

Drug Guru: A computer software program for drug design using medicinal chemistry rules

Kent D. Stewart,^{a,*} Melisa Shiroda^a and Craig A. James^b

^aAbbott Laboratories, Global Pharmaceuticals Research and Development, Abbott Park, IL 60064, USA

^bMoonview Consulting, LLC, San Diego, CA, USA

Received 27 April 2006; revised 6 June 2006; accepted 8 June 2006

Available online 25 July 2006

Abstract—Drug Guru™ (drug generation using rules) is a new web-based computer software program for medicinal chemists that applies a set of transformations, that is, rules, to an input structure. The transformations correspond to medicinal chemistry design rules-of-thumb taken from the historical lore of drug discovery programs. The output of the program is a list of target analogs that can be evaluated for possible future synthesis. A discussion of the features of the program is followed by an example of the software applied to sildenafil (Viagra®) in generating ideas for target analogs for phosphodiesterase inhibition. Comparison with other computer-assisted drug design software is given.

© 2006 Elsevier Ltd. All rights reserved.

1. Introduction

A rich tradition of analog design strategies has evolved for creating new compounds within medicinal chemistry research for biological evaluation. When similar physical properties between lead compound and analog are desired, ‘bioisosteric’ replacements are commonly employed. Where more structurally altered yet still compatible differences between lead compound and analog are desired, non-classical replacements are considered. This latter situation occurs when a chemist desires structures that are outside of the intellectual property of a competitor or when attempting to achieve more dramatic changes in potency or bioavailability. Collectively, these replacements are known to experienced medicinal chemists as ‘rules-of-thumb’ for drug design. Two examples of well-known design rules-of-thumb, the carboxylate-to-tetrazole and amide-to-retroamide replacements, illustrated in Figure 1, have historically been considered to yield analogs of high interest in medicinal chemistry programs. Hundreds of these structural transformations have been reported in the literature and have potential for general applicability and acceptance as design rules.^{1–9} While no single rule is ever guaranteed to achieve the desired endpoint, the traditional rules repre-

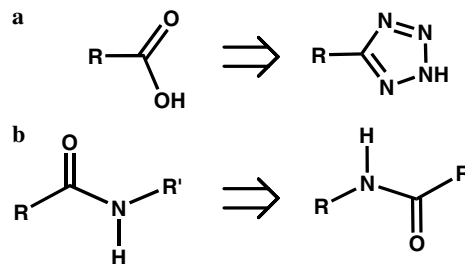


Figure 1. Two well-known examples of rule-of-thumb for designing analogs: (a) the carboxylate-to-tetrazole replacement, (b) the amide-to-retroamide switch. Other examples are listed in Table 1 or described in the text.

sent a useful starting place in a drug discovery effort, particularly when other knowledge such as pharmacophore models, 3D receptor structure, or structure–activity relationship data is limited, low quality, or non-existent.

We have written a web-based computer application, called ‘Drug Guru,’¹⁰ that contains the historical rules-of-thumb as lines of SMIRKS code.¹¹ The name of the program is an acronym for drug generation using rules. The program applies a library of rules to any input structure and then permits visual or computational evaluation of the output structures. To our knowledge, no previously described or commercially available software

Keywords: Computational chemistry; Drug design.

*Corresponding author. Tel.: +1 847 937 1205; fax: +1 847 937 2625; e-mail: kent.d.stewart@abbott.com

accomplishes all of the tasks of Drug Guru, and we propose that it nicely complements other computational approaches. In this article, we describe the basic features of the software, illustrate its use in a retrospective study of sildenafil, and end with a comparison to other computer-aided drug design software.

2. Materials and methods

2.1. Drug Guru rules

The most important and novel aspect of Drug Guru is the collection of rules. We found the SMIRKS transformation protocol useful for encoding rules.¹¹ SMIRKS is a linear text string that represents a graph transformation which, when applied, converts a representation of an input structure (a SMILES code) into a new structure (a new SMILES code). Ten illustrative rules and their corresponding SMIRKS codes are given in Table 1.

In practice, two general kinds of SMIRKS are required to cover the transformations encountered in most medicinal chemistry programs: functional group transformations and molecular framework modifications. In the version of Drug Guru described here, there are 133 and 53 rules in each of these general classes, respectively, for a total of 186 rules. A list of rule categories is given in Table 2. Fourteen functional groups were empirically selected as most frequently encountered in medicinal chemistry research programs. Rules corresponding to entries 1–7 of Table 1 are typical functional group transformations and will be familiar to most experienced chemists. A text mnemonic is assigned to each rule to permit ready comprehension of the general nature of the rule (Table 1, column 1). An extensive literature survey of medicinal chemistry reports is currently underway to produce a more expanded and comprehensive listing of rules and their corresponding SMIRKS. In addition to the functional group transformation rules, a variety of molecular framework modification rules are also encoded, including ring break/form, ring contraction/expansion, and ring replacement rules (Table 2). Rules for entries 8–10 in Table 1 are representative framework modifications. Homologation rules, such as transformations that extend ring substituents by an oxygen, sulfur, carbon, or nitrogen atoms (such as entry 8, Table 1), are examples of rules that currently fall into a ‘miscellaneous’ category (Table 2). Also listed in Table 2 within the category of framework modifications are rules for altering molecular conformation: for example, the addition of geminal methyl groups to flexible chains to exploit the Thorpe–Ingold effect.¹² Two framework modification rules from Table 1, the ring-break rule, entry 9; and the NC-switch rule, entry 10, illustrate more complex and less obvious structural transformations and are described in detail below. A public web-based utility program is available (www.daylight.com/cgi-bin/contrib/react.cgi) that will allow readers to evaluate the SMIRKS supplied in Table 1. Drug Guru is written in Perl programming language, and uses the DayPerl2 interface to the Daylight SMILES, SMARTS, and Reaction Toolkits.

The ring-break transformation (rule #9 in Table 1) finds its basis in the general strategy to take atoms covalently connected within a ring and replace with atoms that are intramolecularly H-bonded to form a ‘virtual’ ring system. Drug Guru has several of these ring-break rules. In the example described here, one of the aryl rings of a fused aromatic ring is replaced with a carbonyl and amine that project from the other aryl ring. This ring-break rule was recently illustrated by Novartis in their scaffold morphing discovery of the anthranilic amide inhibitors of KDR kinase.¹³ This rule and example are illustrated in Figure 2.

