JOURNAL OF CHEMICAL INFORMATION AND MODELING

Article

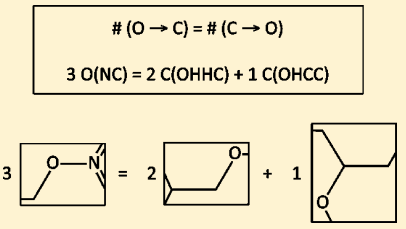
pubs.acs.org/jcim

Lattice Enumeration for Inverse Molecular Design Using the Signature Descriptor

Shawn Martin*

Department of Computer Science, University of Otago, P.O. Box 56, Dunedin 9054, New Zealand

ABSTRACT: We describe an inverse quantitative structure−activity relationship (QSAR) framework developed for the design of molecular structures with desired properties. This framework uses chemical fragments encoded with a molecular descriptor known as a signature. It solves a system of linear constrained Diophantine equations to reorganize the fragments into novel molecular structures. The method has been previously applied to problems in drug and materials design but has inherent computational limitations due to the necessity of solving the Diophantine constraints. We propose a new approach to overcome these limitations using the Fincke−Pohst algorithm for lattice enumeration. We benchmark the new approach against previous results on LFA-1/ICAM-1 inhibitory peptides, linear homopolymers, and hydrofluoroether foam blowing agents. Software implementing the new approach is available at www.cs.otago.ac.nz/homepages/smartin.



INTRODUCTION

Computer-aided molecular design has the potential to greatly accelerate the development of new and useful chemicals, materials, and drugs. One approach to molecular design is through the use of quantitative structure−activity relationships, known as QSARs. In this approach, we first determine a *forward* QSAR capable of predicting biological activities or chemical properties from molecular structures. A forward QSAR can be obtained using numerous different model equations and accompanying regression-based algorithms,[1] or even empirically via expert knowledge and experiment. Next, we invert the QSAR, now searching for molecular structures that yield desirable properties under the model. This is known as the *inverse* QSAR problem.

Methods for solving the forward QSAR problem are numerous,[1] while solutions to the inverse QSAR problem are few. Perhaps the best-known solution to the inverse QSAR problem is virtual screening.[2] In this approach, molecules in a database are evaluated using a QSAR to identify structures that yield desired properties. However, a virtual screen can identify only molecular structures already in the database, it cannot suggest novel structures that may perform better than known compounds. Methods for suggesting truly novel structures include random search,[3] combinatorial and heuristic-based enumeration,[4−6] graphical reconstruction,[7−9] mathematical programming,[10−13] stochastic optimization,[14−19] and our approach using the signature descriptor.[20−23]

All of the above approaches have limitations, both in terms of computation and accuracy. The mixed integer nonlinear programming approaches[10−13] are computationally expensive and can be inaccurate if trapped in local minima. Simulated annealing, genetic algorithms, and tabu search[14−19] are computationally efficient but random: these algorithms can find near-optimal solutions, but are not enumerative (even given limits) so may miss potentially important molecules. Exhaustive enumeration of molecular structures with given constraints is often computationally prohibitive.[4−9]

Our approach is based on exhaustive enumeration.[21] Although our method has been successfully applied to various applications,[20−24] we have in many cases struggled to overcome computational issues related to difficulties in the enumerative steps in our algorithm, in particular those associated with solving a constrained system of linear Diophantine equations.[25] In addition, our method has accuracy limitations inherent to our use of QSAR. This is not due to the particular type of QSAR we use, but rather the general fact that QSAR methods are statistical in nature and cannot be trusted to extrapolate much beyond their training sets.[26,27]

In this paper, we propose a new approach that accommodates these two difficulties. Our new approach uses lattice enumeration via the Fincke-Pohst algorithm[28] to find solutions to the Diophantine equations *near* the QSAR training set. This allows the user to simultaneously control both the amount of computational effort expended and the solution accuracy expected. The new approach provides higher quality results more quickly, and will therefore allow us to tackle more difficult problems in the future. To demonstrate the advantages of the new system, we provide benchmark comparisons on previous applications, including LFA-1/ICAM-1 inhibitory peptides, linear homopolymers, and hydrofluoroether foam blowing agents. Software implementing the new approach is available at www.cs.otago.ac.nz/homepages/smartin.

BACKGROUND

Any approach to the inverse QSAR problem must somehow address the following inter-related subproblems:[20]

Received: April 2, 2012
Published: June 3, 2012

dx.doi.org/10.1021/ci3001748 | J. Chem. Inf. Model. 2012, 52, 1787−1797

(1) **Descriptor Selection.** Calculations involving molecular structure require an encoding of the structure in a manner amenable to mathematical and statistical techniques. This encoding is typically accomplished by means of a molecular descriptor. In our approach we use the signature molecular descriptor.[29]

(2) **Forward QSAR.** Once a descriptor has been selected, a QSAR (or QSARs) must be obtained in order to make predictions related to the molecular properties of interest. Examples of properties we have predicted include peptide IC_{50} values[21] and polymer glass transition temperature[20] (T_g).

(3) **Structure Enumeration.** Novel structures must next be obtained which correspond to valid structures within the context of the chosen molecular descriptor. Mathematically, we must enumerate a discrete subset of valid structures within the space of all possible descriptor vectors.

(4) **Solution Filtering.** Due to the huge number of potential molecular structures, there will always be many more solutions than can be reasonably investigated. From these solutions, we must pick the structures most likely to have desirable properties accurately predicted by our QSAR.

Inverse QSAR is difficult because these subproblems are difficult, both in terms of obtaining accurate solutions and in terms of computational burden. However, the subproblems are also closely related. The fundamental contribution of this paper is that we use the relationships between the subproblems to mitigate their individual difficulties. In this section (Background), we provide an overview of our original approach to the inverse QSAR problem according to steps 1−4, along with the limitations inherent in each step. In the next section (Lattice Enumeration), we provide a description of our modified approach, again using steps 1−4, this time with an explanation of how the new approach greatly reduces the various limitations associated with the original approach.

(1) Descriptor Selection. There are thousands of molecular descriptors, all with relative advantages and disadvantages.[30] However, there are only a few descriptors that have been designed specifically to address the inverse QSAR problem. Kier et al. developed a descriptor based on path length counts;[7] Skvortsova et al. developed another path length descriptor which also includes atom and bond type information;[8] and Faulon et al. developed the signature descriptor employed in our approach.[21,29]

The signature molecular descriptor is based on the molecular graph of a molecule, $G = (V_G, E_G)$, where the graph nodes V_G denote the atoms in the molecule and the graph edges E_G correspond to the bonds between those atoms.[21] In the molecular graph, both the nodes and the edges are labeled: nodes are labeled by the element names of the atoms and edges are labeled according to bond type (a single bond is unlabeled, a double bond is labeled by "=", a triple bond is labeled by "t", and an aromatic bond is labeled by "p"). In this context, a molecule is characterized by a set of canonical subgraphs, each rooted on a different vertex with a predefined level of branching, which we refer to as the height h. The branching of a vertex is an extended degree sequence that describes the local neighborhood, up to the distance h away from the root.

