Joseph B. Moon and W. Jeffrey Howe
Computational Chemistry, Upjohn Laboratories, Kalamazoo, Michigan 49001

ABSTRACT **The design of molecules to bind specifically to protein receptors has long been a goal of computer-assisted molecular design. Given detailed structural knowledge of the target receptor, it should be possible to construct a model of a potential ligand, by algorithmic connection of small molecular fragments, that will exhibit the desired structural and electrostatic complementarity with the receptor. However, progress in this area of receptor-based, de novo ligand design has been hampered by the complexity of the construction process, in which potentially huge numbers of structures must be considered. By limiting the scope of the structure-space examined to one particular class of ligands—namely, peptides and peptide-like compounds—the problem complexity has been reduced to the point that successful, de novo design is now possible. The methodology presented employs a large template set of amino acid conformations which are iteratively pieced together in a model of the target receptor. Each stage of ligand growth is evaluated according to a molecular mechanics-based energy function, which considers van der Waals and coulombic interactions, internal strain energy of the lengthening ligand, and desolvation of both ligand and receptor. The search space is managed by use of a data tree which is kept under control by pruning according to the energy evaluation. Ligands grown by this procedure are subjected to follow-up evaluation in which an approximate binding enthalpy is determined. This methodology has proven useful as a precise model-builder and has also shown the ability to design bioactive ligands.**

INTRODUCTION

The ability of a molecule, such as a drug, to exert a desired biological effect is often related to its affinity for one or more endogeneous receptor molecules. For a ligand to interact optimally with a receptor, it must be able to attain a shape which is at least partly complementary to that of a binding location on the receptor. Additionally, other factors such as electrostatic interactions, hydrogen bonding, hydrophobic interactions, desolvation effects, and cooperative motions of ligand and receptor all influence the binding event and should be taken into account in attempts to design bioactive ligands. Processes such as distribution and metabolism, while they play a critical role in the delivery of the putative ligand to the receptor location, do not reflect a compound's "intrinsic activity" and lie outside the scope of the current discussion.

In principle, it should be possible to design molecules that will bind to a preselected site on a receptor. This is not a simple undertaking, since in most design situations little or no structural information exists to characterize the receptor. One can, however, use "indirect" methods[1] to exploit what *is* known about molecules that elicit the desired biological response (assuming that they interact with the same receptor) to generate a structural and electronic hypothesis of what the receptor recognizes or will accept. Various computer-based methods have been developed to assist in this kind of study.[1–8] Once the hypothesis has been generated it can be used to suggest molecular modifications to improve the activity of known ligands or to identify entirely new structural classes (lead compounds) for study as potential ligands. The latter can be accomplished via searches over large databases of 3D molecular structures to identify molecules which match the hypothesized requirements for activity.[9–16]

The increasing availability of biomacromolecule structures that have been solved crystallographically has prompted the development of "direct" computational methods for molecular design, in which the steric and electronic properties of receptor binding sites are used to guide the design of potential ligands.[1,11,12,17–19] Direct methods generally fall into two categories: (1) design by analogy, in which 3D structures of known molecules (such as from a crystallographic database) are placed in the receptor structure and scored for goodness-of-fit; and (2) de novo design, in which the ligand model is constructed piecewise in the receptor. The latter approach, in particular, offers considerable promise for the development of novel molecules, uniquely designed to bind to the target.

Received August 23, 1990; revision accepted March 15, 1991.
Address reprint requests to either author, The Upjohn Company, Computational Chemistry, 301 Henrietta St., Kalamazoo, MI 49001.

© 1991 WILEY-LISS, INC.

Breckenridge Exhibit 10
Breckenridge v. Novarti
DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

While examples of successful, computer-assisted, de novo design can be found,[20] there are no examples of automated, or computer-driven, de novo construction in the literature (although Wise et al.[21] have reported using the structure-building program GENOA[22] to generate molecules to match a requirements hypothesis). The term "automated de novo design" is used here to refer to the algorithmic construction of a putative ligand from small fragments, guided by steric and electronic constraints imposed by the receptor, plus appropriate consideration of solvation effects and internal strain energy of the ligand.