The NC-switch transformation, rule #10 in Table 1, finds its basis within the general ‘carbon-to-nitrogen’ replacement strategy used in discovery of new heterocycles as core pieces in drugs. The general notion is to take every carbon in a structure and sequentially replace each with nitrogen. In the specific case of rule #10, two atoms are ‘exchanged’ in the situation where a vinyl amine is part of the aromatic system, leading to an ‘NC-switch’ rule. A good historical illustration of an application of this NC-switch rule may be found in the research published by Abbott Laboratories that led to the pyridone class of antibacterial agents typified by ABT-719.²⁷ This rule and example are illustrated in Figure 3.

2.2. Structure input

Input of a chemical structure can be via corporate identifier code, drawing program, or by coordinate file, either individually or in batch. Typical input structures include the current lead structure for a particular project, a competitor compound or a natural ligand. After entering the input structure, the user selects the run button with no additional user information needed. All general transformations are applied in this mode. An ‘expert user’ page is optionally accessed to allow some additional features (described below) to be explored if desired. The input web page is illustrated in Figure 4. Tautomerism of the input structure was found to be an important factor: different tautomers give different output results. When tautomeric possibilities are found by an automatic tautomer check within Drug Guru, the user is queried to select the desired input tautomer(s).

2.3. Evaluation of output

In Drug Guru, the primary method of evaluating the output structures is by visual inspection. Currently, we have implemented 186 rules and empirically observe that a typical medicinal chemistry request will result in 50–150 output structures. Structures with higher structural complexity, that is, high number of functional groups or skeletal connections, yield more instances of rule applicability, thus leading to a greater number of output structures. Drug Guru requires only seconds for generation of results. The visual evaluation and thoughtful analysis of the results by the end-user medicinal chemists typically requires 10–30 min. With the number of output structures in the hundreds, we desired a database management tool that would conveniently permit casual

Table 1. Illustrative listing of rules

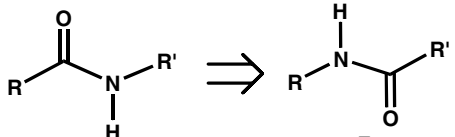
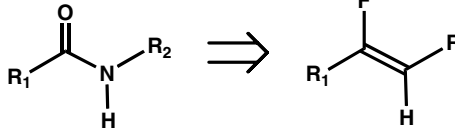

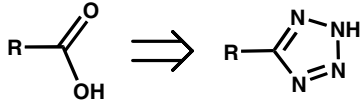
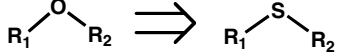
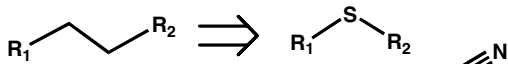
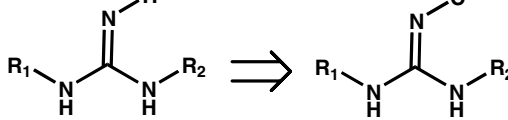

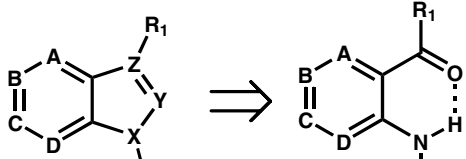
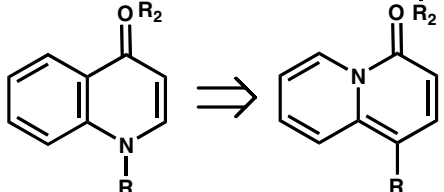
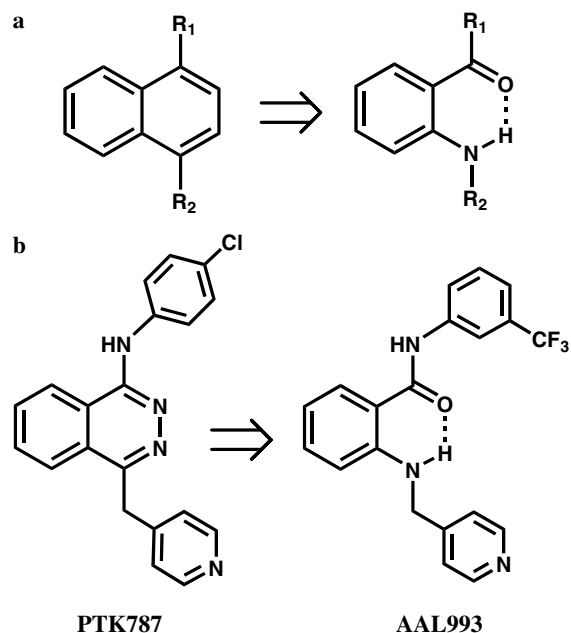
Rule name	Rule illustration	SMIRKS
1 Amide-to-retroamide		<chem>[*!#1:1]-[NH:2]-[C:3](=[O:4])-[C,c:5] >> [*!#1:1]-[C:3](=[O:4])-[NH:2]-[C,c:5]</chem>
2 Amide-to-fluoroolefin		<chem>[*!#1:1]-[NH]-[C:3](=[O])-[C,c:5] >> [*!#1:1]-[CH]=[C:3](-[F])-[C,c:5]</chem>
3 Hydroxy-to-methoxy		<chem>[C,c:1]-[OH] >> [C,c:1]-[O]-C</chem>
4 Carboxylate-to-tetrazole		<chem>[C,c:1]-[C:2](=[O])O[H] >> [C:1]-[C:2]1[n]=[n]-[nH]-[n]=1</chem>
5 Ether-to-thioether		<chem>[*!O:5][C,c:1]([*!O:6])[O][C,c:2]([*!O:7])[*!O:8] >> [*!O:5][C,c:1]([*!O:6])-[S]-[C,c:2]([*!O:7])[*!O:8]</chem>
6 Ethano-to-S		<chem>[C,c:1]-[CH2]-[CH2]-[C,c:2] >> [C,c:1]-[S]-[C,c:2]</chem>
7 Guanidine-to-cyanoguanidine		<chem>[N:2]-[C:3]([N:5])=N([H]) >> [N:2]-[C:3]([N:5])=NC#N</chem>
8 Homologation-C		<chem>[*;R:1]-[C,N,O,S,Cl,Br,F:2] >> [*;R:1]-[CH2]-[C,N,O,S,Cl,Br,F:2]</chem>
9 Ring-break		<chem>[n]([*:20])1:[a:2]2:[c,n:3]:[c,n:4]:[c,n:5]:[c,n:6]:[c,n:7]2:[c]([*:9]):[n]1 >> [C]([*:20]) (=[O]):[a:2]2:[c,n:3]:[c,n:4]:[c,n:5]:[c,n:6]:[c,n:7]2:[NH]([*:9])</chem>
10 NC-switch		<chem>[*:1]:[c]([*:2]):[c:10]([*:3]):[n]([*:4]):[*:5] >> [*:1]:[n]([*:2]):[c:10]([*:3]):[c]([*:4])[*:5]</chem>