We define an atomic signature, σ_x^h, as the canonical subgraph of G consisting of all atoms a distance h from the root x. A molecular signature, Σ^h, is then the set of atomic signatures and the number of times that they occur in the molecular graph. For example, a molecule's height zero signature is simply its chemical formula, i.e. the count of the occurrence of each atom in the molecule. The atomic signatures are by construction inter-related, allowing information about the overall structure of the molecule to be recovered.

The atomic signatures make up the set of molecular descriptors for a molecule. These are expressed in terms of a string of characters that correspond to the canonized subgraph in a breath-first order. Branch levels are indicated by a set of parentheses following the parent vertex. An example of the molecular signature for nitroglycerine is given in Figure 1. A detailed discussion of signature, including prediction performance and comparisons with other descriptors, has been previously published.[29]

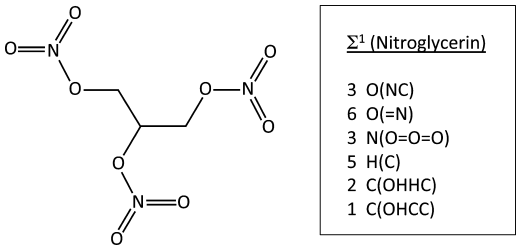


Figure 1. Height one molecular signature for nitroglycerin. On the left, we show the molecular graph for nitroglycerin, and on the right, we give its corresponding height one signature Σ^1 vector. The molecular signature consists of occurrence numbers followed by atomic signatures. The first row, for example, contains 3 O(NC), which indicates that the fragment N−O−C occurs 3 times in nitroglycerin.

(2) Forward QSAR. Once a molecular descriptor has been selected, we compute a QSAR model capable of predicting a property of interest. A QSAR model is computed using a training set of molecular structures with known properties. Once a model has been obtained, we can make predictions on structures not in our training set. Such predictions can be used to evaluate novel structures suggested by the overall inverse QSAR approach.

There are numerous options for computing the QSAR model. The most commonly used QSAR is obtained via multiple linear regression. QSAR models can also be obtained using partial least-squares, ridge regression, neural networks, support vector machines, et cetera. The signature molecular descriptor has been used to compute QSARs via stepwise multiple linear regression[20,21,23,29] and support vector machine classification.[25,31,32]

No matter how a QSAR is obtained, the model will suffer from inaccuracies. These inaccuracies are not limited to a particular descriptor, QSAR model, or training algorithm but are instead inherent in the QSAR approach itself. Inaccuracies can be minimized using cross-validation[26] and measures to avoid extrapolation of the model beyond the original training set.[26,27] Both of these issues are important when solving the inverse QSAR problem.[20] The issue of extrapolation is especially problematic due to the fact that our goal is the exploration of *all possible* molecular structures in search of those with desirable properties. Thus we are likely to enumerate molecules very different from molecules in our training set, and hence, our QSAR will be unable to provide accurate predictions in many instances.

dx.doi.org/10.1021/ci3001748 | J. Chem. Inf. Model. 2012, 52, 1787−1797

(3) Structure Enumeration. Suppose we call the space spanned by the signature descriptor the *signature space*. We notice that not every vector in the signature space corresponds to a possible molecular structure. Consider the vector in the signature space consisting of the single atomic signature:[20]

1 C(HHHH)

This vector does not correspond to a molecular signature because atomic signatures rooted at each atom are not present. In particular, a molecular signature containing the above atomic signature must also contain at least four occurrences of the atomic signature H(C). Since our vector does not contain four H(C), it cannot correspond to a molecular signature.

In order to solve the inverse QSAR problem, we must therefore restrict the signature space. We use constraint equations (described next) to achieve this restriction, giving us the *signature chemical space*, or just the chemical space. In the chemical space, we can filter our solutions using the forward QSAR to identify vectors that exhibit desired properties. These vectors can then be used to generate the molecular graphs corresponding to the identified molecular signatures.

Constraint Equations. The constraint equations enforce conditions necessary for reconstructing molecular graphs from vectors in the signature space. There are in general two types of constraint equations: the graphicality equation and the consistency equations. Additional constraints may also be imposed. For peptide rings, we included a constraint on the number of amino acids in a ring,[21] and for polymers, we included an equation to ensure that the polymers would repeat.[20]

The graphicality equation is necessary if at least one connected graph is to be constructed from the molecular descriptors. This equation uses only the degree of the vertices in the graph. It is derived from the cyclomatic number (also known as the circuit rank) $z = m - n - 1$, where z is the number of cycles in the graph, m is the number of edges, and n is the number of vertices. Denote the degree sequence of the graph by $N = \{n_1, n_2 ..., n_k\}$ where n_i is the number of vertices of degree i. Then $2m = \Sigma_i\, in_i$ and $n = \Sigma_i\, n_i$ In this case, the degree sequence N is graphical if there exists an integer $z \geq 0$ such that:

$$\sum_{k}^{i=1} (i - 2)n_i + 2 = 2z$$

The graphicality equation is a necessary condition for a graph to be connected and can be computed directly from the height zero molecular signature.

The next set of equations is collectively referred to as the consistency equations. Recall that a molecular signature is a collection of interrelated atomic signatures, where each atomic signature describes a particular atom and its neighboring atoms to a predetermined height. In constructing the signature of a molecule, it is guaranteed that a bond in one atomic signature will match up with a bond in another atomic signature, albeit in reverse order. However, blind reconstruction of the molecule requires equations to enforce these relationships among the atomic signatures. This is done by matching bonds between two atoms of one signature to the bonds involving the same atoms in all other signatures.

We will use the notation σ_i^h to describe the atomic signature of height h of an arbitrary atom i. Using σ_i^h as a reference, any bond between the root and one of its children j must be sought in all other atomic signatures in which the positions of the root and child are the transpose of σ_i^h. We use the notation $\#(\sigma_i^{h-1} \rightarrow \sigma_j^{h-1})$ to depict the number of bond types σ_i^h has in common with σ_j^h, including terminal atoms. Clearly, then $\#(\sigma_i^{h-1} \rightarrow \sigma_j^{h-1}) = \#(\sigma_i^{h-1} \rightarrow \sigma_j^{h-1})$. In the case where $i = j$, then $\#(\sigma_i^{h-1} \rightarrow \sigma_j^{h-1})$ must be even. We note that the signature of a bond is height one less than the height of the molecular signature. When $\#(\sigma_i^{h-1} \rightarrow \sigma_j^{h-1})$ is computed, we have to transpose the root i with a child j. While the neighborhood of i was initially probed up to height h, the transposed signature with new root j probes the neighborhood of j only up to height $h - 1$.

The consistency equations can be summarized as follows. A molecular signature Σ^h is consistent if the two following conditions are verified:

(a) For all atomic signatures, σ_i^h and σ_j^h in Σ^h, $\#(\sigma_i^{h-1} \rightarrow \sigma_j^{h-1}) = \#(\sigma_i^{h-1} \rightarrow \sigma_j^{h-1})$.

(b) For all σ_i^h in Σ^h, $\#(\sigma_i^{h-1} \rightarrow \sigma_i^{h-1})$ is even valued.

