

## 2D Structure Depiction

Alex M. Clark,\* Paul Labute, and Martin Santavy

Chemical Computing Group, Inc., 1010 Sherbrooke Street West, Suite 910,  
Montréal, Québec, Canada H3A 2R7

Received December 23, 2005

A comprehensive algorithm for the depiction of 2D coordinates of chemical structures is described. The methods used represent a significant improvement to the state of the art with regard to molecular connection graphs which pose particular difficulty to most layout efforts. Resulting coordinates are consistently of publication quality for a large subset of chemistry. The algorithm is discussed in detail, and measurements of its overall success are presented.

### INTRODUCTION

The fields of cheminformatics and combinatorial chemistry have in recent years allowed scientists to generate vast libraries of chemical structures in various formats. One of the many challenges involved with managing such data is the absence of 2D coordinates suitable for aesthetically presentable output. In cases where reasonable 3D coordinates are available, structures can be visualized using appropriate tools, but this is considerably more difficult for the casual observer than perusing a well-drawn diagram that adheres to conventional structure drawing styles. Worse, much data is stored in a format which is equivalent to a connection table (e.g., SMILES<sup>1</sup>), with no coordinates of any kind, which makes structure viewing infeasible without a software algorithm for generating a meaningful layout.

Considerable prior art exists on the subject of 2D structure generation.<sup>2</sup> The demand for diagram depictions of molecular structures is great, because the stylistic conventions of such pictograms constitute a “natural language” for chemists which can be rapidly comprehended, and is thus highly suitable for communication. Yet despite this importance, the problems involved in producing a fully automated depiction algorithm remain only partially solved. While some of this is due to the conventions for 2D representations of 3D molecular fragments being sometimes arbitrary, the conversion of any graph to an optimal planar layout adhering to a set of rules is a fundamental challenge in computer science.

Fortunately, chemical structures constitute a limited subset of all possible graphs. If one considers only organic chemistry, there can be formulated a relatively small set of rules which cover most of the common environments in which an atom type might occur. Once these rules are stated, the structure can be pieced together and an overall depiction obtained.

Simple algorithms can achieve remarkable success for a moderate fraction of organic molecules. Nonetheless, there are diminishing subsets of chemical structures which pose ever greater obstacles to a depiction algorithm:

(1) Molecules with significantly congested regions require that placement decisions consider global as well as local consequences.

(2) Often, the conventions of structure diagram layout cannot be adhered to, and so, the layout algorithm must be able to decide when and how to compromise.

(3) Most molecules are not planar, and some structures cannot be drawn as if they were; therefore, portions of the structure must be drawn in a stylized pseudo-3D form.

(4) Nontrivial ring blocks are present in many classes of chemical structures, and methods for finding aesthetic layouts are crucial.

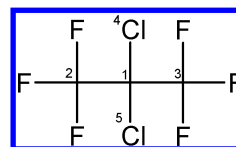
(5) Attempts to define a quantitative metric of success are thwarted by the fact that the quality of the outcome is ultimately subjective.

Previous efforts have typically fallen within one of the following two broad design categories:

1. *Partitioning and Sequential Placement.* The structure graph is partitioned into fragments, each of which is reconstructed and attached in some priority order. Where ideal placement has failed, some kind of nonideal correction must be made.

2. *Energy Minimization.* A goal is stated, that being a well-spaced layout of atoms, ideally with regular spacing and few overlapping bonds. A dynamic iterative optimization procedure is applied. An initial embedding method is required, in lieu of rule-based molecule partitioning.

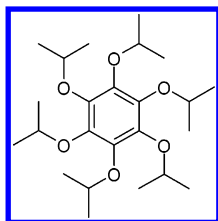
Most depiction algorithms are based on the partitioning and sequential placement method, because good results for many structures can be achieved quickly with relatively little investment of effort. Usually, global vs local decisions are resolved by using a priority weighting of atoms based on the extent to which each is “buried” within the connection graph. Use of the priority rank can often force critical layout decisions to be made early in the layout process, such that remaining placements work around them. For example, in the following structure



the most central carbon atom (1) can be marked for first placement. It has four substituents, two of which are more highly congested (2 and 3) and two of which are terminal

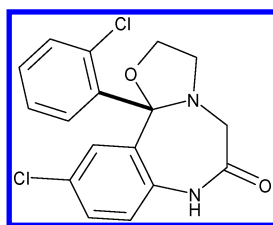
\* Corresponding author phone: (514) 393-1055; fax: (514) 874-9538; e-mail: aclark@chemcomp.com.

(4 and 5). A localized decision can be made to place the most congested atoms opposite one another in the hope that this will achieve the optimal result. While bluntly effective, such a naive approach is too limited to capture many species in organic chemistry. For example, the following structure

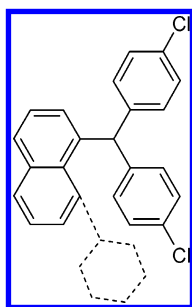


is a case in which a priority congestion algorithm would be of no assistance. Within the context of chemical diagram ideals, the degrees of freedom are the ether C—O—C hinges, which can be assigned one of two directions, (+) or (−). Of the 64 possibilities, only two outcomes match the diagram above. The answer would be simple if the traversal were done in a specific order about the ring, but a sequential algorithm would need to perceive this, and any number of more complex cases. Rather, to be properly general, the algorithm would need to consider the long-term implications of every placement decision. There is no obvious way to implement such an algorithm without an exponential time dependence on graph size.

Most often, the difficulties of reconciling local vs global layout decisions are ignored, and the effort to prevent overlap is instead made using fallback methods for finding a nonideal solution, for example, by stretching or rotating bonds. In the following example



the overlap can be attributed to one bond (shown in bold), which can be rotated, stretched, or flipped to obtain a pleasing result. While this is sometimes effective, there may be at this late stage no simple fix that averts overlap well enough. The following example shows a simple instance where the sequential joining of fragments would likely run into trouble:



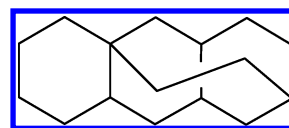
If the naphthalene ring were to be placed first, the next logical substituent would be the 3° aliphatic carbon atom, which is the next most deeply buried atom. Placement of its two *p*-chlorophenyl substituents as above would seem like a

logical next step, except that the final (cyclohexyl) substituent becomes blocked. The adjustments required to avoid overlap are displeasing compared with some of the alternative compromises of which a skilled operator might conceive.

Given the difficulties encountered with sequential algorithms, the appeal of designing a concise and general optimization-based algorithm is high. The problem, however, cannot be simply treated as a special case of molecular mechanics force fields, not in the least because of the lack of a good continuous and differentiable energy function based on aesthetic criteria. If such a function could be found, the remainder of the burden would be transferred to the initial embedding of coordinates, and methods to jump over local minima.

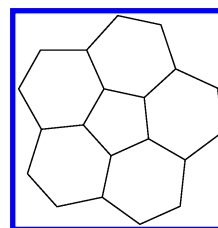
Optimization-based algorithms do exist,<sup>3,4</sup> but they tend to overly favor sparse layout at the expense of stylistic conventions. An ideal layout of a chemical diagram usually consists of regular angles, each with only several discrete degrees of freedom, and the most pleasing solution is not necessarily the most disperse. When a good solution cannot be composed of locally ideal junctures, suitable compromises must be made, which is difficult to express as an objective function.

Algorithms in general use have serious difficulty with nontrivial ring blocks. Sequential algorithms typically take a practical approach: because there is usually at least some core of the ring for which simple trigonometry gives an ideal answer, one approach is to detect this core, then draw the remaining fragments of the ring block around it. This example



has been drawn by using a planar embedding for the three edge-fused cyclohexane rings, followed by different logic for adding the ethylene bridge.

Many ring blocks which have an agreeable planar embedding are still difficult to arrange with sequential trigonometry, especially those which are highly fused and not strictly regular, such as



In this case, there is no way to construct the ring block without distorting at least one of the five- or six-membered rings. The drawing shown above, which maximizes symmetry, happens to be the most aesthetically pleasing, yet an algorithm which operates by placing the atoms or rings one at a time would be unlikely to accomplish this layout.

The use of predrawn templates is a popular approach to dealing with ring blocks, and it is effective because a modest database of fused ring systems can capture a high portion of organic chemistry. The utility of templates can be further extended by removing terminal edge rings, yet even the relatively straightforward connection graphs of drug-like

molecules contain a small but significant proportion of ring systems which cannot easily be depicted by combining simple trigonometry with a lookup list. The incidence of highly connected but planar small rings, large macrocyclic rings, and 3D ring systems makes the problem a serious one which must be treated with sophisticated algorithms.