Table 2. Rule categories and number of rules

<i>Functional group transformations</i>	
Alcohols	3
Amide	18
Amine	10
Carbonyl	17
Carboxylate	29
Catechol	4
Ester	4
Ether	4
Guanidine	10
Halogen	3
Nitro	6
Phenol	3
Sulfonamide	6
Urea	16
<i>Molecular framework modifications</i>	
Ring break	14
Ring contract	5
Ring expand	3
Ring form	15
Ring replace	6
Conformation	2
Miscellaneous	8

Total = 186 rules.

**Figure 2.** (a) Illustration of the ring-break transformation (rule #9 in Table 1). (b) Specific example of the ring-break rule in converting PTK787 to the anthranilic amide analog AAL993.¹³

inspection of this many hits and permit archiving for later study. We selected the commercially available RENE data analysis tool as one that (1) understands cheminformatics and Daylight parameters, (2) permits search and sort capabilities, and (3) uses Oracle-based information storage for data manipulation and archival.¹⁴

Many of the chemical structures that Drug Guru creates are reasonable target structures that, admittedly, will be obvious to most experienced medicinal chemists. This is

not surprising since Drug Guru was programmed with rules derived from the traditional medicinal chemistry knowledge base. Even at this level, the program provides value due to its comprehensiveness. Importantly, in test situations, chemists have additionally indicated that non-obvious structures are also included, thus making the use of Drug Guru an even more fruitful exercise—see the sildenafil example below. Less experienced medicinal chemists gain insight from Drug Guru as a starting place for learning about analog design. The references and example structures provided with each rule provide a starting place for learning practical medicinal chemistry (see Rule History description below).

Drug Guru will occasionally generate low quality, non-drug-like structures (three contiguous heteroatoms as an extreme example). Such output structures result from extrapolating specific rules to general situations, not all of which are relevant, or a practical failure to consider every possible situation to which a SMIRKS code may apply. We have opted not to discard the low quality structures: (1) in practice, the number of low quality structures is small, less than 5% of the output, and not a major nuisance. (2) The unorthodox application of rules sometimes results in very novel output which might lead to an ‘outside-the-box’ idea. Our current strategy with output lists of less than 200 entries is to let the medicinal chemists themselves judge ‘goodness/badness’ of the structures, and the default usage is to not apply any structural or numerical ranking. Further refinement of the rules and/or addition of an optional computational filtering step will reduce the number of low quality structures. In cases of output lists with greater than 200 entries or where other scientific information is available, some filtering and/or ranking of output is needed. Ranking based on calculated physical property is an option which is further described below.

Since it is possible that different rules can result in identical output structures, duplicate entries are grouped in a final step prior to display of all of the results to the user in web page format. A typical output page using two rules from Table 1 is shown in Figure 5. A ‘history’ link is available to provide information on the structural transformation, for example, scientific basis and scope of the transformation, examples of application within medicinal chemistry history including proprietary corporate history, and any interesting unpublished lore about the transformation. An example history page is shown in Figure 6 for the hydroxy-to-methoxy rule. In addition to automatically archiving results in a user area after accessing Drug Guru, options for sharing results with other scientists or exporting data are provided.

2.4. Special features of Drug Guru

While the default usage of Drug Guru by the medicinal chemist requires only input of a structure and pressing the ‘Run’ button, there are several user options that are provided on an ‘expert user’ page. When an input structure has more than one occurrence of the functional group to which a rule applies, the question of how

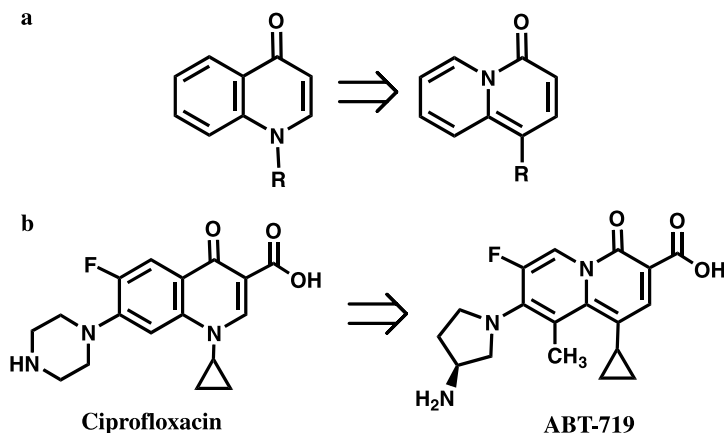


Figure 3. (a) Illustration of the ‘NC-switch’ rule (rule #10 in Table 1). (b) Specific example of the ‘NC-switch’ rule in converting the antibacterial drug ciprofloxacin to ABT-719.²⁷

Drug Guru: Edit Experiment	
Basic Information:	
Title of Experiment:	<input type="text" value="sildenafil_results"/>
Notes: (2000 chars max)	<input type="text" value="This Drug Guru experiment will generate structural analogs of sildenafil."/> <input type="text" value="sildenafil."/>
Enter Starting Structure:	
Enter Structure:	<input type="text" value="A-325043"/> <small>(enter SMILES or A-number)</small>
Read from a File:	<input type="text"/> <input type="button" value="Browse..."/> <input type="checkbox"/> This is a batch experiment
Specify Experiment Type: <small>(learn more about Experiment Types)</small>	
<input checked="" type="checkbox"/> General Structural Transformation <input type="checkbox"/> Increase solubility of input structure <input type="checkbox"/> Decrease serum binding of input structure <input type="checkbox"/> Design around potential metabolism hot spots <input type="checkbox"/> Expert Mode: Select individual rules from the groups	
<input type="button" value="Cancel"/>	<input type="button" value="Save"/>
<input type="button" value="Return to My Home"/>	<input type="button" value="Run >>"/>
<input type="button" value="Save & Return to My Home"/> <input type="button" value="Run Experiment's Rules"/>	