In a recent series of papers,[23–26] Dean and coworkers describe a four-step strategy for automated, de novo drug design. Although their goal has not yet been achieved, there has been considerable progress in algorithm development. Furthermore, their studies make clear the complexity of the de novo construction problem as well as the importance of developing noncombinatorial approaches. In our work, we have chosen to focus on one particular region of the large structure-space that is ultimately the design territory of such methods. By confining the search space to consider only amino acids and related fragments as the molecular building blocks, the construction problem has become quite tractable, and we are able to report the first examples of bioactive ligands designed by automated de novo methods. The putative ligands that result from this construction method are peptides and peptide-like compounds rather than the small organic molecules that are typically the goal of drug design research. The appeal of the peptide building approach is not that peptides are preferable to organics as potential pharmaceutical agents, but rather that: (1) they can be generated relatively rapidly de novo; (2) their energetics can be studied by well-parameterized force field methods; (3) they are much easier to synthesize than are most organics; and (4) they can be used in a variety of ways, for peptidomimetic inhibitor design, protein–protein binding studies, and even as shape templates in the more commonly used 3D organic database search approach described above. We also show that the method need not be restricted to just the 20 natural amino acids; it can easily be extended to include other related fragments of interest to the medicinal chemist.

METHODS

Description of the GROW Method

Overview

The de novo peptide design method has been incorporated in a software package called GROW. In a typical design session, standard interactive graphical modeling methods (using the Mosaic software system,[27] which is based on MacroModel[28]) are employed to define the structural environment in which GROW is to operate. The environment could be the active site cleft of an enzyme, or it could be a set of features on a protein surface to which the user wishes to bind a peptide-like molecule. The GROW program then operates independently of the user to generate a set of potential ligand molecules. Interactive modeling methods then come into play again, for examination of the resulting molecules, and for selection of one or more of them for further refinement.

The method is designed to construct peptide models from a user-selected starting position by iteratively piecing together amino acids in conformations which will interact most favorably with the atoms in the receptor site. For input, GROW operates on an atomic coordinate file generated by the user in the interactive modeling session, plus a small fragment (an acetyl group) positioned in the receptor to provide a starting point for peptide growth. These are referred to as "site" atoms and "seed" atoms, respectively. A second file provided by the user contains a number of control parameters to guide the peptide growth.

The operation of the GROW algorithm is conceptually fairly simple, and is summarized in Figure 1. GROW proceeds in an iterative fashion, to systematically attach to the seed fragment each amino acid template in a large preconstructed library of amino acid conformations. When a template has been attached, it is scored for goodness-of-fit to the receptor site, and then the next template in the library is attached to the seed. After all the templates have been tested, only the highest scoring ones are retained for the next level of growth. This procedure is repeated for the second growth level; each library template is attached in turn to each of the bonded seed/amino acid molecules that were retained from the first step, and is then scored. Again, only the best of the bonded seed/dipeptide molecules that result are retained for the third level of growth. The growth of peptides can proceed in the N-to-C direction only, the reverse direction only, or in alternating directions, depending on the initial control specifications supplied by the user. Successive growth levels therefore generate peptides that are lengthened by one residue. The procedure terminates when the user-defined peptide length has been reached, at which point the user can select from the constructed peptides those to be studied further. The resulting data provided by the GROW procedure include not only residue sequences and scores, but also atomic coordinates of the peptides, related directly to the coordinate system of the receptor site atoms. In the following sections we examine in more detail the individual components that comprise the basic procedure just described.