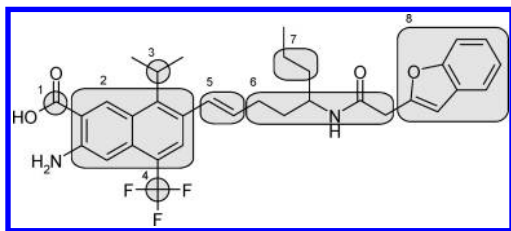
The success of any depiction method can be stated in terms of two goals: (1) to maximize the fraction of molecular connection graphs that are readily discernible to the observer and (2) to maximize the fraction of graphs which are aesthetically ideal and functionally equivalent to the efforts of a skilled artist. An algorithm which produces a high rate of output which is intelligible but not ideal is useful for tasks such as browsing catalogs, whereas one which sacrifices generality in order to achieve quality is useful for presentation purposes and can significantly alleviate a labor-intensive task.

In the remainder of this paper, we present an algorithm which addresses both of these goals and discuss in detail the methods used and the degree of success obtained. Our algorithm takes as its input a molecular connection table, which is restricted to simple atom and bond properties (element, charge, bond order, etc., but no pre-existing coordinates), and produces as its output two arrays (X,Y) which describe the layout of the molecule in the plane. Combining the input and output is sufficient information to produce a visual diagram.

## METHODS

The primary consideration in the design of a high-quality structure depiction method was how to reproduce the high incidence of aesthetically ideal results achieved by sequential ruled-based algorithms while addressing the unsolved problems of efficiently finding ideal solutions when they exist and proposing reasonable compromises when they do not. We also anticipated considerable difficulty with ring systems, for which the treatment must be robust and comprehensive.

Partitioning algorithms begin by dividing the molecular graph into a variety of components. This treatment we use as the first step, and the components remain separated until the final stages of the depiction process. Consider the retroactive analysis of the following structure:



The graph has been partitioned according to the following rationale:

1. Hydrogen atoms are ignored until the final stages of the algorithm.
2. Terminal atoms are not important, because they are implied by their neighbor.
3. Ring blocks (2 and 8) are detected and their rings grouped together.
4. Stereochemically active double bonds (5) are identified and restricted appropriately.
5. Sequences of certain atoms are treated as chains (6).
6. Remaining atoms are paired (7) or isolated (1, 3, and 4).

Once the graph has been partitioned, each block has assigned to it a set of geometric constraints, which is discussed in detail below. Each constraint is an array of angles and distances sufficient to describe all of the atoms within the block relative to one another and the attachments to atoms outside of the block.

The 2D structure is obtained by searching for the best overall solution to the constraints, thus avoiding the problem of premature local decisions locking the depiction process into a globally undesirable outcome. We populate the constraints with relatively few alternatives, each aesthetically ideal within its own context. Under the resulting constraints, essentially any solution which maximizes spread and minimizes overlap is likely to be a high-quality outcome.

A number of approaches were considered for finding the optimal combination of the constrained choices. We established empirically that the selection of 100 weighted random combinations, followed by localized refinement, was sufficient. Failure to produce a high-quality result was almost exclusively due to the formulation of the constraints, not to the selection of a suboptimal combination. Similarly, a simple congestion function was found to be a sufficiently good metric to guide the selection of constraint solution sets.

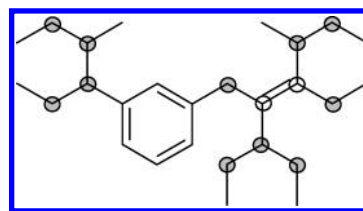
Several postprocessing steps are required, which include the rotational alignment and relative placement of isolated fragments and the placement of hydrogen atoms. Strategies are also required for cases in which the best-found solution to the constraints is still not an acceptable result, for instance, when the possibility space includes only solutions which have overlapping atoms or bonds.

**1. Atoms.** Each atom which is (1) not a terminal atom attached to a nonterminal atom, (2) not assigned to a chain sequence, and (3) not in a ring block is initially assigned a local constraint on the basis of its properties and those of its immediate neighbors. The local constraint is expressed as internal coordinates, where the bond distances are set to the depiction ideal (1.2 Å) and relative angles between the substituents are obtained from a lookup table or calculated using simple trigonometry.

Figure 1 shows the contents of a pattern file which is used to match individual atoms to plausible local geometries. The atom and its bonds to immediate neighbors are encoded in a pattern, some of which include atom and bond wildcards. The first successful match is used. If no patterns are matched, the bond angles are calculated by evenly dividing 360° by the number of neighbors.

In some cases, several patterns apply. Those which are more favorable are given higher weightings. Mirror images of each coordinate set are also included.

**2. Pairs.** To reduce the degrees of freedom and introduce some additional layout bias, adjacent atoms are combined into pairwise constraints whenever possible. In the following example



atoms with local constraints are outlined. The matching is

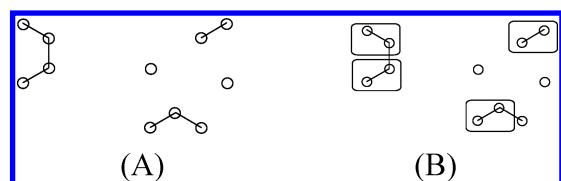
Pattern Geometries		

**Figure 1.** Atom patterns used for local constraints. Each pattern is used to match the central atom with its immediate substituents. Elements may be specified explicitly or with Q for any atom and X for any halide. Bond orders may be explicit or any (dotted line). Weightings are 1.0, unless shown in brackets.

Inputs	Outputs			
F/C=C/F				
CCCC				
CC(C)CC				

**Figure 2.** Selected examples of pairing between adjacent atom constraints.

done by first identifying atom pairs which share a stereochemically restricted bond (the open circles shown above). These are always paired and then removed from the set, which results in a subgraph, as per this example



shown in A. By finding a maximum matching of the molecular graph,<sup>5</sup> an optimum set of pairs can be obtained, such as is shown in B, which indicates four pairs of atoms and three atoms which remain isolated. Including the alkene, a total of five pairs of atoms are matched.

A constraint is created from each atom pair by combining the terms of the constituent atoms, with an additional overall weighting term. Figure 2 shows several examples of paired constraints, each of which has substituents with two possible

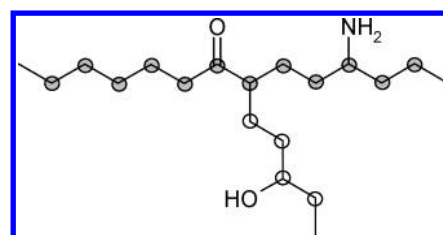
local choices. The first example shows *trans*-difluoroethene, for which half of the possible internal coordinate combinations violate the stereochemistry and so are excluded. The second example is a pairing of two secondary carbon atoms, for which any of the combinations is valid, but an aesthetic bias, introduced by changing the weightings, favors those which produce a zigzag linear sequence. The third example is given no weighting bias, because none of the combinations is locally preferable.

**3. Chains.** Certain atoms are candidates for inclusion in chain blocks. Elements C, N, O, and S are eligible if they are neutral and connected by a single bond, have either two or three heavy atom neighbors and no triple bonds, and do not belong to a ring block. Carbon is also eligible if it has four attachments and at least two of its neighbors are terminal halogens.

Chains are assigned by searching for the longest sequence of eligible atoms. If the path is of length 4 or more, a chain constraint comprised of the sequence is constructed. The atoms from the new chain are marked ineligible, and the search is repeated until no more can be found.

The geometric projection of chain blocks is simple: all bond lengths are set as the ideal (1.2 Å). For di- and trisubstituted atoms, the internal coordinates are assigned as alternating  $\pm 120^\circ$  angles along the chain, and additional substituents are projected into the remaining trigonal interstice. For tetrasubstituted halocarbons, two of the halogens are placed opposite one another ( $\pm 90^\circ$ ) and the remaining two substituents collinear to the chain ( $0^\circ$ ,  $180^\circ$ ).

The following example shows the assignment of chains and layout of a graph with two chains, sizes 12 (solid circles) and 4 (open circles):



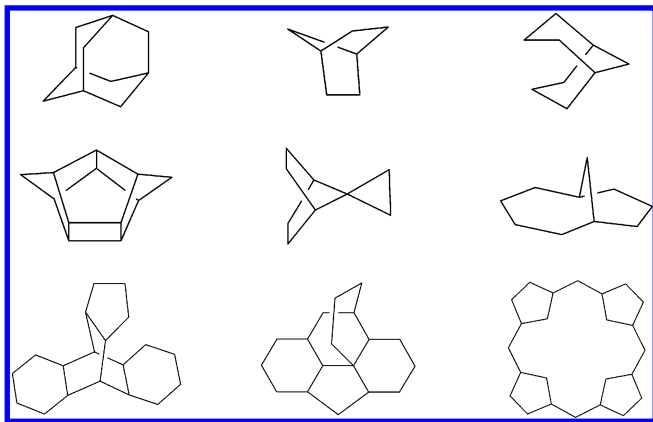
A block constraint is produced for each constructed chain, allowing one degree of freedom for its mirror image.