Figure 4. The structure entry web page of Drug Guru. In this example, the structure of sildenafil is accessed by entering the corporate code, A-325043, for this drug.

many times Drug Guru applies the rule needs to be answered. The example of carrying out the hydroxy-to-methoxy transformation on the two hydroxy groups of dopamine, see Figure 7, best illustrates this feature. In the default mode, only the two mono-methylated resulting structures are obtained. Optionally, an exhaustive application of the rule produces a third di-methylated structure. This differentiation between single and exhaustive application of a rule becomes particularly important for rules involving replacement of hydrogen atoms. In this case, selection of exhaustive replacements can lead to a very large number of output structures.

Another user option is to allow multiple generations of application of the set of Drug Guru rules to an input structure. In the default setting, a single round of transformations is applied. Optionally, the output structures from the first round can be automatically re-submitted for a second round of transformations. This has the interesting effect of generating quite novel molecular structures for evaluation. An example of a two-genera-

tion experiment is shown in Figure 8. In practice, the combinatorial expansion leads to a very large number of structures (thousands) for a rule set selection of 100–200 rules; therefore, the number of generations is currently limited to four.

Within Drug Guru, there is an option provided to study the output list according to calculated physical property, such as log *P*, rule-of-5, polar surface area (PSA), or rotatable bonds. In test studies with known drugs, no obvious bias or uneven trend in the calculated physical properties of the output molecules was evident: structures with a continuum of both increased and decreased physical properties were observed with approximately equal frequency. A typical distribution is shown in Figure 9 using Gleevec as input. Sorting the output list according to physical property is provided as a user option and can facilitate identifying structures of high interest. Of particular value is the mathematical difference in property when comparing the input and output structures. When the ‘Δ-property’ column is selected

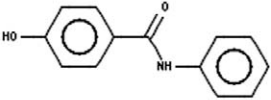
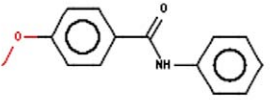
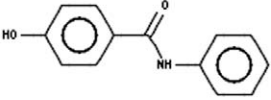
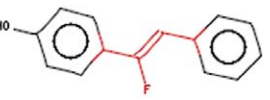
Rule	Generation	Parent Structure	Structure
hydroxy to methoxy	1		
		enlarge history	enlarge history
amide to fluoroolefin	1		
		enlarge history	enlarge history

Figure 5. Example output page from Drug Guru using rules 2 and 3 from Table 1. The output page contains four columns, with each separate output structure as a new row. Column 1 is the name of the rule, column 2 is the generation number, and columns 3 and 4 are the input and output structures, respectively, shown as a 2D depiction. Atoms directly involved in the transformation are highlighted in color to facilitate visual inspection. This example shows only two output structures, but in practice, lists of 50–150 structures are common.

for sorting, the user can quickly tabulate all of the structural transformations that give a desired result. For example, users can see all of the changes that lead to an increase in ClogP or a decrease in PSA. In Figure 10, a Δ -PSA calculation is illustrated.

As a critical aspect of Drug Guru, we note that the rule list is not static, and new rules can be added at any time. For example, new structural transformations reported in the literature or proprietary research discoveries of potential general applicability are excellent sources of new rules. There is also the capability of creating subsets of rules for special purposes. For example, in addition to the set of general transformation rules described above, we have included separate sets of rules for increasing solubility, decreasing albumin binding, or decreasing metabolic liability of the input structure. These rules are less well documented and more anecdotal in character, but nonetheless, still very useful in practice. As an example of this kind of additional Drug Guru rule, one of the metabolism rules is a ‘pyridine-block’ rule which adds a methyl group to the *ortho*-position of a pyridine. This rule is based on the strategy to sterically block the facile oxidation of pyridines with an *ortho* substituent. An example from the Roche group of an application of this design rule has recently been published.¹⁵

3. Results

3.1. Example of Drug Guru applied to known drug system

Sildenafil (Viagra[®]) is a phosphodiesterase-5 inhibitor approved in 1998 for treatment of male erectile dysfunction.

In the application of 186 rules to the structure of sildenafil, 91 output structures were created by Drug Guru. While discussion of the entire output list is beyond the scope of this publication, a few illustrative examples will be given.

Five of the 10 rules listed in Table 1 are applicable to sildenafil and yield output structures represented in Figure 10. Five rules do not apply because their functional groups are not present in the input structure. The homologation-C rule (extend every substituent on a ring by one methylene unit) gives several non-redundant output structures and only two, **36** and **39**, are illustrated. The ethano-to-S and ether-to-thioether rules give structures **62** and **69**, respectively. In this example, the more interesting and diverse structures result from molecular framework changes. The NC-switch rule described above gives rise to two structures possessing new [6.5] heterocyclic core rings, output structures **64** and **65**. Gratifyingly, structure **64** possesses the heterocycle found in vardenafil (Levitra[®]), a drug in the same pharmaceutical class as sildenafil and shown in Figure 11. Another NC-switch output, structure **65**, possesses a heterocycle not previously encompassed within sildenafil or vardenafil patents.¹⁶ In another example of a framework change, the ring-break rule results in output structures **60** and **78** (only **78** is illustrated) in which the aromatic 5-membered ring of sildenafil is opened to give an amino-ketone structure. An intramolecular hydrogen bond between the amino and carbonyl groups of **78** has the potential of forming an intramolecular H-bond, and thus mimicking the aromatic ring of sildenafil. As discussed above, this ‘virtual’ ring strategy has precedent in other studies. No literature reports of studies of chem-