Library construction

Because most amino acids are quite flexible, a large number of template structures must be tested during the growth procedure to ensure adequate

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

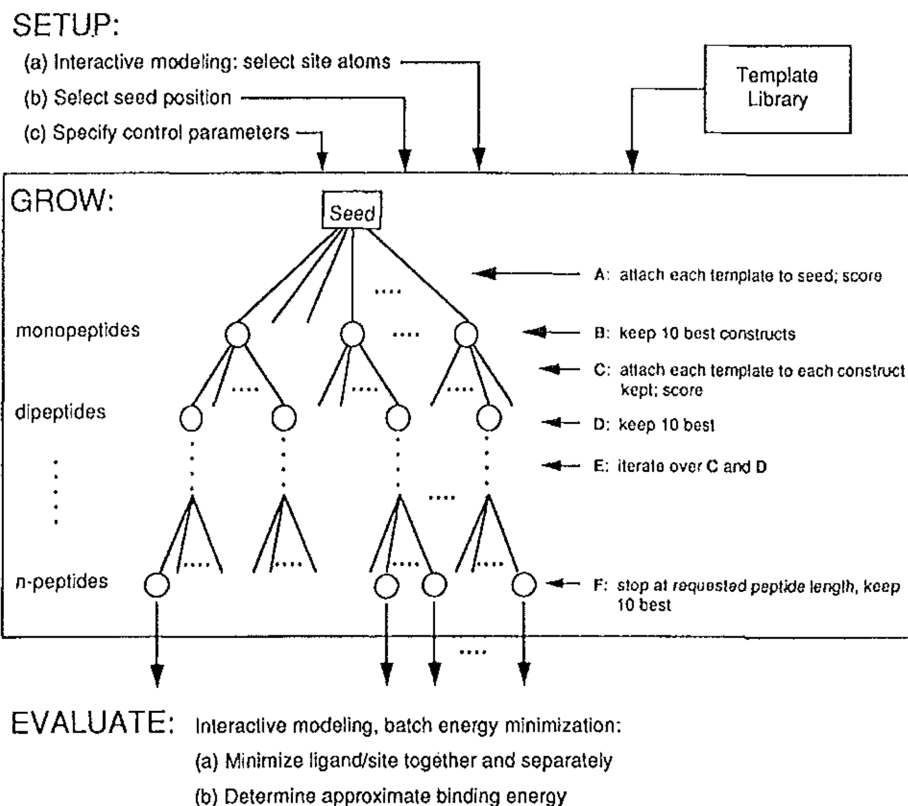


Fig. 1. Schematic overview of the operation of the GROW algorithm. The site-and seed coordinate file and the command file (described later) are provided to the GROW procedure by the user. Growth can be visualized as a tree process in which each library template is attached to the seed (**A**) and then evaluated by the scoring function. Of the resulting 6000+ constructs, the 10 best are kept for the next level (**B**). 10 is the default retention; a command file keyword can be used to broaden the search at any stage. To each retained monopeptide/seed construct are attached all library templates, which are again scored (**C**). After pruning (**D**), the process is repeated (**E**) until the specified peptide length (specified in the command file, see Fig. 5) is reached (**F**). In this tree diagram, circles represent those nodes selected (based on highest scores across the entire level) for further growth. Uncircled nodes are pruned. Horizontal dots denote continuation across all template additions, and vertical dots represent the iterative process of tree growth.

coverage of the conformational space accessible to each residue. The template library was generated with the Mosaic modeling program in conjunction with the MacroModel/BatchMin[28] (version 2.5) implementation of the AMBER[29] forcefield. The same forcefield implementation was used for all energy-related work described herein. Starting models of the 20 standard amino acids were constructed as *N*-acetyl-*N*′-methylamides (Fig. 2A), followed by energy minimization.* The models were then subjected to a search procedure in which conformers were generated by varying all flexible torsion angles in the amino acids by random increments. Any conformer which contained two nonbonded heavy atoms at a separation of <2.0 Å was discarded. After 3,000 to 5,000 viable conformations were produced for each amino acid, the structures were subjected to a *partial* energy minimization (15 iterations of block diagonal Newton–Raphson minimization) to relieve significant internal strain energies. At this point, each conformation was compared to every other conformation so that duplicate structures would be discarded. Two conformations were considered to be identical if no atomic positions differed by more than 0.3 Å when the structures were aligned by superpositioning of their N-terminal amide atoms. The remaining conformations were sorted in ascending energy order and were stored in the template library along with their energies. Templates of nonstandard amino acids, pseudodipeptides, and organic terminal groups were constructed in the same manner, em-

