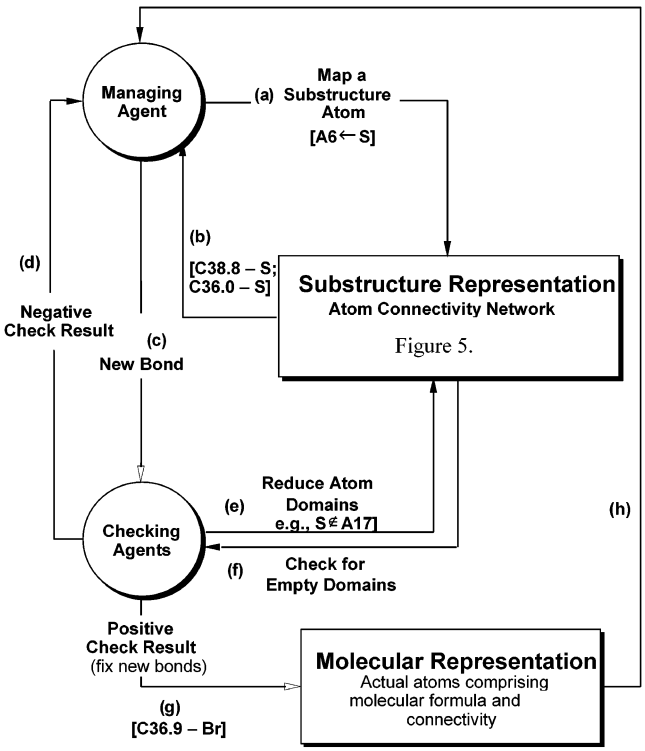

Figure 8. Structure generation in HOUDINI.

HOUDINI maintains a special counter which tracks the current number of remaining alternative substructures containing that substructure atom. As soon as the counter reaches zero, the substructure atom is removed from the SR together with all of the bonds connecting this atom to remaining substructure atoms, even though the domains of these atoms may contain some valid actual atom values other than null. To minimize unnecessary computation, as soon as all of the atoms in a particular alternative substructure are singly labeled with actual atoms, the remaining alternative neighborhoods of that PAG are removed from consideration since the constraint has been satisfied. Of course, the removed substructures are restored if backtracking to a previous state occurs.

The nature of interactions between the constraint agents and the main data structures in HOUDINI can be illustrated using Figures 5 and 8. Figure 5, as indicated earlier, depicts the initial state of an SR (see **Computer Representation** above) in which the domain of each of the substructure atoms in the network is described in Table 1. Some atoms (e.g., A2) have a number of neighbors exceeding their valence, indicating that their chemical environments are represented by a PAG.

In this example, the SR agent plays the role of managing agent. In the first step (path a, Figure 8), the SR agent selects a substructure atom to label—for example, atom A6—and then labels it with one of the actual atoms in its domain, divalent sulfur in this case [A6←S]. Next, the program determines which *already-labeled* substructure atoms are bound to the newly labeled substructure atom A6: A2 and A5, carbon atoms C38.8 and C36.0, respectively. The corresponding list of new bonds—C38.8-S and C36.0-S—is then reported to the managing agent (path b). These new bonds are then passed, one by one, to the checking agents (path c) which perform two tasks. First, they test for

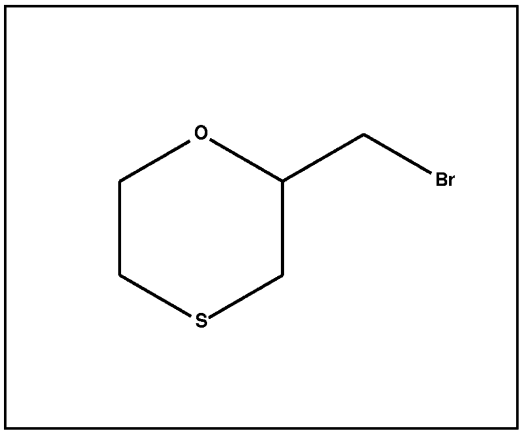

Figure 9. Structure of the unknown of molecular formula C_5H_9BrOS.

consistency with the global constraints and forbidden substructures. If a violation is detected, the checking agent returns control to the managing agent (path d) which then removes the label assigned to A6 and chooses another.