**4. Rings.** Ring blocks are treated in a specialized manner, as they ultimately have few degrees of freedom, and their layout is most readily facilitated by specific algorithms which are activated depending on the particular topology of the ring block. Each ring block is explicitly generated and expressed in terms of Cartesian coordinates. Once the ring block has been projected, its immediate substituents are placed; then, the member atoms are recast in internal coordinates and assigned as a block constraint.

The assignment of atoms to ring blocks and the generation of the graph of ring adjacencies is done using variations on standard graph algorithms.<sup>6</sup> Enumeration of individual rings within each block is done using an algorithm to detect the smallest set of smallest rings.<sup>7</sup>

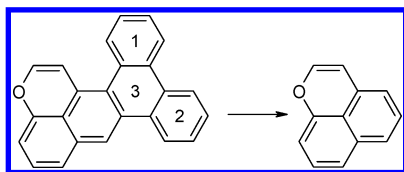
The layout of ring blocks in Cartesian coordinates is described in the sections that follow. Various methods are used, which are activated depending on the ring block topology.





**Figure 3.** Selected examples of templates, showing especially those with a stylized 3D representation, which is difficult to achieve with embedding algorithms.

**Terminal Ring Peeling.** Most ring systems for organic molecules either are very simple or can be made simpler by removing terminal rings. To this end, we use the ring adjacency graph to implement a loop which successively peels off rings which share exactly one edge with the remaining set of rings.



In this example, three rings are peeled off in the sequence indicated. Once only the core three fused rings remain (right), there are no longer any terminal rings suitable for peeling, and so, the sequence terminates.

The majority of drug-like molecules contain only ring blocks for which the ring adjacency graph is a tree, and so, the peeling algorithm reduces the block down to a single polygon, which is constructed using simple trigonometry. Otherwise, the peeling algorithm is used as a preprocessing step for the algorithms described subsequently, except in the presence of a ring with 12 or more edges, in which case the macrocycle embedding algorithm is activated directly.

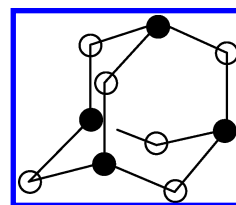
**Template Match.** Lookup in a database of predefined ring templates is the preferred way to reconstruct core ring blocks which cannot be further decomposed by terminal edge ring peeling. The mapping, matching, and retrieval are very fast compared to alternative methods and have the intrinsic advantage of allowing certain common nonplanar ring blocks to be predrawn using their conventional stylized representations. Figure 3 shows several examples of useful templates, many of which are not derivable from an algorithm which presumes the existence of a reasonable planar layout.

The list of core ring blocks for the template database was obtained using an automated script operating on a large number of compound databases. Each ring block was peeled according to the algorithm above, and those which yielded an irreducible core were retained. The resulting set of unique ring blocks was reduced to approximately 300 structures by arbitration, removing particularly macrocycles and blocks with high ring counts and low actual occurrences. The 2D coordinates for each were defined manually and used by the algorithm hereafter.

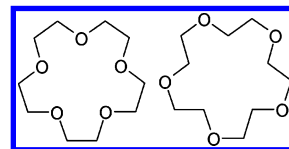
Preprocessing using the ring-peeling algorithm greatly increases the applicable range of the template match, because any number of appended side rings do not interfere with the lookup, provided they are small and do not form rings of rings.

To match the peeled ring block subgraph with one of the predefined templates, the graph nodes are labeled by the results of a graph traversal walk-class algorithm.<sup>8</sup> The same algorithm is preapplied to each ring block template, and so, graphs can be reordered according to their label and mapped directly by comparing their adjacency lists. Identical graphs implies a match, and therefore, coordinates can be transferred.

In some cases, ring blocks have higher graph symmetry than positional symmetry. For example, in the popular representation of adamantane shown below,

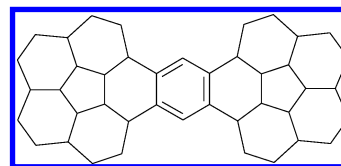


the connection graph has only two types of graph-unique carbon atoms (indicated above by the solid and empty circles), yet there are no symmetry elements in the 2D drawing thereof. The presence of substituents makes the degenerate mappings inequivalent. A set of graph automorphisms is generated,<sup>9</sup> up to a cutoff point. Each permutation is scored so as to bring heteroatoms closer to the center and encourage exosubstituted ring atoms to be oriented outward. Crown ethers are another example of structures particularly sensitive to the choice of permutation



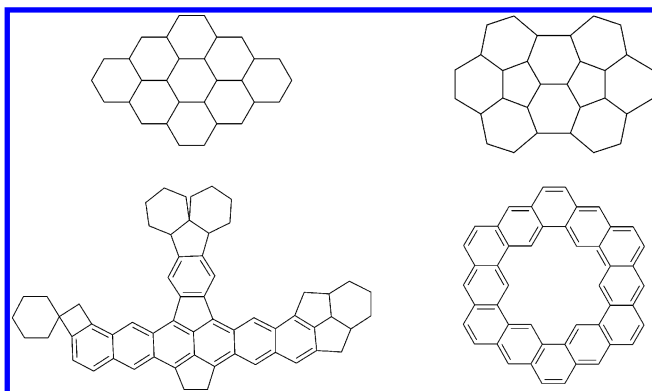
in which case the inward orientation of the heteroatoms (as shown in the left structure) is highly preferred.

**2D Embedding.** While the majority of ring blocks in drug-like small organic molecules can be dealt with quickly by combining templates and ideal polygons, alternative methods are employed when this is not possible. For example, highly connected ring structures such as



are rare enough that they are unlikely to be in a template database. Unless the ring block is comprised entirely of polygons which tessellate the plane, simple trigonometry is likely to produce a distorted or warped layout.

Our present method uses the techniques of distance geometry<sup>10</sup> to depict complex planar ring blocks (without macrocycles). The embedding algorithm proceeds as follows:



**Figure 4.** Selected examples of ring systems produced by the planar embedding algorithm.

1. Create a distance matrix estimate, **A**, in which  $A_{ij}$  is the length of the shortest path between atoms  $i$  and  $j$  in the ring block molecular graph.

2. Use matrix diagonalization to calculate 2D Cartesian coordinates  $x_i$  for each atom  $i$  such that the Cartesian distance matrix  $\{D_{ij} = |x_i - x_j|\}$  best reproduces (in a least-squares sense) the matrix  $\{A_{ij}\}$ .

3. Use a truncated Newton nonlinear optimization algorithm<sup>11</sup> to refine the coordinates by minimizing the energy function

$$f(x_1, \dots) = \sum_{\text{pairs } i-j} C_{ij} (|x_i - x_j|^2 - d_{ij}^2)^2$$

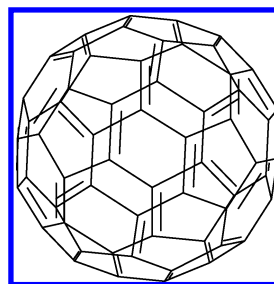
where the sum applies only to bonds, angles interior to rings, and torsions in six-membered rings;  $d_{ij} = 1.2$  for atoms related by a bond,  $d_{ij}^2 = 2(1.2)(1.2)[1 + \cos(2\pi/n)]$  for atoms related by an interior angle of a ring of size  $n$ , and  $d_{ij} = 2(1.2)$  for atoms related by a torsion; the  $\{C_{ij}\}$  are empirically determined proportionality constants (3 for bonds and 1 for angles and torsions).

The embedding in steps 1 and 2 generally results in good initial coordinates. However, in rare cases of long chains of rings, there is an increasing probability of poor embeddings (e.g., with rings placed on the “wrong side”). For this reason, long chainlike ring blocks are subdivided into smaller ring blocks (containing approximately five rings each), each of which is embedded separately using steps 1 and 2 prior to concatenation and refinement in step 3.

The refinement function of step 3 includes strictly intraring terms, which means that rings are distorted from regular geometry only when the fused ring environment is incompatible with a layout consisting of regular polygons.

Examples of large fused ring systems are shown in Figure 4, none of which can be reduced by peeling to a single ring. While a well-implemented sequential ring-traversal algorithm can produce reasonable results for a large fraction of planar ring systems, the method described above is quite general and robust.

**3D Embedding.** There are some cases where all attempts to depict a ring block in a planar manner fail to give acceptable results. Fullerenes, congested cages, and bridge structures are examples of fundamentally 3D chemical structures for which a perspective style depiction is preferred. A dramatic example is  $C_{60}$



which is almost always drawn in a nonplanar style. While planar embeddings do exist for fullerenes, they are neither popular nor straightforward to obtain by algorithm.