We give examples of consistency equations for a height 1 signature in Figures 2 and 3. Examples of coefficient matrices for systems of consistency equations are given in the Experimental Results section. Additional explanation has also been published.[21] Finally, we note that the constraint equations are a *necessary*, but not sufficient, condition for obtaining a molecule from a signature vector. In other words, a molecule will give rise to a signature vector that satisfies the constraints, but the reverse is not necessarily true: a signature vector that satisfies the constraints may have no corresponding molecule representation. In practice this does not occur too often,

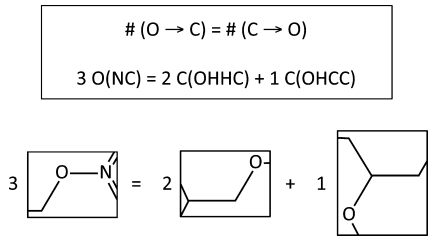


Figure 2. Nonmodulus consistency equation for nitroglycerin. Here we show the consistency equation obtained by matching fragments containing O−C in nitroglycerin. This equation states that O → C must occur the same number of times as C → O over all fragments. In nitroglycerin, O → C occurs 3 times in O(NC), and C → O occurs twice in C(OHHC) and once in C(OHCC).

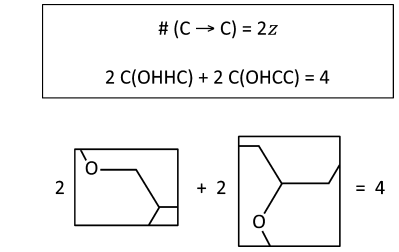


Figure 3. Modulus consistency equation for nitroglycerin. Here we show the consistency equation obtained by matching fragments containing C−C in nitroglycerin. This equation states that C → C must occur an even number of times in all fragments. In nitroglycerin, C → C occurs twice in C(OHHC) and twice in C(OHCC). More precisely, the bond C → C occurs once in C(OHHC), but C(OHHC) occurs twice in nitroglycerin (giving a total of 2); while the bond C → C occurs twice in C(OHCC), and C(OHCC) occurs once in nitroglycerin (again giving a total of 2).

dx.doi.org/10.1021/ci3001748 | J. Chem. Inf. Model. 2012, 52, 1787−1797

although a thorough investigation has not been undertaken. In any case, since the constraints are necessary, we will not miss any potential molecules in our search.

Solving the Constraints. The constraint equations form a system of equations with unknown occurrence numbers x_i corresponding to atomic signatures σ_i^h. Our system is comprised of homogeneous, modulus, and nonhomogeneous equations. The homogeneous equations are consistency equations of type (a), and the modulus equations (also homogeneous) are consistency equations of type (b). Nonhomogeneous equations include the graphicality equation (also a modulus equation), and specialized constraints such as the number of amino acid in a peptide ring[21] or the requirement that a polymer repeat.[20] The modulus equations can be written as homogeneous equations by adding a dummy variable, as was done using the variable z in the graphicality equation.

The solutions to the constraint equations should give molecular signatures that correspond to valid molecular structures. Since valid molecular signatures consist of vectors with non-negative integer occurrence numbers, these solutions should take on non-negative integer values. Thus, our system of equations is Diophantine in nature. Examples of the Diophantine systems we examine can be found in the Experimental Results section.

Our equations form a Diophantine system with solutions restricted to be non-negative. Mathematically, solutions to such a system can be expressed using a Hilbert basis. A Hilbert basis is a set of non-negative integer valued vectors which can be combined linearly using non-negative integer coefficients to obtain every solution to the Diophantine system. Chemically, linear combinations of the Hilbert basis correspond molecular signatures.

In our original approach to the structure enumeration problem, an algorithm adapted from Contejean and Devie[33] was implemented to solve our system of linear Diophantine equations. This algorithm uses a geometric interpretation of Fortenbacher's algorithm[34] and will return the Hilbert basis. Although the computational complexity of the Hilbert basis problem is unknown, counting the Hilbert basis is #P-hard, which implies that solving the Hilbert basis is more intractable than an NP-complete problem.[35] In our experience with the Contejean–Devie solver, we have been able to obtain the full Hilbert basis in the case of peptide rings.[21] We obtained a full basis for a subset of the constraint equations using hydrofluoroether foam blowing agents,[23] and a partial Hilbert basis in the case of polymers.[20] In the case of glucocorticoid receptor ligands, the Contejean–Devie solver was abandoned in favor of brute-force enumeration.[22] We also used an alternate algorithm for computing the Hilbert basis[36] but had little success. Finally, we developed methods to exploit the sparsity of the constraints by reducing the Diophantine equations prior to finding the Hilbert basis, but again met with limited success.[25]

Our original approach also involved enumerating linear combinations of the Hilbert basis to generate solutions. This is a combinatorial problem because we are taking, for example, n basis vectors p at a time, then multiplying them by coefficients in a certain range (chosen to correspond to molecules up to a certain size). Further, the number of vectors n in a Hilbert basis is a priori unknown (in contrast to the typical situation involving a linear system), so that the enumeration can be easy for certain problems but extremely difficult for other problems. Only in the case of the peptide rings[21] were we able to fully enumerate the desired linear combinations.

Molecular Reconstruction. Once signatures corresponding to valid molecular structures have been determined, we must also reconstruct molecular graphs from the signature vectors. The algorithm we use to enumerate all molecular graphs corresponding to a target signature is based on an isomer enumeration algorithm.[37] This algorithm was adapted to enumerate isomers matching user specified signatures[38] and belongs to the class of orderly algorithms.[39]

The molecular reconstruction algorithm is combinatorial, but the complexity can be controlled using the degeneracy of the signature descriptor. The number of molecules which can be reconstructed from a given target signature is the degeneracy of that signature. Ideally, the degeneracy of a descriptor should be as low as possible while still allowing for high correlation with molecular properties.[40] For signature, the degeneracy can be decreased by increasing the signature height.[38] However, for large signature heights, correlations can result from overfitting. In practice, we find that we have to sacrifice some degeneracy in order to obtain predictive QSAR equations.

(4) Solution Filtering. By enumerating the chemical space, we have in theory all possible molecular signatures corresponding to valid molecular structures. We can now filter these signatures using our forward QSAR to predict which structures have desirable properties. In practice, however, there are typically many more solutions than we can reasonably investigate, even using the QSAR as a filter. In previous applications, we have obtained anywhere from hundreds of solutions for peptide rings to almost a billion solutions for polymers. Furthermore, it is very difficult to assess the quality of a given solution, even if it is predicted to have desirable properties. In most cases, we have resorted to using multiple QSARs as well as secondary properties: for the peptide rings,[21] we used size and topological constraints (we were interested in peptide rings with 9 amino acids only); for hydrofluoroether foam blowing agents,[23] we used multiple QSARs as well as restrictions on size (up to 31 atoms), composition (number of oxygen atoms), and exclusion of cycles; for glucocorticoid receptor ligands, we used multiple QSARs and Lipinski's rule of five;[22] and for polymers, we used multiple QSARs, size restrictions, and a confidence measure on solution accuracy.[20] In each of these applications, the goal has been to produce quality solutions, but in each case a quality solution has had a different interpretation.