Second, checking agents remove labels from the domains of substructure atoms which are rendered inconsistent with the newly chosen label and then check for empty domains (path e and f)). For example, if A6 is labeled with the only sulfur atom in the molecule, then sulfur can be removed from the domain of A17 [S∉A17] since it is joined to an atom (A3) other than A2 or A5. This empties the domain of A17 and an empty domain of a substructure atom normally leads to a violation and backtracking; however, in this case, the substructure atom A17 is part of a parallel atom group and its domain includes a null value. Since the removal of all atom labels from the domain still leaves a value, the null value, the domain is not "empty" and no violation is reported. However, A17 is no longer considered to be part of the SR, thereby assigning ACF A16-A3-A15 to substructure atom A3. Should backtracking occur at a later step, A17 will be restored to its original state. There are other consequences of labeling A6 as sulfur. Bromine can be removed from A19 (the null value remains) since A2 (a CH_2 group) is valence-satisfied if joined to A1 and A6. The neighbors of A6 (A2 and A5) do not conflict with the comparable information contained in the sulfur-centered ACF, A25-A24-A26. As A19 is no longer considered (see above), ACF A18-A2-A20 can be assigned to substructure atom A2. The structure of the unknown is shown in Figure 9.

In the absence of any detected violation by the checking agents, the two new bonds are fixed in the hyperstructure (path g) and control is returned to the managing agent (path h). If the checking agents can infer additional required bonds in the hyperstructure, e.g., to preclude a forbidden substructure, those are fixed as well before control is returned to the managing agent.

Program Performance. An initial version of the program was developed to determine the practicality of reducing the concept of convergent structure generation to a program of higher performance than one based on structure reduction. The program was coupled to the spectrum interpretation program INTERPRET, and the performance of this HOUDINI-based version of SESAMI was compared to that of the latest version of COCOA-based SESAMI. Naturally occurring compounds of diverse structure were used in the comparison.[30]

Two measures were of particular importance: execution time and the extent to which program performance was degraded with increasing ambiguity in the collective substructural input. The latter was considered to be especially significant because of the extensive use of those 2D NMR experiments which give rise to atom−atom correlations in which the number of intervening bonds cannot be assigned with certainty. For the purpose of establishing the viability of convergent structure generation in the shortest time possible, the version of the HOUDINI program used in this performance study was not optimized. Despite this, HOUDINI demonstrated faster execution times than COCOA and less degradation in performance with the increasing ambiguity in the collective substructural input that often accompanies an increase in the number of atoms in the unknown.

SUMMARY AND CONCLUSIONS

A new method of structure generation called convergent structure generation has been devised which combines elements of both structure assembly and structure reduction. In particular, program HOUDINI incorporates three new features to address the shortcomings of current structure generators and improve program efficiency. First, the *entire* set of required substructural inferences—both explicitly and ambiguously defined—is initially expressed a single, integrated substructural representation. This integrated representation is richer in information content than the set of individual inferences since it reveals structural relationships between the individual substructures. Second, in convergent structure generation, the integrated structural representation is used to *simultaneously* construct required substructures and to reduce the search space of the problem in a process that is *managed* so as to maximize the efficiency of convergence. Third, parallel alternative substructure processing allows the inferences in even a large family of alternative inferences to be processed simultaneously rather than sequentially, and, at the same time, along with unambiguous substructural inferences.

Preliminary experience in problem solving suggests that convergent structure generation enhances performance of structure elucidation systems relative to those built around structure reduction based structure generators.