Our method uses a 3D embedding algorithm for caged molecular graphs (in which all atoms have more than two ring bonds) and as a fallback when poor structures result from the 2D embedding algorithm. The 3D embedding algorithm proceeds as follows:

1. Use 3D distance geometry to obtain a set of Cartesian coordinates; this procedure is the same as steps 1 and 2 of the 2D embedding algorithm described above except that the calculation is performed in 3D.

2. Use a truncated Newton nonlinear optimization algorithm to refine the coordinates by minimizing the energy function

$$f(x_1, \dots) = \sum_{\text{pairs } i-j} C_{ij} (|x_i - x_j|^2 - d_{ij}^2)^2$$

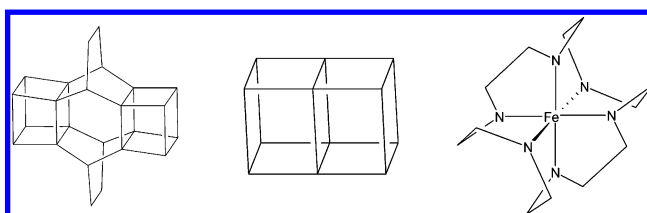
where the sum applies to all bonds and angles;  $d_{ij} = 1.5$  atoms related by a bond, and  $d_{ij}^2 = 2(1.5)(1.5)[1 + \cos a]$  for angles ( $a$  is either  $180^\circ$ ,  $120^\circ$ , or  $109^\circ$ , depending on the atom type); the  $\{C_{ij}\}$  are empirically determined proportionality constants (3 for bonds and 1 for angles).

3. Orient and project the resulting 3D coordinates so that substituents (including fuse points for rings removed by peeling) are at the perimeter of the  $xy$  projection of the structure. To do this, each atom is assigned a weight,  $w_i$ , that is 5 if the atom has substituents and 1 otherwise and the rotation matrix  $R$  that maximizes

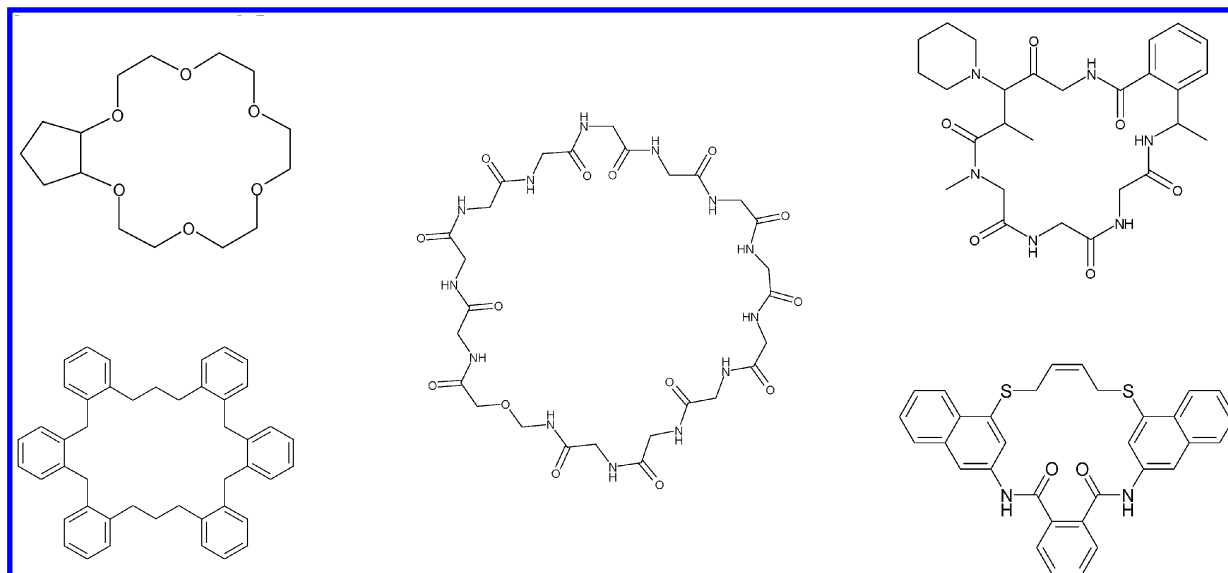
$$f(R) = \sum_i w_i |PR(x_i - x_0)|^2$$

$$x_0 = \sum_i w_i x_i / \sum_i w_i, \quad P = \text{diag}(1, 1, 0)$$

is determined by matrix diagonalization.  $R$  is then perturbed slightly (by sampling) to minimize the number of atoms that would overlap upon projection and to minimize the number of bonds that would appear to be intersecting upon projection. The final coordinates are determined by the transformation  $PR(x - x_0)$ .

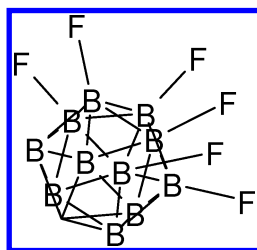


**Figure 5.** Selected examples of ring systems produced by the 3D embedding algorithm.



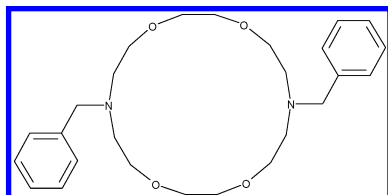
**Figure 6.** Selected examples of structure layouts produced by the macrocycle embedding algorithm, including placement of substituents.

The 3D embedding algorithm generally produces good results (as shown in Figure 5), although some structures are inherently difficult to depict; for example, the following result



has competing requirements of substituent placement and of atom/bond overlap that challenge the orientation procedure of step 3. Generally speaking, substituent placement takes priority over atom/bond overlap.

**Macrocycle Embedding.** Ring blocks which contain one large ring (between 12 and 100 atoms) and some number of smaller rings fused or bridged to it are classified as macrocycles and treated slightly differently. Without special consideration, macrocycles would be formulated as ordinary polygons, bearing increasing resemblance to circles as they become larger. For example the following layout of a substituted 18-crown-6 macrocycle is quite correct, but is a poor depiction:



In our approach, the large ring which defines the ring block as a macrocycle is treated first, by mapping the vertexes onto the outline of a “honeycomb” grid of hexagons. For convenience, templates for rings in the 12–100 size range were preselected and stored in a predefined list. For odd-sized rings, the mapping is made to the next largest template and the last coordinate dropped. Before proceeding, each of the rotational permutations is tested in order to maximize the distance from the center of substituents which are not

part of the macrocycle and minimize that for heteroatoms, which makes chain substituents and fused side rings more tractable.

Each distinct group of rings which is fused with the macrocyclic ring is then projected, using the same matrix diagonalization method as described for 2D embedding. The projected coordinates are connected to the macrocycle at the appropriate attachment points.

Fitness function terms are generated in a similar manner as for the planar 2D embedding method, except that angular terms between atoms forming part of the macrocycle are generated for an ideal of  $120^\circ$  rather than  $360^\circ/N$ . Other fused rings are defined by objective distances derived from regular polygon ideals. Unlike in the planar 2D embedding method, here, it is not sufficient to supply only intraring terms, because of the zigzag nature of the macrocycle. It is quite common for fused rings to overlap the macrocycle itself, unless distance terms are added between atoms of the macrocycle and those of rings which are immediately attached to it, which will force the ring to buckle as much as necessary.

Figure 6 demonstrates the efficacy of the macrocycle embedding algorithm. In each case, the macrocycle backbone is aligned to a regular structure, about which heteroatoms, substituents, and embedded rings are well-positioned.

**Ring Fusion.** If any rings were peeled at the beginning of the procedure, they are reattached in the reverse order from which they were removed. Because they share only a single edge with the ring core, the required trigonometry is trivial: the regular polygon can be drawn onto one of the two sides of the shared edge, and that which minimizes congestion is chosen.

**Substituent Placement.** Because the final translation to internal coordinates requires all of each atoms’ neighbors to be defined as angles and distances, the projection of the immediate substituents must be part of the ring block solution.

Projections are done while still in Cartesian space and are usually a simple matter of maximizing the angular distance from the immediate ring neighbors. In some cases, such as ring junctures or irregular shapes, it is necessary to sample

relative congestion at various candidate positions, and assign varying importance to terminal and nonterminal substituents.

**5. Sampling.** The main loop of the depiction algorithm consists of finding a fixed number of structures by random selection from the available constraints. The default number of sample structures is 100, which is quite adequate and is used for all comparisons described within.

*Internal Coordinates.* Prior to the main loop, all of the possible ways to arrange the structure have been described as weighted collections of constraints regarding individual atoms or blocks of atoms. For each local or block constraint, a random number is generated, which selects a set of internal coordinates, with a probability proportional to the weighting of the set. When a particular choice has been made, internal coordinates are assigned to each of the atoms within the set.

*Cartesian Coordinates.* Once the structure has been fully described by internal coordinates, it is converted to Cartesian coordinates using a simple graph traversal. Different connected components are, for the present, defined to exist in a separate space. The final rotational alignment of each connected component is as yet undefined.