LATTICE ENUMERATION

Even though our original approach to inverse QSAR has been successfully applied to various problems, it has some nontrivial limitations. The most prohibitive limitation is computational and involves the structure enumeration step in which we compute the Hilbert basis for the constraint equations. This problem is known to be more intractable than NP-complete, since counting the Hilbert basis is #P-hard.[35] If the Hilbert basis is obtained, we have another computational limitation since we must enumerate linear combinations of the basis vectors to obtain solutions. This is a combinatorial process that is also prohibitive. Finally, successful enumeration of molecular structures leads to a third difficulty, namely the identification of high quality structures from among the huge number (typically millions) of enumerated solutions.

The motivation behind our original approach was to explore the entire chemical space in order to identify molecular structures with desirable properties. Unfortunately, this approach results in a huge amount of wasted effort. We are

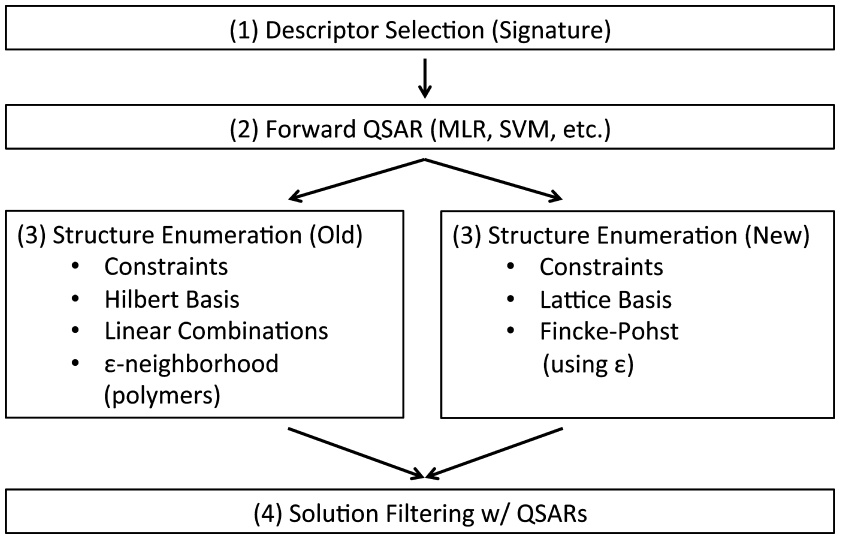


Figure 4. Differences between the previous and new approaches. Here we show a flowchart comparing the previous approach with the new approach. Proceeding from the top to the bottom, steps 1 and 2 are unchanged between the two methods (note MLR is multiple linear regression). In the previous approach, step 3 solved for the Hilbert basis and enumerated linear combinations of the basis vectors, as shown on the left. The ε-neighborhood filter was first used for the polymer data set.[20] In the new approach, step 3 solves for a Lattice basis and uses the Fincke–Pohst algorithm to enumerate solutions, shown on the right. Step 4 is again unchanged between the two approaches.

at great expense enumerating millions of structures, only to discard almost all of them in the final steps.

In contrast, the motivation behind our new approach to inverse QSAR is to enumerate only molecular structures that we know will be of high quality. There are two measures of solution quality for inverse QSAR. First, we are interested in solutions with desirable properties as predicted by the QSAR, and second, we are interested in solutions that are *accurately* predicted by the QSAR. By imposing both of these conditions, we obtain the best possible results from our inverse QSAR.

While the identification of structures with desirable properties as predicted by a QSAR is application specific, the identification of solutions accurately predicted by a QSAR is general. In fact, this is an important problem in both machine learning and statistics, as motivated by the question "how do we know which predictions are accurate and which predictions are inaccurate?" This question has been addressed in cheminformatics with the development of discriminators for prediction accuracy in QSARs.[27,41,42]

For our new approach, we use a solution quality metric reported by Sheridan et al.[27] In a large study using data sets covering eight different properties and tens of thousands of training molecules, they found that the most accurately predicted properties were for molecules with the highest similarity and/or the most neighbors in the training set. This result is intuitive in that we expect regressions to have the highest accuracy in domains nearest the training data. For our new approach to inverse QSAR, we measure accuracy of prediction using a metric that is calculated in terms of the Euclidean distance to the nearest molecule in the training set (details to follow). We first used this metric in the context of inverse QSAR for polymers.[20]

By defining solution quality as Euclidean distance to the nearest molecule in the training set, we can avoid enumerating the entire chemical space. We can instead enumerate the chemical space *near* our original data set. This new approach simultaneously eliminates three of the prohibitive difficulties of our original approach: we no longer compute the Hilbert basis, we no longer enumerate linear combinations of the Hilbert basis, and we no longer screen for accurately predicted solutions. In other words, we no longer enumerate millions of solutions only to discard most of them in the final steps. Instead, we enumerate only the solutions that we know will be accurately predicted.

In the remainder of this section, we provide details for our new approach to inverse QSAR. In terms of the steps 1−4 introduced in the previous section (Background): Steps 1 and 2 remain unchanged; step 3 contains all the major modifications; and step 4 is now entirely application specific, in that solutions are prescreened for prediction accuracy in step 3. A diagram highlighting the differences between the previous and new approaches is shown in Figure 4.

(3) Structure Enumeration. Our new approach to structure enumeration still solves the constraint equations described in the previous section (Background). Instead of computing the Hilbert basis, however, we now relax the inequality constraints and use lattice enumeration. To describe this approach precisely, we introduce some notation. Recall that signature space contains all possible signature vectors, but that chemical space contains only signature vectors that satisfy the constraint equations.

Suppose that we have n atomic signatures spanning the signature space, and that $\mathbf{x} \in \mathbb{Z}^n_{\geq 0}$ is a signature vector in chemical space. Suppose that there are m constraint equations, and that $A_{m\times n}$ contains the coefficients of the constraint equations, not including dummy variables. Since $\mathbf{x}$ is a vector in chemical space, we know that $\mathbf{x}$ satisfies the constraint equations so $A\mathbf{x} = [0, ..., 0, 2z_1 + c_1, ..., 2z_p + c_p, c_{p+1}, ..., c_q]^T$, with $z_1, ..., z_p \geq 0$. Let $\mathbf{b} = [0, ..., 0, 2z_1 + c_1, ..., 2z_p + c_p, c_{p+1}, ..., c_q]^T$. The zero entries of $\mathbf{b}$ correspond to the homogeneous constraint equations, the $z_1, ..., z_p$ are the dummy variables for the modulus constraint equations, and $c_1, ..., c_q$ are the constants corresponding to the nonhomogeneous equations. Note that we are defining p and q implicitly: if there are no modulus equations, then $p = 0$, and similarly a lack of nonhomogeneous equations would give $q = 0$.

dx.doi.org/10.1021/ci3001748 | J. Chem. Inf. Model. 2012, 52, 1787−1797

Now form an augmented coefficient $m \times (n + p)$ matrix

$$A^+ = \left[\begin{array}{c|c} & 0 \\ A & -2I_p \\ & 0 \end{array}\right]$$

where I_p is the $p \times p$ identity matrix. Let $\mathbf{x}^+$ be an augmented signature vector such that $\mathbf{x}^+ = [\mathbf{x}, z_1, ..., z_p]^T$. Note that $A^+\mathbf{x}^+ = [0, ..., 0, c_1, ... c_q]^T$ and let $\mathbf{b}^+ = [0, ..., 0, c_1, ... c_q]^T$. Finally, let $\{\mathbf{x}_1, ..., \mathbf{x}_k\}$ denote the signature vectors for each molecule in the QSAR training set, and let $\{\mathbf{x}_1^+, ..., \mathbf{x}_k^+\}$ denote the set of augmented vectors. Refer to the Experimental Results section for examples of the constraint equations and corresponding coefficient matrices.