ACKNOWLEDGMENT

The financial support of this research by the National Institutes of Health (Grant GM62457) is gratefully acknowledged. Additionally, without the responsive computer support of Frank Davis and Mary Rushton of the Electronics Shop in the Department of Chemistry and Biochemistry, the timely execution of this project would not have been possible.

REFERENCES AND NOTES

(1) Shelley, C. A.; Hays, T. R.; Munk, M. E.; Roman, R. V. An Approach To Automated Partial Structure Expansion. *Anal. Chim. Acta* **1978**, *103*, 121−132.

(2) Masinter, L. M.; Shridharan, N. S.; Lederberg, J.; Smith, D. H. Applications of Artificial Intelligence for Chemical Inference. XII. Exhaustive Generation of Cyclic and Acyclic Isomers. *J. Am. Chem. Soc.* **1974**, *96*, 7702−7714.

(3) Kerber, A.; Laue, R.; Grüner, T.; Meringer, M. MOLGEN 4.0. *Match* **1998**, *37*, 205−208.

(4) Shelley, C. A.; Munk, M. E. CASE, A Computer Model of the Structure Elucidation Process. *Anal.Chim. Acta* **1981**, *133*, 507−516.
(5) Carhart, R. E.; Smith, D. H.; Gray, N. A. B.; Nourse, J. G.; Djerassi, C. GENOA: A Computer Program for Structure Elucidation Utilizing Overlapping and Alternative Substructures. *J. Org. Chem.* **1981**, *46*, 1708−1718.
(6) Christie, B. D.; Munk, M. E. Structure Generation by Reduction: A New Strategy for Computer-Assisted Structure Elucidation. *J. Chem. Inf. Comput. Sci*. **1988**, *28*, 87−93.
(7) Christie, B. D.; Munk, M. E. The Application of Two-Dimensional Nuclear Magnetic Resonance Spectroscopy in Computer-Assisted Structure Elucidation. *Anal. Chim. Acta* **1987**, *200*, 347−361.
(8) Funatsu, K.; Miyabayashi, N.; Sasaki, S. Further Development of Structure Generation in the Automated Structure Elucidation System CHEMICS. *J. Chem. Inf. Comput. Sci*. **1988**, *28*, 18−28.
(9) Blinov, K. A.; Elyashberg, M. E.; Molodtsov, S. G.; Williams, A. J.; Martirosian, E. R. An Expert System for Automated Structure Elucidation Utilizing Proton−Proton, Carbon-Proton and Nitrogen−Hydrogen 2D NMR Correlations. *Fresenius J. Anal. Chem.* **2001**, *369*, 709−714.
(10) Lindel, T.; Junker, J.; Kock, M. COCON: From NMR Correaltion Data to Molecular Constitution. *J. Mol. Model.* **1997**, *3*, 364−368.
(11) Carabedian, M.; Dubois, J.-E. Inferring Extended Virtual Knowledge from an EPIOS Conversion Graph of Overlapping Substructures. *J. Chem. Inf. Comput. Sci*. **1994**, *34*, 701−706.
(12) Will, M.; Fachinger, W.; Richert, J. R. Fully Automated Structure Elucidation − A Spectroscopist's Dream Come True. *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 221−227.
(13) Funatsu, K.; Sasaki, S. Recent Advances in the Automated Structure Elucidation System, CHEMICS. Utilization of Two-Dimensional NMR Information and Development of Peripheral Functions for the Examination of Candidates. *J. Chem. Inf. Comput. Sci*. **1996**, *36*, 190−204.
(14) Funatsu, K.; Katsumi, M.; Sasaki, S. Computer Program for Predicting the Number ofCarbon-13 NMR Signals Based on Chemical Structure. *Comput. Enhanced Spectrosc.* **1986**, *3*, 87−90.