*Candidate Selection.* Each Cartesian solution is measured by applying a simple congestion function, and the solution set with the lowest value is selected. The congestion function for a candidate structure is

$$\text{congestion} = \sum_{i,j} \frac{1}{\text{dist}^2(i,j) \times \text{weight}(i) \times \text{weight}(j)}$$

where  $(i,j)$  are all the nonbonded atoms belonging to the same connected component;  $\text{dist}^2(i,j)$  is the square of the distance between  $i$  and  $j$  and is bounded at a minimum of 0.0001;  $\text{weight}(i \text{ or } j)$  is the local atom pattern weighting for the selected choice or 1 if not applicable.

The sampled structure with the lowest congestion score is selected as being the best starting point. A localized refinement is performed by iteratively testing each of the atom and pair constraints to determine whether a single change can make an overall improvement to the congestion function. The loop terminates when a full pass fails to result in a better congestion score, and the present structure is taken to be the best solution to the constraints.

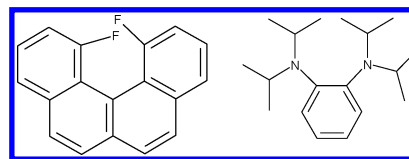
**6. Postprocessing.** A further sequence of steps is typically required before the layout is complete.

*Throwback.* The method described above leads to a set of constraints which includes mainly choices which are locally ideal. For some molecules, there is no combination of choices which would result in an uncongested structure. During the latter stages, each connected component of the molecular graph is analyzed for occurrences of very close contacts between atoms, or bonds other than those of rings which overlap. For each case where this occurs, the shortest path is determined between the atoms affected, and the atoms occurring along the path are recorded.

The atoms which occur in any such conflict are resubmitted recursively to the depiction algorithm, with a special parameter which causes the degrees of freedom about these "throwback atoms" to be increased. All throwback atoms are expressed as single-atom constraints, and their sets of possible internal coordinates are combinatorially expanded to include variations where connections to other throwback

atoms (except when both occur in the same ring) include deviations of  $15^\circ$  and  $25^\circ$ .

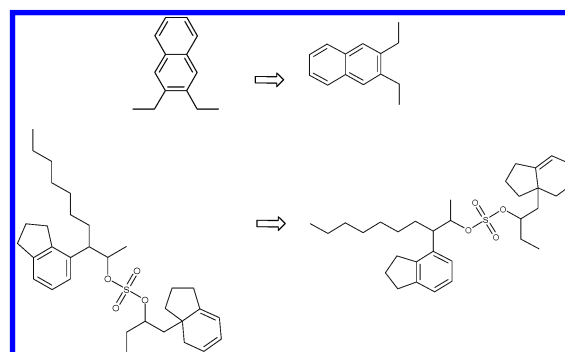
The additional degrees of freedom generally allow an acceptable solution to be found by sampling, for example,



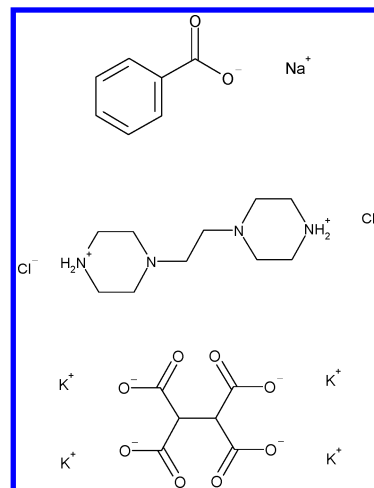
Both of the structures shown above have no viable solution involving  $120^\circ$  angles and uniform bond lengths. By allowing more flexibility, reasonable solutions can be found.

*Stretch, Rotate.* If increasing the degrees of freedom still results in overlapping atoms or crossed chain bonds, the conflicting atoms and bonds are identified. The paths between them are subjected to a correction procedure which involves stretching and rotating bonds until the overlap has been removed.

*Fragments.* Each connected component is rotated, first, in order to maximize its horizontal extent and, second, to maximize the number of bonds lined up to  $15^\circ$  absolute increments.



Structures consisting of multiple connected components must be placed relative to one another. Each component is categorized as either polyatomic or monatomic. Polyatomic components are placed largest first, with the remainder taking up space below. Monatomic components are placed near a complementary atom, ideally, an atom which has already been placed and has opposite charge.



*Hydrogen Placement.* Hydrogen atoms are considered as an afterthought, because they are not displayed by most



output methods. If the input structure features explicit hydrogens, they are placed in order to fill up the remaining interstices between heavy atoms.

**7. Implementation.** All algorithms described have been implemented using the scientific vector language (SVL), which is the underlying engine of the Molecular Operating Environment (MOE). The algorithms specific to structure depiction occupy ca. 3000 lines of code, while general-purpose algorithms for graph manipulation, linear algebra, and nonlinear optimization are documented intrinsic features of MOE.<sup>12</sup>

## RESULTS AND DISCUSSION

**Performance.** A database of 20 000 structures which were published in the context of medicinal chemistry were subjected to the depiction layout algorithm. Within approximately one standard deviation, the molecules contained between 20 and 40 heavy atoms. On an ordinary personal computer (Intel Xeon, 3.0 GHz, Red Hat Linux, single-threaded execution), an average throughput of 38 molecules per second was achieved. The speed is sufficient for interactive real-time use and is also applicable to the processing of moderately large databases.

**Comparison.** To illustrate cases where the current method overcomes particular difficulties, a collection of structures was assembled. The structures are shown in Figure 7, arranged and numbered according to the primary obstacle. The de novo 2D structure depiction capabilities offered by ChemDraw,<sup>13</sup> Daylight,<sup>3</sup> Ogham,<sup>14</sup> and CACTVS<sup>15</sup> are shown alongside. The input data were provided as SMILES strings, which provide no predefined coordinates which might be used as hints.

**1. Detection of an Ideal Solution.** Most organic molecules can be depicted well by combining locally ideal environments in an arbitrary way. Sometimes, however, an overlap of atoms or bonds would occur. The challenge is to find a combination that does not lead to global congestion. For some molecules, the number of ideal solutions is small and the nonideal solutions numerous. The present method finds an ideal solution for each of the structures **1a–c**. When no ideal solution is found, most algorithms will resort to fallback methods in order to salvage the result, such as is seen for the ChemDraw output for **1c** and the Daylight and Cactvs outputs for **1b** and **1c**. The Ogham output for **1c** does not fix overlapping atoms, which results in a depiction which is unusable.

**2. Absence of an Ideal Solution.** For many organic molecules with limited degrees of freedom, no combination of locally ideal placement decisions exists which combine to form an ideal result, and so, the duty of the depiction algorithm is to find a solution incorporating an inoffensive selection of compromises that produces a structure that is adequately legible and aesthetic. Structures without ideal solutions are shown (**2a–e**), and it can be seen that the variety of methods available produce mixed results. Our method combines the relaxing of constraints with an effective sampling/refinement algorithm, which frequently finds a result which minimizes the necessary compromises. The output from Cactvs tends to suggest that fixes are made within a local scope, whereas Daylight largely abandons aesthetic considerations altogether. Ogham seems to ignore

the deleterious effect of overlapping atoms in many cases, whereas the ChemDraw output finds an agreeable compromise in each case.

**3. Chain Blocks.** Seemingly trivial molecules such as perhalogenated alkanes present a global layout problem if placement decisions are made locally and irreversibly. Just one poor decision regarding placement of cis or trans throws the entire layout into disarray. Specific logic for such chains or nonlocal optimization is required in order to obtain the preferred result.

**4. Double-Bond Stereochemistry.** Restricted rotation stereochemistry must be incorporated into the layout algorithm, preferably at an early stage, because the local choice of cis or trans may have globally significant implications. In example **4**, the output from Daylight appears to have corrected the stereochemistry retroactively and, in doing so, produced an overlapped structure which does not resemble the molecule being described.

**5. Congested Small Rings.** The selection of templates is important for many small bridged structures because a specific style is expected for them. The examples **5a–c** are not sufficiently interconnected to warrant embedding as 3D structures, yet they are difficult to draw algorithmically. It is prudent to ensure a ring block embedding which allows a reasonable placement of substituents, which is exacerbated by the high crowding and irregular angles.

**6. Spirocenters.** Atom-fused ring blocks cause difficulties for some depiction algorithms. Although the present method includes some spirocenters in its ring template database, the planar embedding method described has no difficulty with these cases.