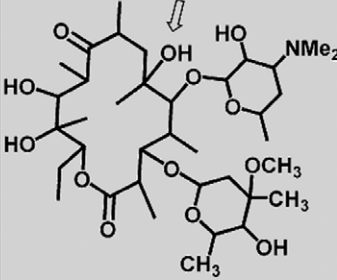
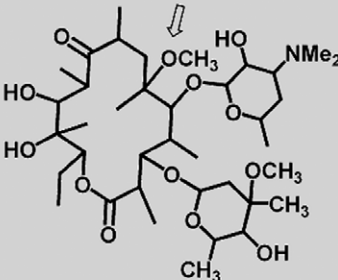
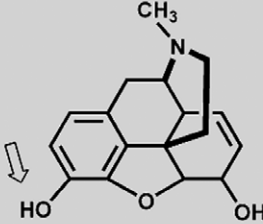
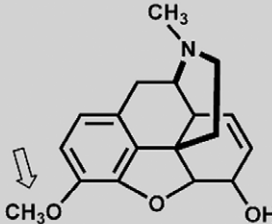
Rule Details	
Rule Name:	hydroxy to methoxy
Rule Group:	Alcohol
Description:	<p>This transformation will methylate any hydroxy groups that are present in the molecule.</p> $R-OH \Rightarrow R-OCH_3$ <p>Two classic examples of a hydroxy to methoxy transformation are shown below:</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p>erythromycin</p> </div> <div style="text-align: center;">  <p>clarithromycin</p> </div> </div> <div style="display: flex; justify-content: space-around; align-items: flex-end; margin-top: 20px;"> <div style="text-align: center;">  <p>morphine</p> </div> <div style="text-align: center;">  <p>codeine</p> </div> </div> <p>A nice review of hydroxy structural replacements can be found in the following review: Patani, G. A. and LaVoie, E. J., Chem. Rev. 96, 3147-3176 (1996), particularly pages 3150-3152.</p>

Figure 6. Example of Rule History Page.

ical systems related to structures **60**, **65**, and **78** in inhibition of phosphodiesterase-5 or utility in treatment of male erectile dysfunction could be found. Therefore, compounds based on these structures may represent interesting targets for further analoging around sildenafil and vardenafil. The exact structure of vardenafil, shown in Figure 11, is produced from sildenafil by Drug Guru in a two-generation run when both the homologation-C and NC-switch rules are applied. It should be emphasized that information from sildenafil structure-activity studies was not used in the creation of the rules of Drug Guru; therefore, these examples shown here are truly derived from the history of medicinal chemistry, implemented as computer-coded rules-of-thumb for compound design Figure 12.

3.2. Comparison with other computational chemistry software

The overall objective of Drug Guru is to facilitate drug discovery, a goal that is shared with other computer-aid-

ed drug design software: docking and scoring programs, such as DOCK, FlexX, AutoDock, GLIDE, ChemScore, and GOLD, de novo ligand construction programs, such as LigBuilder, SkelGen, Ludi, GrowMol, and SPROUT, scaffold-hopping programs such as LeapFrog, EA-Inventor, and FEPOPS, and pharmacophore analysis programs, such as COMFA, Disco, Catalyst, and GASP.¹⁷ However, these other programs have operational strategies that are fundamentally different from Drug Guru. Drug Guru does not rely on a pre-existing database of ligand structures or utilize energy-based fitness functions to score or assemble ligands. No prior structure-activity data are required. Drug Guru uses a set of pre-selected and well-defined medicinal chemistry rules to construct new structures. Ranking is optional and currently is based on calculated physical properties. ‘Comparison’ of Drug Guru with the computer programs listed above is useful mainly in thinking about synergy, rather than ranking performance capabilities. Many of these programs could conceivably be used to help analyze the output of Drug Guru. We have

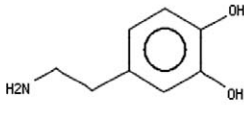
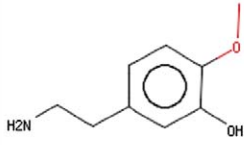
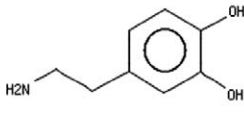
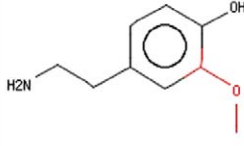
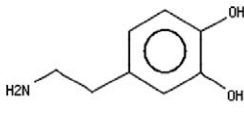
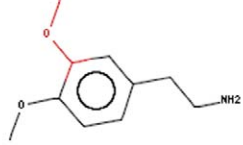
History			
Rule	Generation	Parent Structure	Structure
hydroxy to methoxy	1	 enlarge history	 enlarge history
hydroxy to methoxy	1	 enlarge history	 enlarge history
hydroxy to methoxy	1	 enlarge history	 enlarge history

Figure 7. Example of an ‘exhaustive’ application of a Drug Guru rule. The first two output structures result from a single application of the rule. The last structure results from application of the rule at all possible sites exhaustively.

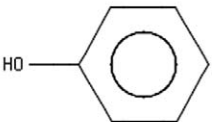
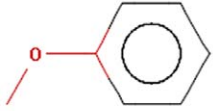
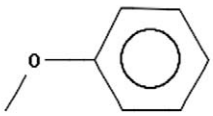
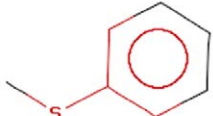
History			
Rule	Generation	Parent Structure	Structure
hydroxy to methoxy	1	 enlarge history	 enlarge history
ether to thioether	2	 enlarge history	 enlarge history

Figure 8. Example of a ‘multi-generation’ use of Drug Guru. The generations are indicated in column 2. In generation 1, phenol is transformed to anisole by the hydroxy-to-methoxy rule. In generation 2, the anisole formed in generation 1 is transformed to thioanisole by the ether-to-thioether rule.