*Unless otherwise indicated, the convergence criterion used for all energy minimizations discussed in this paper was an rms gradient of <0.1 kcal/Å, with the BatchMin/MacroModel PRCG minimizer.

Find authenticated court documents without watermarks at docketalarm.com.

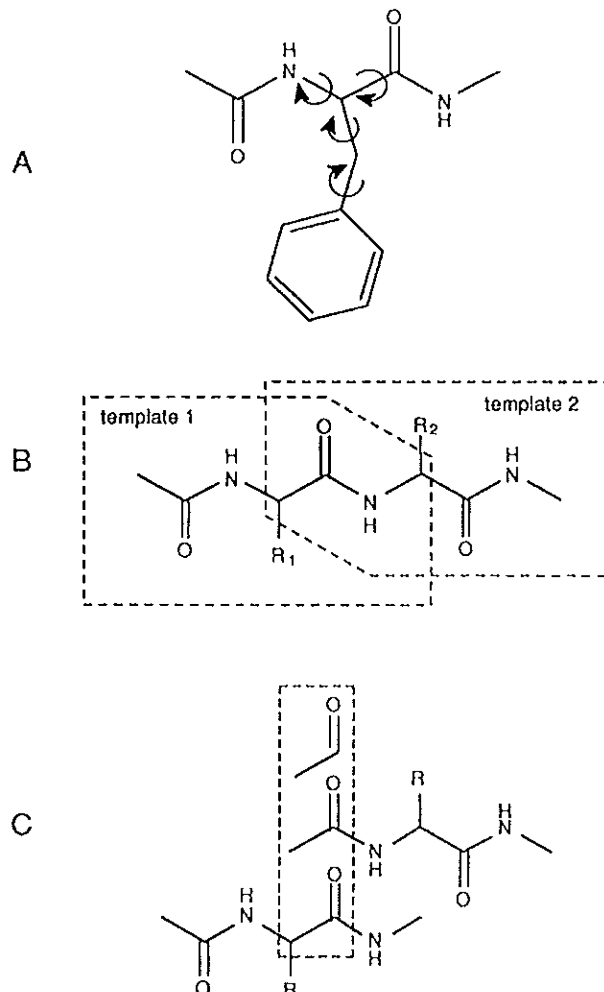


Fig. 2. (A) Template generation method using phenylalanine as an example: bonds marked with arrows are rotated by random increments to generate additional conformations. This is followed by contact filtering, partial minimization, and duplicate elimination. (B) Template connection method: amide end groups are superimposed to connect two templates together in the proper geometries to form peptides. (C) Template alignment method: the two alignments of a template with the seed group are shown. The alignment used depends on the direction in which the peptide is to be grown.

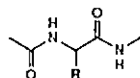
ploying the extended parameter set (in addition to the original[29] AMBER parameters) provided by the MacroModel/BatchMin implementation.

Figure 3 lists the contents of the template library and the number of unique conformations stored for each residue. During a GROW run, from 300 to 1,000 lowest energy conformations are typically utilized for each amino acid; the default is 300. For comparison, values in parentheses indicate the number of *initial* conformations generated for the residues during library construction. Of the 2,000 trial conformations of alanine, for example, partial energy minimization and duplicate elimination reduced the set to 171 unique conformations. As might be expected, this type of reduction in the number of conformations was not seen with the pseudodipeptides and certain of the other residues, due to their extreme flexibility. The implications of template flexibility will be discussed in a later section.