**7. Macrocycles.** Strategies for handling macrocyclic ring systems appear to have received inadequate consideration in previous efforts. Other algorithms are evidently restricted largely to simple template matching or embedding as regular polygons, which increasingly resemble circles as the ring becomes larger (e.g., output from ChemDraw for **7a–f**). Even matching to a template is not sufficient, as the output of Ogham and Cactvs for **7b** does not align the heteroatoms in the conventionally accepted way. Placement of bond-fused side rings is seldom pleasing (e.g., **7c** and **7d**), and deeply embedded small rings generally wreak havoc (e.g., **7e**). Large macrocycles with many substituents and, hence, high degrees of freedom cause solution-finding difficulties that are similar to chainlike systems (e.g., **7f**). The output of ChemDraw for **7a** also demonstrates incorrect stereochemistry of the double bond emerging from the ring.

**8. Ring Template Matches.** Template matching for stylized nonplanar ring blocks is significantly more successful if the algorithm removes side rings in advance, as is carried out by our algorithm. The output for **8a** suggests that ChemDraw and Cactvs combine a peeling technique with their ring block embedding methods.

**9. Planar Embedding.** Fallback methods for ring blocks are important when simple trigonometry and templates do not apply. The 2D embedding method used in the current work produces consistently good results, whereas example **9** suggests that methods employed by other algorithms are typically less effective.


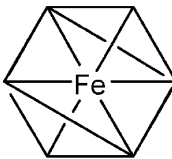
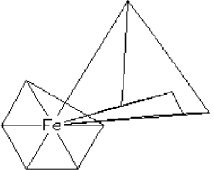
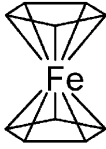
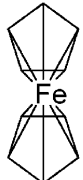
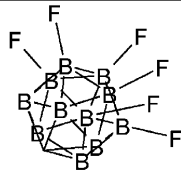
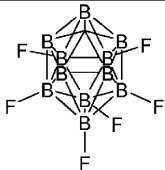
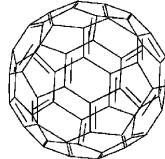
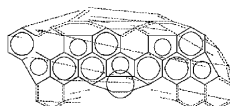
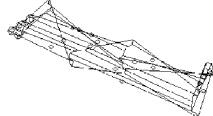
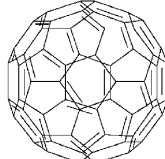
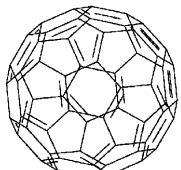
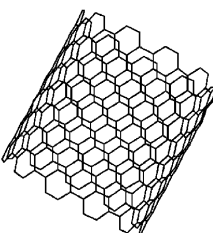
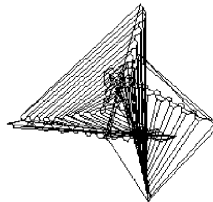
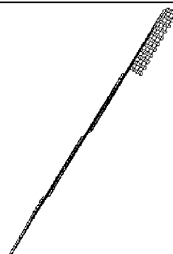
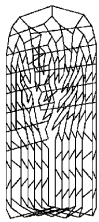
**10. 3D Ring Systems.** Highly interconnected ring blocks can seldom be embedded well by planar methods, and for cases not common enough to warrant a template, a method

	Current work	ChemDraw	Daylight	Ogham	Cactvs
1a					
1b					
1c					
2a					
2b					
2c					
2d					
2e					

	Current work	ChemDraw	Daylight	Ogham	Cactvs
3					
4					
5a					
5b					
5c					
5d					
6					
7a					

	Current work	ChemDraw	Daylight	Ogham	Cactvs
7b					
7c					
7d					
7e					
7f					
8a					
8b					
9					



	Current work	ChemDraw	Daylight	Ogham	Cactvs
10a					
10b		(no result)	(no result)	(no result)	
10c					
10d		(no result)			

**Figure 7.** Selected examples of depicted structures. Output coordinates from the current work, ChemDraw, and Cactvs were used directly to prepare the diagrams. Output from Daylight and Ogham were obtained in the form of bitmap images. Whenever the connective junctures were clear, the outlines were traced and the resulting data used to prepare diagrams in a consistent style.

which takes into account the 3D properties produces far better results than one which attempts to enforce planarity. In the examples **10a–d**, the other algorithms appear to either match a template, produce an indecipherable mess, or fail entirely.

**Bulk Analysis.** Three databases of molecular structures were assembled in order to collect statistics pertaining to the differences between the original 2D structures and their algorithmically depicted equivalents.

1. *Jubilant.* A collection of 20 000 organic molecules from a medicinal chemistry database compiled from the *Journal of Medicinal Chemistry* was used without modification.<sup>16</sup> The aesthetic drawing style used by the collators is considered to adhere strongly to the ideals of the algorithm described in this paper and is, therefore, a reasonable approximation of our upper bound for aesthetic comparisons.

2. *Maybridge.* A collection of 44 000 lead druglike compounds was obtained from Maybridge<sup>17</sup> and used without modification. The structures already existed in 2D form, but the style is not necessarily in keeping with that which we consider to be ideal.

3. *Flattened.* A collection of 4 400 diverse druglike molecules was assembled. The structures, originally 3D models, were converted into 2D structures by a flattening algorithm,<sup>4</sup> which produces results that are recognizable but seldom of presentation quality. This data set can be considered as a minimum bound for aesthetic comparisons.

Four sets of metrics were collected for each database, each structure of which was prescaled so as to bring the average bond length to unity.

1. *Bonded Distances.* Bond distances between non-hydrogen atoms were determined. Deviations from 1 were collected and frequencies accumulated at intervals.

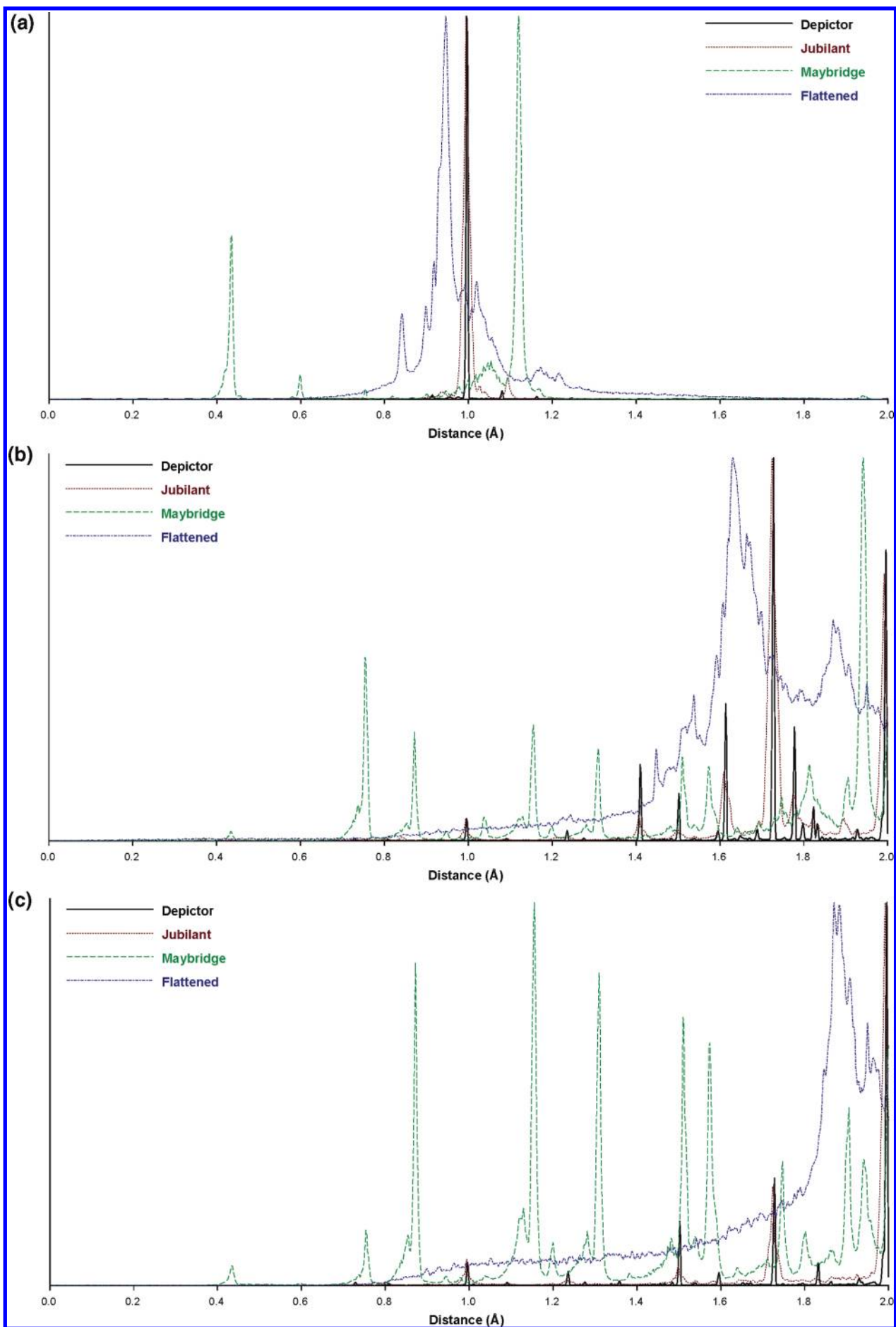
2. *Nonbonded Distances.* Distances between nonbonded atoms were determined and frequencies accumulated at intervals.