Our goal is to enumerate vectors $\mathbf{x}$ in the chemical space such that $\|\mathbf{x} - \mathbf{x}_i\| \leq \varepsilon\|\mathbf{x}_i\|$ in Euclidean norm for a given $\varepsilon > 0$ and each $\mathbf{x}_i \in \{\mathbf{x}_1, ..., \mathbf{x}_k\}$. We perform this enumeration in three stages. First, we compute a basis N^+ spanning the null space $A^+\mathbf{x}^+ = 0$. This is done using a modulus basis and the Smith normal form. Second, we use the Fincke–Pohst algorithm[28] for lattice enumeration using the basis N^+ to obtain solutions $\mathbf{x}^+$ near the origin. Third, we translate the solutions $\mathbf{x}^+$ to each vector in the training set $\mathbf{x}_1^+, ..., \mathbf{x}_k^+$ (thus solving $A^+\mathbf{x}^+ = \mathbf{b}^+$) and screen the results to eliminate any solutions with negative coefficients. Using this approach, we eliminate the need to compute a Hilbert basis, instead using an algorithm with complexity controlled by a user provided parameter $\varepsilon > 0$.

Modulus Basis. In order to obtain a basis N^+ of the null space $A^+\mathbf{x}^+ = 0$, we must first compute a basis which eliminates the dummy variables $z_1, ..., z_p$ from $\mathbf{x}^+ = [\mathbf{x}, z_1, ..., \mathbf{z}_p]^T$. Denote by A_m the submatrix of A containing only the rows corresponding to the modulus and graphicality equations. We want a basis with matrix N_m that spans the null space of A_m modulo 2. That is, we want N_m such that any solution $\mathbf{x}$ of $A_m\mathbf{x} = 2I_p\mathbf{z} = 0 \bmod 2$ can be expressed as $\mathbf{x} = N_m\mathbf{u}$ for some $\mathbf{u} \in \mathbb{Z}^n$. This can be accomplished using reduced row echelon form (Gaussian elimination) of A_m then solving for $x_1, ..., x_p$ in terms of $x_{p+1}, ..., x_n, z_1, ..., z_p$. Note that this calculation can be done modulo 2 provided the dummy variables $z_1, ..., z_p$ are replaced by equivalent dummy variables $\tilde{z}_1, ... \tilde{z}_p$ as the elimination proceeds (in other words $A_m\mathbf{x} = 2I_p\mathbf{z}$ is equivalent modulo 2 to $A_m^r\mathbf{x} = 2I_p\tilde{\mathbf{z}}$, where A_m^r is the reduced row echelon form of A_m). Since there are p equations we are guaranteed $(n + p - p) = n$ free variables, each corresponding to a basis vector recorded as a column in N_m. Now $A_mN_m\mathbf{u} = 0 \bmod 2$, and $\mathbf{x} = N_m\mathbf{u}$.

Smith Normal Form. We next want a basis with matrix N^- for the null space of A^-, the submatrix of A *not* including the modulus or graphicality equations. This basis can be computed using the Smith normal form. Suppose $B_{m\times n}$ is a matrix with entries in $\mathbb{Z}$. The Smith normal form provides unimodular (invertible over $\mathbb{Z}$) matrices $L_{m\times m}$ and $R_{n\times n}$ such that $LBR = D_{m\times n} = \mathrm{diag}(d_1, ...d_s, 0, ..., 0)$, where $d_i \in \mathbb{Z}_{>0}$ and $d_i|d_{i+1}$. There are numerous algorithms for computing the Smith normal form. Generally these algorithms are polynomial time, with effort expended to avoid obtaining intermediate results involving very large integers.[43]

Using the Smith normal form, we can solve a linear system of Diophantine equations.[44] As an example, suppose $B\mathbf{x} = 0$. Since R is unimodular, there exists $\mathbf{y} \in \mathbb{Z}^n$ such that $R\mathbf{y} = \mathbf{x}$. Then $B\mathbf{x} = 0 \Leftrightarrow LB\mathbf{x} = L0 = 0 \Leftrightarrow LBR\mathbf{y} = 0 \Leftrightarrow D\mathbf{y} = 0 \Leftrightarrow \mathbf{y} = [0, ..., 0, y_1, ..., y_{n-s}]^T$, where $y_1, ..., y_{n-s}$ are free variables in corresponding to the last $n - s$ columns of R. Thus the solutions to $B\mathbf{x} = 0$ are given by integer linear combinations of the last $l = n - s$ columns of R. In our case, we would use the matrix A^-N_m to obtain R and D. The last columns of R give a basis with matrix N^- for the solutions to $A^-N_m\mathbf{u} = 0$. In other words, we get N^- such that $A^-N_mN^-\mathbf{v} = 0$, with $\mathbf{x} = N_mN^-\mathbf{v}$. In practice, we replace the null basis N_mN^- with an equivalent lattice basis known as the LLL basis, due to an algorithm of Lenstra, Lenstra, and Lovasz.[45] The LLL basis has properties better suited to lattice enumeration (short, nearly orthogonal vectors) and has been shown to improve the performance of the Fincke–Pohst algorithm.[28]

Fincke–Pohst Algorithm. The basis N_mN^- of solutions to $A^+\mathbf{x}^+ = 0$ defines a lattice in signature space. We enumerate points on this lattice using the Fincke–Pohst algorithm.[28] The Fincke–Pohst algorithm provides an enumeration of lattice points within an ε neighborhood of the origin. A lattice basis is given, and every lattice point within ε of the origin is returned. Suppose, for example, that the standard basis $\mathbf{e}_1, ..., \mathbf{e}_l$ were provided to the Fincke–Pohst algorithm, where $\mathbf{e}_i$ is the basis vector with 1 in the ith position and zeros elsewhere. In this situation, the Fincke–Pohst algorithm would perform a brute-force enumeration of every vector $\mathbf{x} \in \mathbb{Z}^l$ with $\|\mathbf{x}\| < \varepsilon$. In our case, however, many of these vectors would not satisfy the constraints. Instead, we use the lattice basis N_mN^- which guarantees that any lattice point $\mathbf{x}$ returned will satisfy $A^+\mathbf{x}^+ = 0$. We thus avoid enumerating vectors that will not solve the constraints.