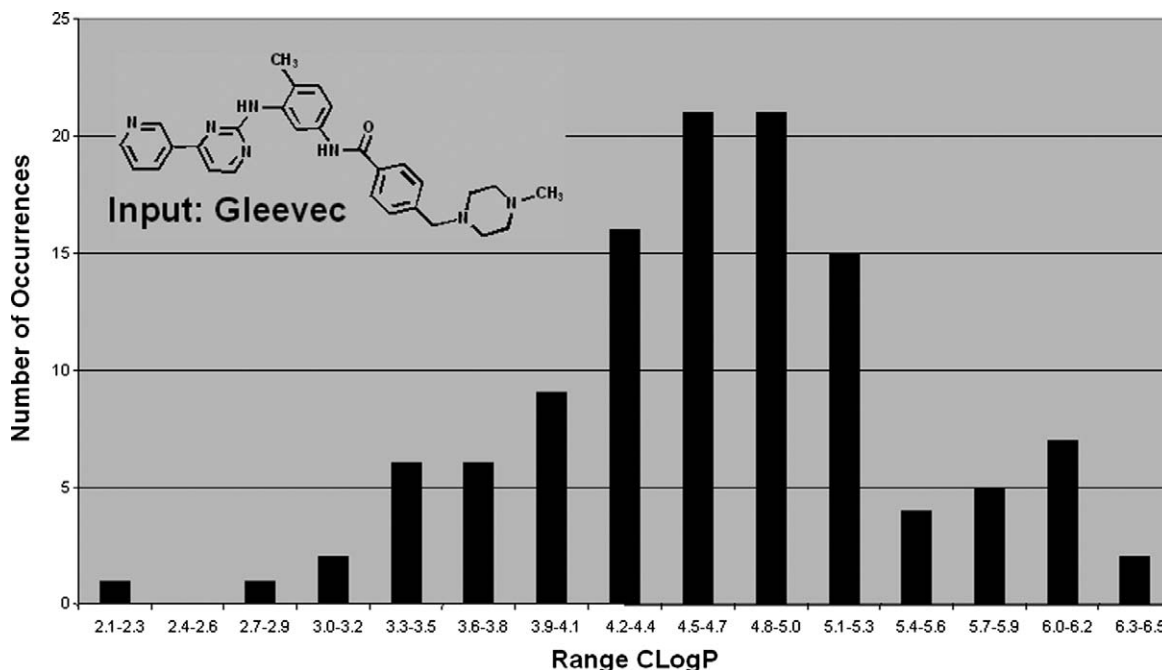


Figure 9. Distribution of the Clog *P* values that are calculated for the 116 output structures resulting from using Gleevec as input structure. Gleevec has a Clog *P* of 4.5 and would be located at the center of the distribution.

History			TPSA		
Rule	Generation	Parent Structure	Structure	Polar Surf.	Δ Polar Surf.
amide to fluoroolefin	1			20.23	-29.1
		enlarge history	enlarge history		

Figure 10. Example of calculation of physical property of a Drug Guru output structure. The polar surface areas, PSA, of the input and output structures are 49.33 and 20.23 Å², respectively. The web page reports the PSA of the output along with the mathematical difference, the Δ -PSA value, of 29.1. In this case, the decrease in PSA is indicated with a minus sign. This ' Δ ' column is available for all calculated properties and can be sorted to find all structural changes that give a consistent change, for example, all transformations that lead to a decrease in polar surface area.

already experimented with taking Drug Guru output directly into docking and pharmacophore programs for further evaluation (work not reported here).

We are aware of four commercial products, BIOSTER, EMIL, WABE, and BIOISOSTER, developed independently, and possessing some features in common with Drug Guru. Unfortunately, published scientific descriptions of each are limited or unavailable, precluding a detailed comparison. The following brief description is intended to draw attention to this area and to stimulate further research. The first commercial package, BIOSTER (bioisosterism), is a database of pairs of molecules differing in one structural element—termed bioanalogous pairs in the original literature citation.¹⁸ A search of this database will generate literature examples of pairs of compounds that illustrate many of the same transformations encoded within Drug Guru. Rather than represent the changes as a list of general rules, as

in Drug Guru, BIOSTER compiles a large number of specific examples (14,300 bioanalogous pairs in the 2005.1 release). Drug Guru differs from BIOSTER in showing the structural changes within the context of a single input molecule, rather than listing literature examples. The second commercial package, EMIL (example mediated innovation for lead evolution), combines a database analogous to the BIOSTER database (3500 'optimization schemes' in the 2003 v.2.4 release) with structure input and viewing that is analogous to Drug Guru.¹⁹ Unlike Drug Guru, the transformations are not collectively organized into rules-of-thumb written in SMIRKS language, but rather each literature example of molecule pairs serves as a separate rule. Like Drug Guru, new transformations can be added to EMIL via programming within the EMIL software. EMIL is installed as a stand-alone application (more recent versions are optionally web-based) in contrast to both BIOSTER (requires ISIS environment, Molecular Design