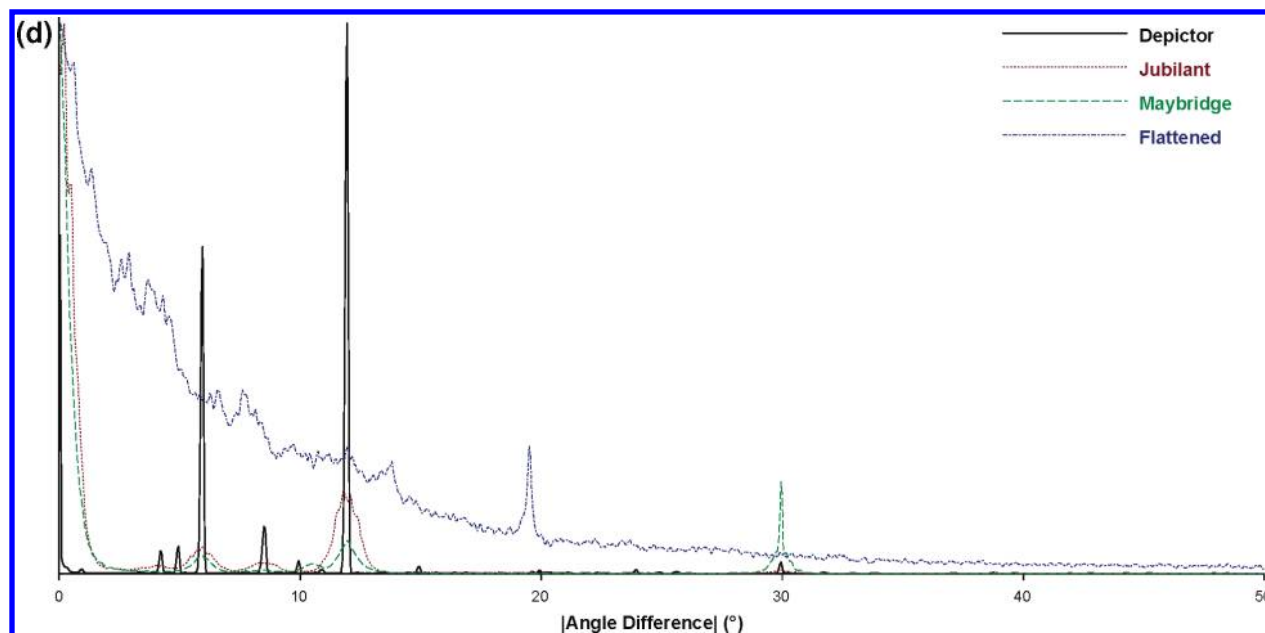
3. *Nonbonded/No Common Neighbor Distances.* An additional category was tabulated in which atom pairs sharing a common neighbor were not included.

4. *Angles.* Atoms of type C, N, O, S, and P with two or three neighbors were examined for their deviation from the nominal chainlike ideal of 120°. Absolute differences from 120° were accumulated at intervals.

The Depictor data set is a catenation of the three databases, in which each structure has been redrawn using our algorithm. In each case, the function value is formulated as the frequency of occurrence divided by the number of structures in the data set, so that the values from the different data sets are directly comparable. The y axes show the log of the relative frequency.

Figure 8a shows the relative distributions of bond distances, normalized to an average of unity. The Depictor





**Figure 8.** (a) Graph of log frequency of occurrence vs normalized distances between bonded atoms. (b) Graph of log frequency of occurrence vs normalized distances between nonbonded atoms. (c) Graph of log frequency of occurrence vs normalized distances between nonbonded atoms with no common neighbor. (d) Graph of log frequency of occurrence vs absolute angular deviation from  $120^\circ$  of bonded atom sequences, which are typically drawn in a zigzag form.

output set shows very little deviation from 1, because consistent bond lengths are a cornerstone of the algorithm. The Jubilant data set also shows a very tight uniformity of bond lengths, whereas the nonideal data sets are more variably distributed.

Parts b and c of Figure 8 show the proportions of nonbonded contacts closer than two bond lengths, where Figure 8c excludes contacts which share a common neighbor. Salient features include the scarcity of contacts closer than one bond length in the Depictor set, the regularity of particular distances, and a high coincidence between the frequency of approach distances of the Jubilant and Depictor data sets.

Figure 8d is an indication of how often atoms which are usually drawn with  $120^\circ$  angles between neighbors are represented in a distortion from this ideal. Particularly prominent peaks arise due to five- and seven-membered rings ( $12^\circ$ ,  $6^\circ$  and  $8.6^\circ$ ,  $4.3^\circ$ ), but otherwise, the deviations are similarly quite rare in both the Depictor and Jubilant sets.

Collectively, these graphs suggest that the Depictor output is highly regular, successful in avoiding undue congestion, and statistically very similar to the nominally ideal Jubilant data set.

**Human vs Machine.** In the tradition of Turing,<sup>18</sup> a test was formulated in order to establish whether it is possible for the depiction algorithm described in this paper to consistently foil attempts by a trained chemist to discern the difference between a human-drawn structure and an algorithmically depicted equivalent.

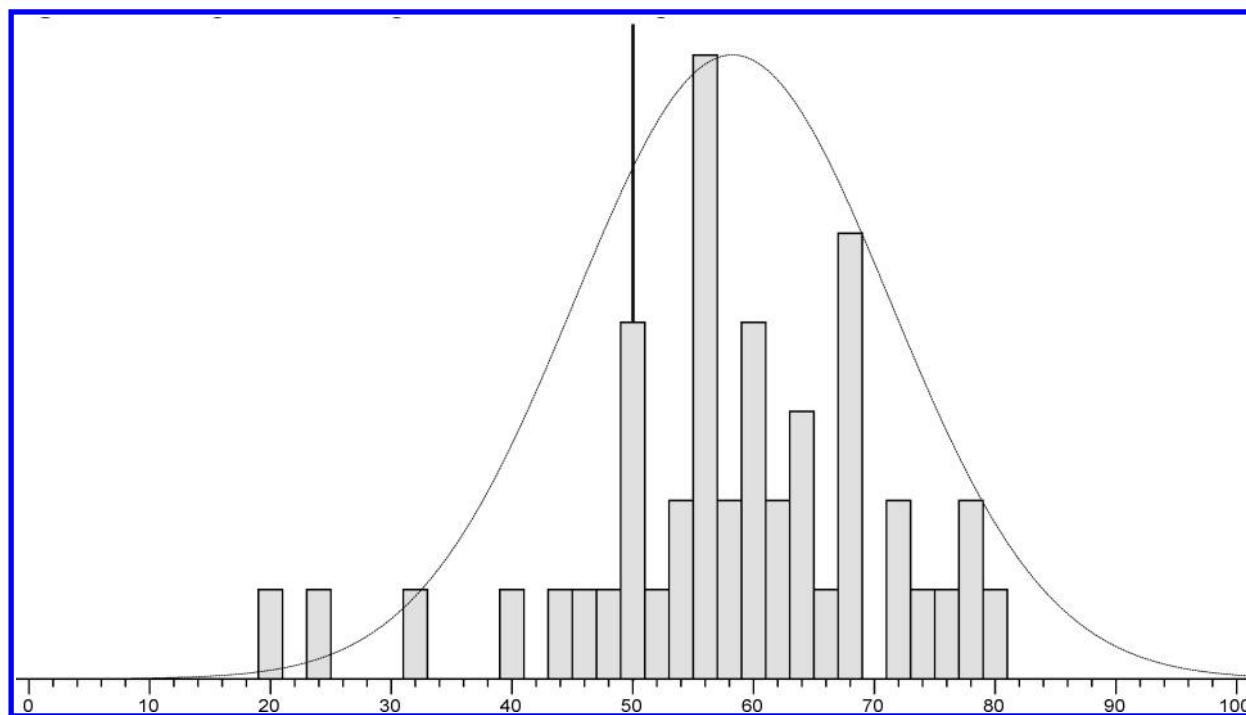
The formulation of the test involved 45 participants, for each of whom 50 compounds were selected randomly from the Jubilant data set. Each of the structures was displayed in two forms, one having been drawn by an expert chemist, the other created de novo by the depiction algorithm, and both were rendered in the same style. The test subject was asked to make an educated guess as to which structure was software-generated.

The ideal outcome would be a 50% rate of correct identification. As can be seen in Figure 9, the distribution has a wide error margin due to the relatively small set sizes and a significant variation of ability to ascertain the correct answer, but the 50% mark is nonetheless within one standard deviation of the mean, which is strong evidence that the results are not only as good as but routinely indistinguishable from the expert drawings. It was noticed that the rotational alignment of fragments was the main way in which medicinal chemists scored unusually highly in this test, because the choice of orientation carries implicit meaning which is beyond the scope of the present method.