Application of a *partial* energy minimization during library construction produces structures that lie near, but not generally at, energetic minima. Since energetic minima of a bound ligand will not necessarily correspond to minima of an unbound ligand, restriction of templates to unbound minimum-energy conformations represents an unwarranted

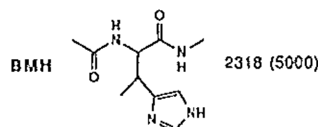
DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

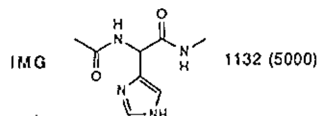
Standard Amino Acids

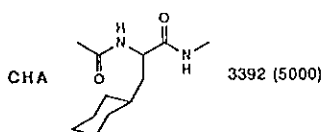


ALA	171 (2000)	LEU	1108 (5000)
ARG	4987 (5000)	LYS	4743 (5000)
ASN	2706 (5000)	MET	4661 (5000)
ASP	1505 (5000)	PHE	3485 (5000)
CYS	2123 (3000)	PRO	53 (2000)
GLN	3734 (5000)	SER	1598 (5000)
GLU	3213 (5000)	THR	1702 (5000)
GLY	271 (1000)	TRP	4537 (5000)
HIS	4026 (5000)	TYR	4732 (5000)
ILE	1478 (5000)	VAL	346 (5000)

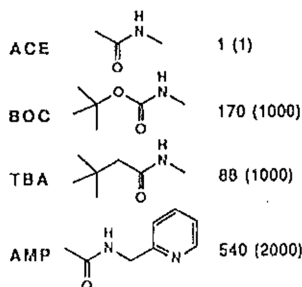
Non-standard Amino Acids







Terminal Groups



Pseudodipeptides

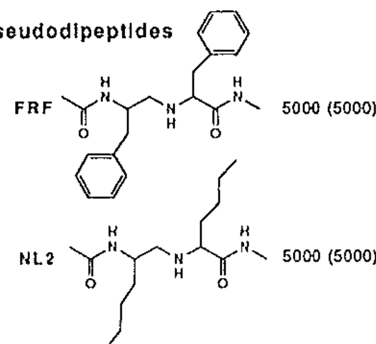


Fig. 3. Contents of the template library. At present, the template library contains standard L- and D-amino acids, several nonstandard residues, organic terminators, and pseudodipeptides, some of which are shown here. The table indicates, for each fragment, its 3-character identifier, which can be specified in the control file for running GROW in restricted mode, a parenthesized value which indicates the number of initial conformations generated for that fragment during library construction, and an unparenthesized value which indicates the number of conformations that survived the partial minimization and duplicate elimination steps during library construction. Data shown for standard amino acids apply equally for L- and D-forms.

constraint. The collection of amino acid templates that resulted from the procedure just outlined represents a broad sampling over low-energy conformational space. The assumption made is that such fragments can be connected together to form peptides with low internal conformational energy; adverse interactions *between* residues are dealt with at a later stage.

The acetyl and amide end groups placed on the amino acid models serve two purposes. First, they produce some of the conformational restriction experienced by individual amino acids when they are connected in a polypeptide chain. They also provide a convenient way to connect the templates during peptide construction; two templates can be joined together simply by superimposing the N-terminal amide of one template onto the C-terminal amide of another (Fig. 2B).

Seed fragment positioning

The placement of the seed fragment, while separate from the GROW method itself, has a great influence on the outcome of a GROW procedure. A poorly positioned seed can prevent designed peptides from reaching important interaction sites in the receptor. Because of this sensitivity, we have examined a number of techniques for choosing reasonable seed positions. In the few cases in which an X-ray crystallographic structure of a bound ligand is available, atoms within the ligand can be used to form a

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE

fastcase® Smarter legal research.