The Fincke–Pohst algorithm is an exhaustive enumeration which uses the Cholesky factorization.[28] Because the Cholesky factorization gives upper triangular matrices, components of a lattice vector can be computed one at a time, thereby reducing redundant calculations. Using the Fincke–Pohst algorithm, the cost of computing a single lattice vector is $O(l^2)$, where l is then number of lattice basis vectors. Since the Fincke–Pohst algorithm enumerates lattice points within an ε neighborhood of the origin, the overall cost is at most $O(l^2\varepsilon^l)$. This is due to the fact that the number of lattice points within a ball is proportional to the volume of the ball, which is $O(\varepsilon^l)$ for a ball with radius ε. This is an exponential complexity, but it is controlled by the user selected parameter $\varepsilon > 0$. In addition, if $\varepsilon > 0$ is fixed while l varies, the algorithm can be expected to run in polynomial time.[28]

The Fincke–Pohst algorithm is particularly well-suited for our approach because it will enumerate lattice points within a ball, while other lattice enumerations algorithms use a box. This difference allows computations on lattices in much higher dimensions than competing algorithms,[28] important for applications in chemical space. In addition, a spherical distance is the preferred measure of QSAR predication accuracy.[27]

Solution Translation. In the final step, we translate the lattice solutions $\mathbf{x}$ to $A^+\mathbf{x}^+ = 0$ such that they produce signature vectors in chemical space near the QSAR training set $\{\mathbf{x}_1, ..., \mathbf{x}_k\}$. For a given $\varepsilon > 0$, our goal is to enumerate the set $S = U_iS_i$, where $S_i = \{\mathbf{x} \geq 0$: there exists $\mathbf{x}^+$ with $A^+\mathbf{x}^+ = \mathbf{b}^+$ and $\|\mathbf{x} - \mathbf{x}_i\| < \varepsilon\|\mathbf{x} - \mathbf{x}_i\|\}$. To enumerate S_i, we use the Fincke–Pohst algorithm with the null basis N_mN^- to enumerate the set $F_p = \{\mathbf{y}: A^+\mathbf{y}^+ = 0$ and $\|\mathbf{y}\| < \varepsilon \max_i\{\|\mathbf{x}_i\|\}\}$. Now $S^i = \{\mathbf{x}_i + \mathbf{y} \geq 0: A^+\mathbf{y}^+ = 0$ and $\|\mathbf{y}\| < \varepsilon\|\mathbf{x}_i\|\}$ is a translation and restriction of F_p (we denote by $\mathbf{x}_i + \mathbf{y} \geq 0$ the requirement that each component of $\mathbf{x} = \mathbf{x}_i + \mathbf{y}$ is non-negative).

Summary. Our new algorithm for structure enumeration has eliminated prohibitive computational difficulties from our original approach to inverse QSAR, while simultaneously improving the predicted accuracy of our solutions. By replacing

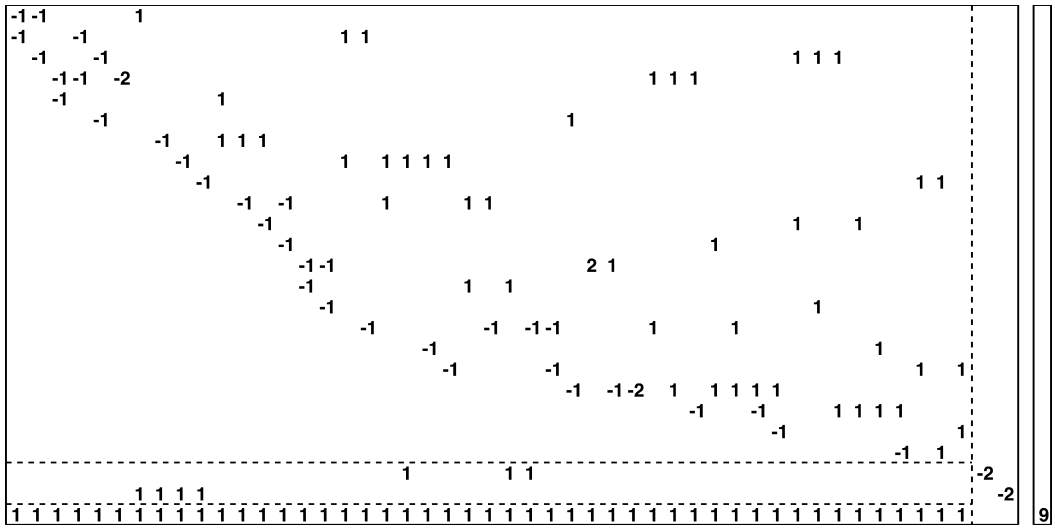


Figure 5. Constraint equations for the peptide rings. Here we show the augmented coefficient matrix A^+ on the left and the constant vector $\mathbf{b}^+$ on the right. The coefficient matrix A is to the left of the vertical dotted line. From top to bottom within A, we use dotted lines to separate the homogeneous equations, the modulus equations, and the nonhomogeneous equation. The matrix A_m corresponds to the modulus equations and occurs to the left of the submatrix $-2I_p$, which corresponds to dummy variables z_1, ..., z_p, where $p = 2$. Note that there is no graphicality equation.

the Hilbert basis computation with lattice enumeration, we can now control the computational effort we expend. This is a major improvement since the Hilbert basis solvers can provide neither estimated time to completion nor bounds on the number of basis vectors produced. In addition, the lattice enumeration completely eliminates the necessity of producing linear combinations of Hilbert basis vectors, another computationally prohibitive step. In practice, both the Hilbert basis solver and the computation of linear combinations were terminated early in applications.[20,22,23] Our new approach is more efficient because we avoid enumerating millions of solutions prior to discarding most of them in the final steps. We now enumerate *completely* (no early terminations) only the solutions likely to be useful, at the same time using much less effort than was previously required.

EXPERIMENTAL RESULTS

We benchmark our new approach using three previously studied applications: peptide rings,[21] hydrofluoroether foam blowing agents,[23] and polymers.[20] We focus on the structure enumeration step 3 for each application, comparing computational burden and resulting solutions for both the original and new approaches. Computational burden is measured by time on an iMac 2.7 GHz quad core i5 with 8 GB of RAM (all previous work was redone using the iMac). We also provide examples of the constraint equation coefficient matrices A, A^+. The examples are provided along with the code for performing the calculations at www.cs.otago.ac.nz/homepages/smartin.

Peptide Rings. The peptide ring data set was provided by researchers at the University of New Mexico (UNM) Health Sciences Center.[46,47] The peptide rings were developed to inhibit leukocyte trafficking and localization by mediating adhesion of leukocytes to the endothelium, specifically by inhibiting leukocyte functional antigen-1 (LFA-1) and its ligand intercellular adhesion molecule-1 (ICAM-1). The data set produced at UNM consisted of 16 peptide rings with 9 amino acids per ring, along with associated activity values measured experimentally using IC_{50}.

For this data set, we constructed signature vectors at the amino acid level, treating each amino acid as an atom in our molecular graph. We computed 47 height 1 amino acid signatures (essentially symmetric amino acid trimers) and from these signatures obtained 24 consistency equations and 1 nonhomogeneous constraint to ensure 9 amino acids per peptide ring.[21] Of the 24 consistency equations, there were 2 modulus equations. Since each amino acid signature was degree 2, the graphicality equation was always satisfied and was not included in our calculations. The coefficient matrices A and A^+ are shown in Figure 5, along with the submatrix A_m corresponding to the modulus equations.

Using our original approach, we considered 24 of the 25 constraints, excluding the nonhomogeneous equation.[21] We obtained 2222 Hilbert basis vectors from the Contejean–Devie solver in 1 min and 46 s. Next we computed linear combinations of basis vectors to obtain 223 solutions with 9 amino acids per ring in 2 h, 9 min, and 16 s. An ε value of 1.25 was calculated for these solutions.