(15) Elyashberg, M. E.; Blinov, K. A.; Williams, A. J.; Martirosian, E. R.; Molodtsov, S. G. Application of a New Expert System for the Structure Elucidation of Natural Products from Their 1D and 2D NMR Data. *J. Nat. Prod.* **2002**, *65*, 693−703.
(16) Lindel, T.; Junker, J.; Kock, M. 2D NMR-Guided Constitutional Analysis of Organic Compounds Employing the Computer Program COCON. *Eur. J. Org. Chem*. **1999**, *3*, 573−577.
(17) Kock, M.; Junker, J.; Maier, W.; Will, M.; Lindel, T. A COCON Analysis of Proton-Poor Heterocycles − Application of Carbon Chemical Shift Predictions for the Evaluation of Structural Proposals. *Eur. J. Org. Chem*. **1999**, *3*, 579−586.
(18) Maier, W. New Approaches to Computer-Aided NMR Interpretation and Structure Prediction. *Computer-Enhanced Analytical Spectroscopy;* Wilkins, C. L., Ed.; Plenum Press: New York, 1993; Vol. 4, pp 37−55.
(19) Christie, B. D.; Munk, M. E. The Role of Two-Dimensional Nuclear Magnetic Resonance Spectroscopy in Computer-Enhanced Structure Elucidation. *J. Am. Chem. Soc.* **1991**, *113*, 3750−3757.
(20) Madison, M. S.; Schulz, K.-P.; Korytko, A. A.; Munk, M. E. SESAMI: An Integrated Desktop Structure Elucidation Tool. *Internet J. Chem.* **1998**, *1*, Article 34.
(21) Bohanec, S.; Zupan, J. Structure Generation of Constitutional Isomers from Structural Fragments. *J. Chem. Inf. Comput. Sci*. **1991**, *31*, 531−540.
(22) Peng, C.; Yuan, S.; Zheng, C.; Hui, Y. Efficient Application of 2D NMR Correlation Information in Computer-Assisted Structure Elucidation of Complex Natural Products. *J. Chem. Inf. Comput. Sci*. **1994**, *34*, 805−813.
(23) Faulon, J.-L. Stochastic Generator of Chemical Structure. 2. Using Simulated Annealing to Search the Space of Constitutional Isomers. *J. Chem. Inf. Comput. Sci*. **1996**, *36*, 731−740.
(24) Steinbeck, C. SENECA: A Platform-Independent, Distributed, and Parallel system for Computer-Assisted Structure Elucidation in Organic Chemistry. *J. Chem. Inf. Comput. Sci*. **2001**, *41*, 1500−1507.
(25) Meiler, J.; Will, M. Automated Structure Elucidation of Organic Molecules from ^{13}C NMR Spectra Using Genetic Algorithms and Neural Networks. *J. Chem. Inf. Comput. Sci*. **2001**, *41*, 1535−1546.
(26) Tsang, E. *Foundations of Constraint Satisfaction*; Academic Press: London; San Diego, 1993.
(27) Nuzillard, J.-M.; Naanaa, W.; Pimont, S. Applying the Constraint Satisfaction Problem Paradigm to Structure Generation. *J. Chem. Inf. Comput. Sci*. **1995**, *35*, 1068−1073.
(28) Munk, M. E.; Madison, M. S.; Schulz, K.-P.; Korytko, A. SESAMI: A Program for the Reduction of Spectral Properties to Chemical Structure. CIC, Thirteenth Workshop, Nov. 13−15, 1998, Bad Dürkheim, Germany. www.ccc.uni-erlangen.de/cic/workshop98/paper8.html.
(29) Peng, C.; Bodenhausen, G.; Qui, S.; Fong, H. H. S.; Farnswort, N. R.; Yuan, S.; Zheng, C. Computer-Assisted Structure Elucidation: Application of CISOC−SES to the Resonance Assignment and Structure Generation of Betulinic Acid. *Magn. Reson. Chem.* **1998**, *36*, 267−278.
(30) Schulz, K.-P.; Korytko, A.; Munk, M. E. Applications of a HOUDINI-Based Structure Elucidation System. *J. Chem. Inf. Comput. Sci*. **2003**, *43*, 1447−1456.

CI034057R