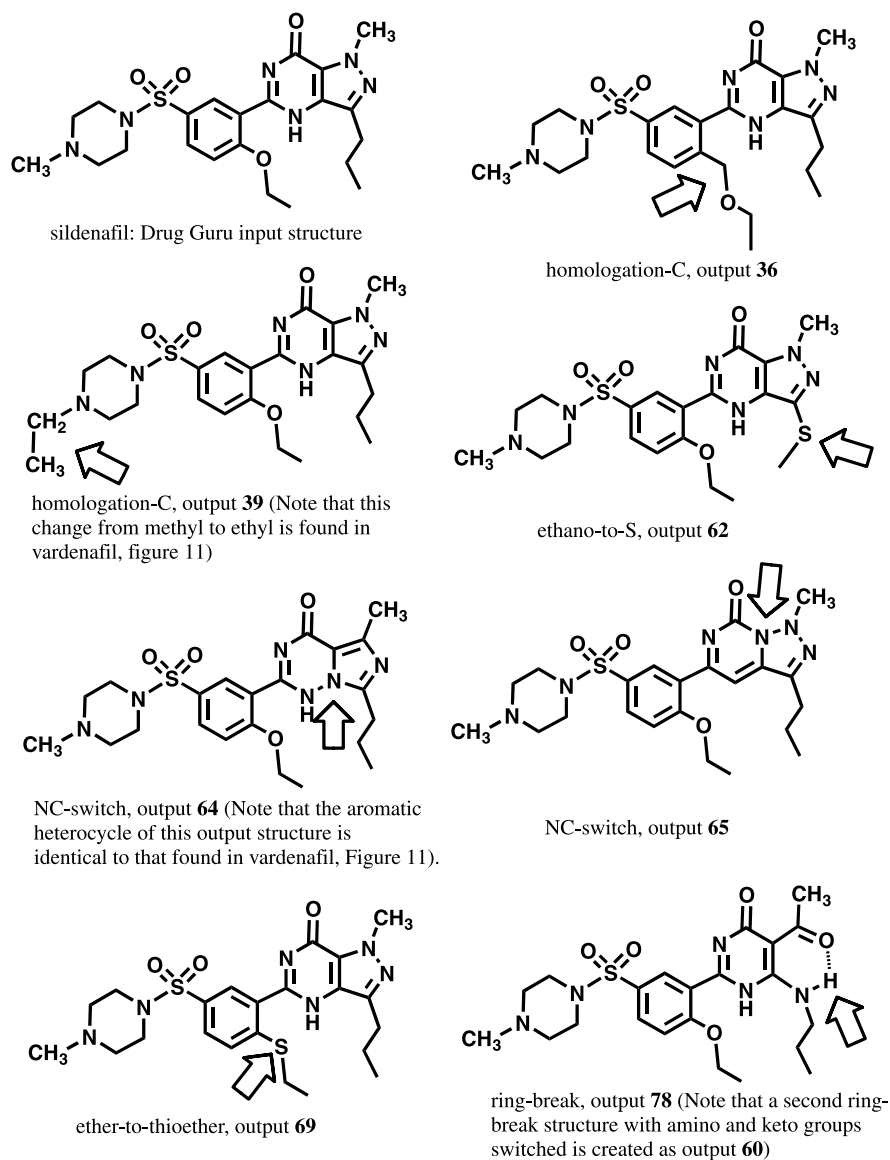


Figure 11. Examples of Drug Guru output for the input structure of sildenafil. The location of structural alteration is denoted with an arrow. The total list contained 91 structures, and the arbitrary ranking within the list is designated here.

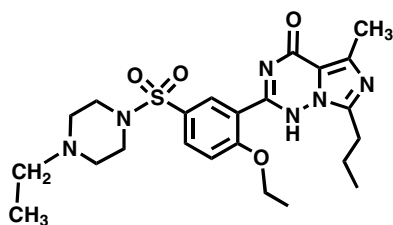


Figure 12. Chemical structure of vardenafil. This structure is created by Drug Guru in a two-generation run using sildenafil (Fig. 11) as input. The two generations use (1) the homologation-C rule, and (2) the NC-switch rule.

Limited) and Drug Guru (requires Daylight toolkit and Oracle environments). A third commercial package, WABE, generates isosteres of an input molecule according to an atom-replacement algorithm with an output ranking based on electrostatic similarity or receptor-

based scoring.²⁰ The fourth commercial package, BIO-ISOSTER, transforms an input structure according to approximately 300 'target-specific' changes derived from literature reports of kinase, protease, ion channel, phosphodiesterase, and nuclear receptor ligands.²¹ In addition to these commercial products, publications from the Merck, Novartis, GlaxoSmithKline, Celera Genomics, and Organon computational groups mention proprietary databases of functional group replacements and implementation within compound design programs.^{22–26}

3.3. Limitations of this approach

From the listing of commercially available packages and publications discussed above, it is evident that molecular replacement rule-based software for drug design is an emerging and exciting area of research. In the case of Drug Guru, we feel that there are two main limitations

that happen to also be limitations common to other computer-aided drug design programs: (1) imperfect ranking scheme and (2) lack of synthetic chemistry knowledge. The first limitation is related to the age-old question of ‘how do you recognize a drug when you see it?’ The example of the sildenafil analogs illustrated above is telling. As a chemist confronted with a list of 91 suggestions, how does one know, a priori, that number **64** (the vardenafil analog) is the structure on which to focus attention? Also important is the question of how good the remaining 90 structures are? Unequivocal answers to these questions are not possible. The main intent of Drug Guru is to generate ideas. We submit that the idea-list that a medicinal chemist has in mind at any one time is by its nature incomplete, and Drug Guru is designed to help fill that gap. However, idea generation is not usually cited as the rate-determining step in drug discovery. The ranking that Drug Guru currently provides is based on calculated physical properties and has acknowledged weaknesses. We speculate that assessment of ‘success frequency’ for individual rules may provide a novel way of ranking compounds for future evaluation. Chemists could conceivably prioritize transformations that have historically performed well in creating drug candidates, that is, transformations with the best ‘yield.’ Confounding any single prioritization scheme is the fact that analog design strategies at the outset of a research project differ from those in a mature program close to a clinical candidate. Currently in Drug Guru, all design rules are treated equally. We welcome suggestions on the optimal computational protocol for differentially weighting the rules.

Assessing synthetic feasibility is an important part of target evaluation. In an ideal setting within any computer-aided drug design software, the user would receive quick feedback whether suggested output structures were conveniently accessible from available starting materials, or would require multi-step syntheses with varying degrees of difficulty and precedent. In fact, when Drug Guru was evaluated with a test audience of 25 medicinal chemists, incorporation of synthetic chemistry knowledge into the evaluation process was the single most requested improvement. Unfortunately, a fully satisfactory computational protocol to conveniently, and reliably, assess laboratory access to computer-generated ideas is currently not possible. One simple, yet speculative, possibility for Drug Guru would be to assign a ‘difficulty ranking’ to each rule. Rules that correspond to a common laboratory operation, such as a methylation (the hydroxy-to-methoxy rule, rule 3, [Table 1](#)), would be ranked ‘high priority.’ Rules that correspond to potentially laborious multi-step procedures if interpreted literally, such as ring transformations (for example, NC-switch, rule 10, [Table 1](#)), would be ranked ‘low priority.’ Subsequent sorting of the output list by the medicinal chemist would yield a list of the more synthetically accessible suggestions high on the list. More extensive software links between Drug Guru and organic chemistry synthesis design programs, such as Lhasa, Chiron, CAESA, etc, or reaction databases, such as REACCS, can be envisioned. Research into incorporating synthetic chemistry input into Drug Guru is in progress.

4. Conclusions

In this report, we describe a new web-based computer application that applies a library of medicinal chemical transformation rules to an input structure and then permits evaluation of the resulting output structures. The name of the new software captures this intellectual process of drug generation using rules with the acronym ‘Drug Guru.’ It is hoped that Drug Guru will provide an intellectual guide to medicinal chemists in the increasingly difficult task of discovering new compounds as potential drug candidates.