Using our new approach to inverse design with $\varepsilon = 1.25$, we computed the same 223 solutions in 1/2 s.

Hydrofluoroether Foam Blowing Agents. Hydrofluoroethers (HFEs) have been suggested as a replacement for hydrochlorofluorocarbons (HCFCs) in insulating foam applications, due to the fact that HCFCs have ozone depletion potential.[48] Weis et al.[23] have previously assembled a data set of HFEs and performed inverse QSAR as described in the Background section to suggest novel HFEs. The data set consists of 76 HFEs and associated normal boiling points values (T_b). An intersecting set of 15 HFEs has associated vapor-phase thermal conductivity (λ_v) values. Inverse QSAR was performed on the HFEs, with screening performed on predicted T_b, λ_v, and various topological properties.[23] As a result, seven high-quality HFE solutions were suggested to replace the standard foam blowing agent HCFC R-141b.

For the HFE data set, 22 height 1 atomic signatures were produced and 5 constraint equations were obtained, including 3 homogeneous consistency equations, 1 modulus consistency equation, and 1 graphicality equation. The coefficient matrices

dx.doi.org/10.1021/ci3001748 | J. Chem. Inf. Model. 2012, 52, 1787–1797

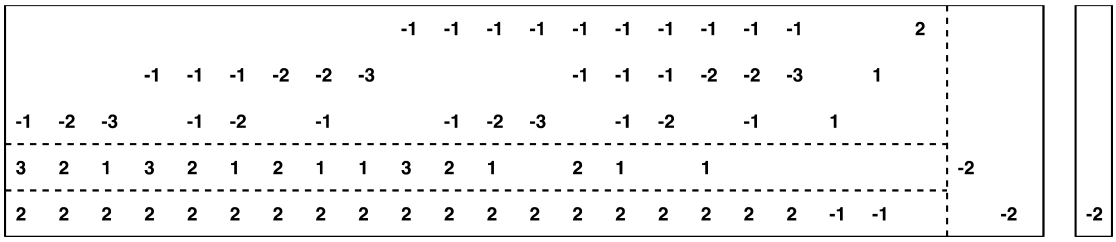


Figure 6. Constraint equations for HFE data set. The augmented coefficient matrix A^+ is on the left and the constant vector $\mathbf{b}^+$ is on the right. The coefficient matrix A is to the left of the vertical dotted line. From top to bottom within A, we use dotted lines to separate the homogeneous equations, the modulus equation, and the nonhomogeneous equation. The nonhomogeneous equation in this case is the graphicality equation, which is also a modulus equation. The matrix A_m therefore corresponds to the modulus equation and graphicality equation and occurs to the left of the submatrix $-2I_p$, which corresponds to dummy variables z_1, ..., z_p, where $p = 2$.

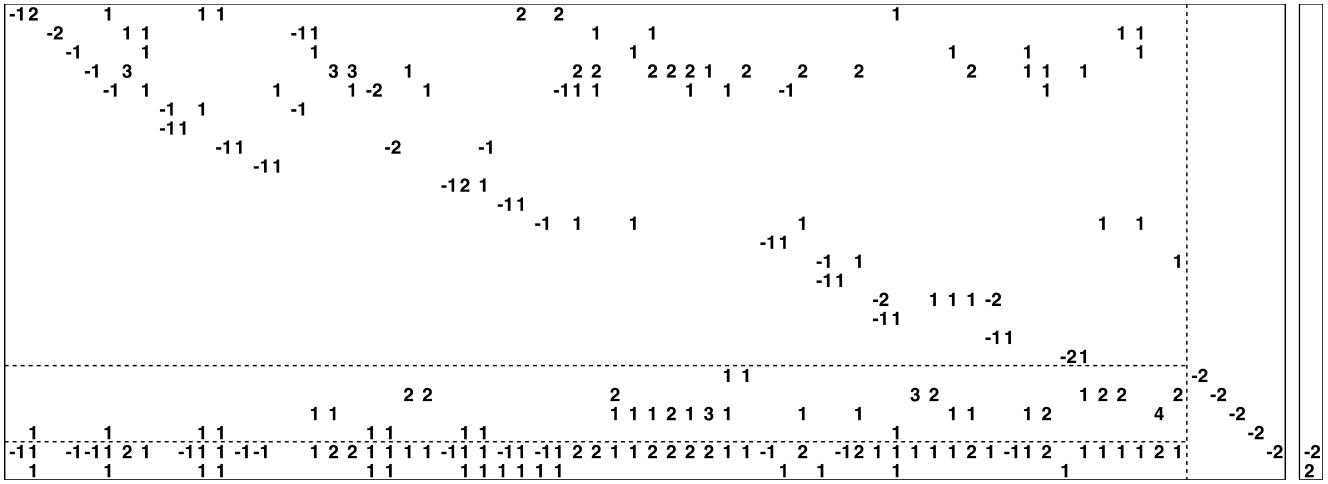


Figure 7. Constraint equations for linear homopolymers. The augmented coefficient matrix A^+ is on the left and the constant vector $\mathbf{b}^+$ is on the right. The coefficient matrix A is to the left of the vertical dotted line. From top to bottom within A, we use dotted lines to separate the homogeneous equations, the modulus equations, and the nonhomogeneous equations. The nonhomogeneous equations include both the graphicality equation and the polymer repeat equation (last row). Note that the graphicality equation is both nonhomogeneous and modulus. The matrix A_m corresponds to the modulus equations and graphicality equation and occurs to the left of the submatrix $-2I_p$, which corresponds to dummy variables z_1, ..., z_p, where $p = 5$.

A and A^+ are shown in Figure 6, along with the modulus equation submatrix A_m.

We revisited the inverse design work done by Weis et al. on the HFE data set.[23] We excluded the modulus and graphicality equations from the 5 constraints (leaving the 3 homogeneous equations) to obtain 64 Hilbert basis vectors in less than 1 s with the Contejean–Devie solver. Linear combinations of up to 5 basis vectors yielded 3 305 766 solutions with between 4 and 31 atoms in 35 min and 8 s. Those solutions were screened using two QSARs and some topological constraints (see Weis et al. for details[23]) to obtain 52 solutions. An ε value of 0.4 was calculated for these solutions. Under the original approach, we discovered 170 111 unique solutions in the $\varepsilon = 0.4$ neighborhood.

Using the new approach, we discovered 876 463 unique solutions (including all of the previously mentioned 170 111 solutions). The new approach took 19 min and 54 s.

Polymers. We have previously tested our original approach to inverse QSAR on the design of linear homopolymers.[20] For this problem, we collected 34 polymers for which 6 properties of interest were known, including molar volume, amorphous density, Fedors-type cohesive energy, van Krevlen-type cohesive energy, specific heat capacity, and glass transition temperature. QSARs were developed for each property and inverse QSAR was performing using the QSARs as screens along with the solution confidence metric suggested by Sheridan et al.[27] In particular, solutions were required to lie within a normalized Euclidean distance of 0.1 of the training set.

For the polymer data set, we obtained 63 atomic signatures of height 1. From these signatures, we computed 19 homogeneous consistency equations, 4 modulus consistency equations, and 1 graphicality equation. We also included 1 nonhomogeneous equation to ensure that the polymers would repeat. The coefficient matrices A and A^+ are shown in Figure 7, along with the modulus equation submatrix A_m.