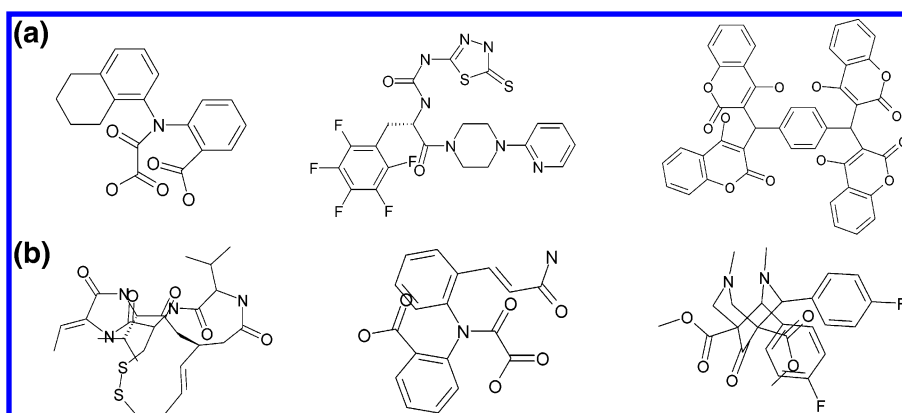
**Failure Detection.** A database of 40 196 structures from the Jubilant data set was subjected to the depiction algorithm, and an effort was made to ascertain what fraction of this realistic sample could be considered unacceptable for presentation purposes.

The first step was the reduction of the data set by applying a systematic algorithm for detecting cases worthy of further investigation. Any structure which did not activate at least one of the following scenarios was assumed to be successful: (1) any bond is 20% longer than the unit bond length; (2) any nonbonded heavy atoms are closer than 50% of the unit bond length; (3) any C, N, or O atoms (with no triple bond) that have neighbors within  $5^\circ$  of linearity; and (4) any two bonds intersect.

Approximately 2000 structures were selected by applying these rules and were examined subsequently. At this point, the judgment of success became subjective. Any structure that was excessively tangled was considered to be a failure, but in many cases, the criteria were raised or lowered depending on the difficulty of finding an adequate layout: if the best human-drawn structure was also visually unpleasant, the depiction was not judged harshly. Minor layout imperfections were not considered grounds for failure, unless they resulted in unnecessary crossed bonds or congestion of more than two atoms. Examples of imperfectly drawn



**Figure 9.** Histogram of Turing test results; 45 samples, mean 58.1%, std. dev. 12.8.



**Figure 10.** (a) Selected examples of marginally successful depiction results. (b) Selected examples of depiction results which are deemed unsuccessful.

structures that were accepted are shown in Figure 10a, and those considered failures are shown in Figure 10b.

Overall, 141 structures were considered to be unsuccessful, which leads to a 99.65% success rate. Failed depictions (0.35%) were, in most cases, readily discernible to the eye but not suitable for high-quality presentations.

**Summary.** The combination of chemical logic and embedding algorithms for determination of the constraints, sampling of the constraints to find the most disperse combination, and followed by polishing and occasional retrofixing has been found to be very successful. The method which we have presented is fast, effective, and general. Its design readily facilitates incremental improvement and, in its present state, has a high success rate for producing aesthetically ideal diagrams for a large subset of organic chemistry.

While structure diagrams for important presentations should still be verified, the algorithm described represents a significant advancement over presently available tools. Unsatisfactory output is sufficiently rare that the depiction of whole databases, or on-demand generation of structure diagrams, can be undertaken with confidence. We anticipate

that the existence of tools of the quality described will alleviate a significantly time-consuming chore, allowing chemists to immediately communicate their results in the structure diagram form which is preferred by the community at large. We also anticipate that the availability of reliable diagram generation tools will improve the productivity of individual computational chemists, who will be able to more quickly assimilate chemical information.

For future work, significant scope exists for improving the final rotational alignment of fragments, which could incorporate commonly preferred heterocycle orientations, the maximizing of symmetry along the axes, and consistency among analogue series. The production of multiple ring block solutions, when more than one stylized planar representation is available, would allow certain structures to be solved more effectively. Templates for substructures other than ring blocks may be considered, as well as improvements to the relatively simplistic 3D ring-embedding/flattening algorithm. Refinement of the sampling algorithm to use guided heuristics, and possibly dynamic modification of the constraint parameters, may be considered.



Most of the development and testing has been done with organic molecules relevant to life sciences, but additional consideration of metals which have more diversity in their valences and bonding patterns, as well as attention to cage structures such as buckyballs, nanotubes, zeolites, and higher-order macrocycles, would extend the applicability of the algorithm.

## REFERENCES AND NOTES

- (1) Weininger, D. SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *J. Chem. Inf. Comput. Sci.* **1988**, 28, 31–36.
- (2) (a) Helson, H. E. Structure Diagram Generation. *Rev. Comput. Chem.* **1999**, 13, 313–398. (b) Weininger, D. Depict. Graphical Depiction of Chemical Structures. *J. Chem. Inf. Comput. Sci.* **1990**, 30, 237–243. (c) Fricker, P. C.; Gastreich, M.; Rarey, M. Automated Drawing of Structural Molecular Formulas under Constraints. *J. Chem. Inf. Comput. Sci.* **2004**, 44, 1065–1078.
- (3) *DEPICT*; Daylight Chemical Information Systems Inc.: Aliso Viejo, CA. Depiction examples herein are those produced by the Web page [http://www.daylight.com/smiles/f\\_smiles.html](http://www.daylight.com/smiles/f_smiles.html) (accessed Feb 23, 2006).
- (4) For comparison purposes, a 2D variation of conventional molecular modeling force fields was used to successively push the initial 3D structure into a reasonably well-spaced planar form.
- (5) Rothberg, E. Implementation of Algorithms for Maximum Matching on Nonbipartite Graphs. Ph.D. Thesis, Stanford University, Stanford, CA, 1973.
- (6) Cormen, T. H.; Leiserson, C. E.; Rivest, R. L. Graph Algorithms. In *Introduction to Algorithms*; MIT Press: Cambridge, MA, 1990; pp 463–629.
- (7) (a) The algorithm used is unpublished work. The “smallest set of smallest rings” can be defined and implemented in a number of ways, of which any robust method is suitable. (b) Berger, F.; Flamm, C.; Gleiss, P. M.; Leydold, J.; Stadler, P. F. Counterexamples in Chemical Ring Perception. *J. Chem. Inf. Comput. Sci.* **2004**, 44, 323–331.
- (8) Each node is initially given the same label. The graph is traversed in order to obtain a new set of labels which reflects the rank and uniqueness according to the graph and present set of labels. By iteratively perturbing the lowest ranking of the new labels and reapplying the traversal, a unique set of ordered labels is eventually obtained.
- (9) (a) In this work, automorphic graphs are generated using a brute force method with a maximum permutation limit. (b) Faulon, J.-L. Isomorphism, Automorphism Partitioning and Canonical Labelling Can Be Solved in Polynomial-Time for Molecular Graphs. *J. Chem. Inf. Comput. Sci.* **1998**, 38, 432–444.
- (10) Crippen, G. M.; Havel, T. F. *Distance Geometry and Molecular Conformation*; John Wiley & Sons: New York, 1988.
- (11) Gill, P.; Murray, W.; Wright, M. H. *Practical Optimization*; Academic Press: New York, 1981.
- (12) The source code for the algorithms described within, and those referred to, is written in SVL and packaged as part of MOE; it may be examined and used under terms of the MOE user license, which is available from the Chemical Computing Group, Inc., 1010 Sherbrooke Street West, Suite 910, Montréal, Québec, Canada. <http://www.chemcomp.com> (accessed Feb 23, 2006).
- (13) *ChemDraw Pro*, version 10; CambridgeSoft Corp.: Cambridge, MA. Depiction examples herein are those produced using the Paste Special SMILES command.
- (14) *Ogham*; OpenEye Scientific Software, Inc.: Santa Fe, NM. Depiction examples herein are those produced by the Web page <http://demo.eyesopen.com/cgi-bin/depict> (accessed Feb 23, 2006).
- (15) (a) 2D coordinates for comparison purposes were obtained using an evaluation copy of the CACTVS toolkit, version 3.313. (b) Up to date versions of the CACTVS toolkit can be obtained at the site <http://www.xemistry.com/academic> (accessed Feb 23, 2006).
- (16) Jubilant Biosys Pvt. Ltd., Columbia, MD.
- (17) Maybridge, Tintagel, Cornwall, England.
- (18) Turing, A. M. Computing Machinery and Intelligence. *Mind* **1950**, 50, 433–460.

CI050550M