Acknowledgments

Funding for the Drug Guru project was provided by the Abbott Laboratories Innovation Grant Program. Acknowledgment is also made to the Abbott Laboratories Medicinal Chemists who contributed many ideas for structural transformations (rules). Cheminformatics and web programming was carried out by Moonview Consulting, LLC.

References and notes

1. Wermuth, C. G., Ed., *The Practice of Medicinal Chemistry*; 2nd Ed.; Academic: New York, 2003.
2. Sneader, W. *Drug Prototypes and their Exploitation*; John Wiley & Sons: New York, 1996.
3. Burger, A. *Prog. Drug Res.* **1991**, *37*, 287–371.
4. Chen, X.; Wang, W. *Annu. Rep. Med. Chem.* **2003**, *38*, 333–346.
5. Patani, G. A.; LaVoie, E. J. *Chem. Rev.* **1996**, *96*, 3147–3176.
6. Thornber, C. W. *Chem. Soc. Rev.* **1979**, *8*, 563–580.
7. Lipinski, C. A. *Annu. Rep. Med. Chem.* **1986**, *21*, 283–291.
8. Spatola, A. *Ann. Rep. Med. Chem.* **1981**, *16*, 199–209.
9. Rudinger, J. In *Drug Design*; Ariens, J., Ed.; Academic: New York, 1971; Vol. 2, pp 319–419.
10. Drug Guru is a trademark of Abbott Laboratories. The software is proprietary to Abbott Laboratories and Moonview Consulting, LLC. Readers desiring more information are encouraged to contact the first author.
11. James, C. A.; Weininger, D.; Delaney, J., *Daylight Theory Manual*, Daylight Chemical Information Systems, Inc., Irvine, CA. www.daylight.com/dayhtml/doc/theory/theory.toc.html.
12. Jung, M. E.; Piizzi, G. *Chem. Rev.* **2005**, *105*, 1735–1766.
13. Furet, P.; Bold, G.; Hofmann, F.; Manley, P.; Meyer, T.; Altmann, K.-H. *Bioorg. Med. Chem. Lett.* **2003**, *13*, 2967–2971.
14. RENE: www.daylight.com/products/rene.html.
15. Trejo, A.; Arzeno, H.; Browner, M.; Chanda, S.; Cheng, S.; Comer, D. D.; Dalrymple, S. A.; Duntun, P.; Lafargue, J.; Lovejoy, B.; Freire-Moar, J.; Lim, J.; McIntosh, J.; Miller, J.; Papp, E.; Reuter, D.; Roberts, R.; Sanpablo, F.; Saunders, J.; Song, K.; Villasenor, A.; Warren, S. D.; Welch, M.; Weller, P.; Whiteley, P. E.; Zeng, L.; Goldstein, D. M. *J. Med. Chem.* **2003**, *46*, 4702–4713.
16. Sildenafil patents US 5250534, US 5346901, US 5719283; vardenafil patents US 6362178, US 6566360.
17. Representative references: docking/scoring programs: (a) Perola, E.; Walters, W. P.; Charifson, P. S. *Proteins: Struct. Funct. Bioinf.* **2004**, *56*, 235–249; De novo pro-

- grams: (b) Stahl, M.; Todorov, N. P.; James, T.; Mauser, H.; Boehm, H. -J.; Dean, P. M. *J. Comput. Aided Mol. Des.* **2002**, *16*, 459–478; Scaffold hopping programs: (c) Jenkins, J. L.; Glick, M.; Davies, J. W. *J. Med. Chem.* **2004**, *47*, 6144–6159; Pharmacophore programs: (d) Patel, Y.; Gillet, V. J.; Bravi, G.; Leach, A. R. *J. Comput. Aided Mol. Des.* **2002**, *16*, 653–681.
18. Ujvary, I. *Pestic. Sci.* **1997**, *51*, 92–95, The BIOSTER program is commercially available from Accelrys Inc., www.accelrys.com.
 19. Fujita, T. In *Trends in QSAR of Molecular Modeling 92*; Wermuth, C. G., Ed.; ESCOM: Leiden, 1993; pp 143–159, The EMIL program is commercially available from CompuDrug International, Inc, www.compudrug.com.
 20. Sayle, R., Skillman, G.; Nicholls, A., *Abstracts*, 227th National Meeting of the American Chemical Society, Anaheim, CA, March 28–April 1, 2004, American Chemical Society: Washington, DC. The WABE program is commercially available from OpenEye Scientific Software, <http://www.eyesopen.com/products/applications/wabe.html>.
 21. Balakin, K. V.; Tkachenko, S. E.; Okun, I.; Skorenko, A. V.; Ivanenkov, Y. A.; Savchuk, N. P.; Ivashchenko, A. A.; Nikolsky, Y. *Chem. Oggi* **2004**, *22*, 15–18, The BIOISO-STER program is commercially available from Chemical Diversity Labs, Inc. www.chemosoft.com.
 22. Sheridan, R. P. *J. Chem. Inf. Comp. Sci.* **2002**, *42*, 103–108.
 23. Ertl, P. *J. Chem. Inf. Comp. Sci.* **2003**, *43*, 374–380.
 24. Lewell, X. Q.; Jones, A. C.; Bruce, C. L.; Harper, G.; Jones, M. M.; Mclay, I. M.; Bradshaw, J. *J. Med. Chem.* **2003**, *46*, 3257–3274.
 25. Southall, N. T.; Ajay *J. Med. Chem.* **2006**, *49*, 2103–2109.
 26. Wagener, M.; Lommerse, J. P. M. *J. Chem. Inf. Model.* **2006**, *46*, 677–685.
 27. Li, Q.; Chu, D. T. W.; Claiborne, A.; Cooper, C. S.; Lee, C. M.; Raye, K.; Berst, K. B.; Donner, P.; Wang, W.; Hasvold, L.; Fung, A.; Ma, Z.; Tufano, M.; Flamm, R.; Shen, L. L.; Baranowski, J.; Nillius, A.; Alder, J.; Meulbroek, J.; Marsh, K.; Crowell, D.; Hui, Y.; Seif, L.; Melcher, L. M.; Henry, R.; Spanton, S.; Faghieh, R.; Klein, L. L.; Tanaka, S. K.; Plattner, J. J. *J. Med. Chem.* **1996**, *39*, 3070–3088.