We excluded the two nonhomogeneous equations in order to perform our original approach to structure enumeration. The remaining homogeneous constraints yielded 25 Hilbert basis vectors. Including the nonhomogeneous equations gave an additional 4707 particular solutions. We limited the combinations by allowing up to 3 basis vectors and solution component restrictions such that no component was larger than the corresponding component in any of the training set (see the work of Brown et al. for details[20]). These solutions were screened with the QSARs to obtain 79 high quality solutions. The entire process took 4 h, 25 min, and 4 s. Using the 79 high quality solutions, an ε value of 0.14 was calculated (note that our ε is defined differently from the normalized Eucliden distance used in the work of Brown et al.[20]). Within this ε

dx.doi.org/10.1021/ci3001748 | J. Chem. Inf. Model. 2012, 52, 1787–1797

neighborhood, there were 1074 of the 1327 solutions originally enumerated.

Using our new approach to structure enumeration with ε = 0.14, we identified 212 303 unique solutions to the constraints, including all of the 1074 solutions identified in the original approach. The enumeration took 1 h, 52 min, and 40 s.

Comparisons. It is instructive to compare the overall performance of our new approach against the original approach. Table 1 contains ε neighborhood sizes, timings, and number of solutions enumerated using both approaches on each of the three data sets considered. From Table 1, it is evident that the new approach is both faster and more thorough than the previous approach.

Table 1. Comparison of New and Original Approach on Three Data Sets[a]

		no. solutions		time (h:min:s)	
data set	ε	original method	new method	original method	new method
peptides	1.25	223	223	2:09:16	0:00:01
HFEs	0.4	170111	876463	0:35:08	0:19:54
polymers	0.14	1074	212303	4:25:04	1:52:40

[a]Performance measures for both methods on the three data sets are given in the last three rows of the table. Data sets include peptide rings, hydrofluoroether foam blowing agents (HFEs), and polymers. For each data set, we include the ε neighborhood used (second column), the number of solutions found using the original and new approaches (third and fourth columns), and the time required using the original and new approaches (fifth and sixth columns).

For the peptide rings, the new approach is many orders of magnitude faster, but for the HFEs and polymers it is only approximately twice as fast. The new method is faster because we are combining two steps from the original approach (basis enumeration and linear combinations) into a single lattice enumeration. However, the new method also limits the search space to an ε neighborhood around the training data. This is not a constraint in the original approach and we might in principle miss interesting solutions outside the ε neighborhood. In practice, however, these are the solutions less likely to be accurately predicted by the QSAR.[27]

The new method is also more *thorough* than the original approach. In the case of the peptides, both methods found the same 223 solutions. In the case of the HFEs and polymers, however, the new approach found significantly more solutions than the previous approach. In fact, the new method provides a guarantee that it will find *every* solution within the ε neighborhood. The original method cannot provide this guarantee unless every possible linear combination of Hilbert basis vectors is enumerated (this was only possible for peptide data set).

It is nevertheless surprising how many solutions were missed by the original approach. These solutions were missed due to the combinatorial infeasibility of combining Hilbert basis vectors to obtain solutions. In the case of the HFEs, it was unreasonable to go beyond combinations of five basis vectors to obtain solutions. As a result, over 700 000 solutions within ε = 0.4 of the training set were missed. In the case of the polymers, it was unreasonable to go beyond three basis vectors. For the polymers, we missed over 200 000 solutions within ε = 0.14 of the training set.

Selecting ε. There is a single parameter ε in our new approach which explicitly controls the number of solutions and implicitly affects the reliability of the resulting QSAR predictions. As ε increases, we obtain more solutions, but we also decrease the likelihood of reliable predictions. The question arises: how do we select ε? In practice, there are two main factors which influence our decision.

The first factor is computational. Although our new approach is an improvement of the previous method, it is still an exhaustive combinatorial enumeration, subject to the inherent computational difficulties of such an approach. Therefore, large values of ε will remain out of reach for our method. Nevertheless, our experiments indicate that huge numbers of solutions will still be accessible.

Depending on the application, we may obtain on the order of a million solutions. Further, the number of solutions will depend on the constraint equations. For the three applications previously considered (peptide rings, HFEs, and polymers), we obtained different solution distributions versus ε, as illustrated in Figure 8. In the case of the peptides, very restrictive constraints (exactly nine amino acids per ring) yielded relatively few solutions, with an apparent limit at 223. In the case of the HFEs, there were huge numbers of solutions, with an apparently subexponential growth as ε increased. For polymers, the growth of number of solutions appeared exponential with ε.

The second main factor in choice of ε is the result of our QSAR screens. Ultimately, the goal of an inverse QSAR method is to *reliably* suggest compounds with *desirable* properties. If for a given ε we have very few desirable solutions, we are forced to increase ε. In general, this will decrease the

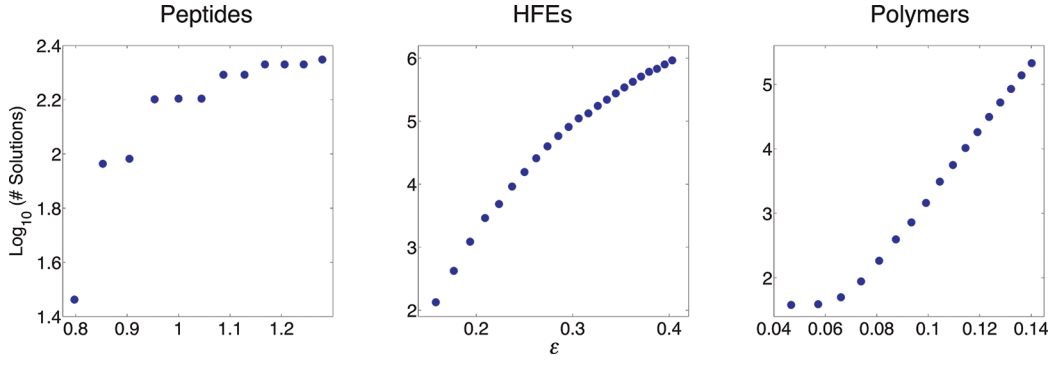


Figure 8. Distribution of solutions with ε. Here we show how the number of solutions obtained varies with the ε parameter. On the left, the number of solutions for the peptide ring application reaches a plateau around ε = 1.2. In the middle, the number of solutions for the HFE application appears to grow subexponentially with ε. On the right, the number of solutions for the polymer application grows exponentially.

dx.doi.org/10.1021/ci3001748 | J. Chem. Inf. Model. 2012, 52, 1787−1797

reliability of our QSAR predictions. On the other hand, we may be able to rapidly increase the number of solutions available for consideration, as indicated by our results on the HFEs and polymers (Figure 8). Eventually, of course, ε will become too large, in which case the constraints should be revisited in an effort to further restrict the chemical space. How the constraints might be altered will the subject of future work, as discussed in the Conclusions (next).

In summary, the best advice for choosing ε, both in terms of computation and chemistry, is to select the smallest ε that yields compounds with desirable properties. If suitable compounds cannot be identified, increase ε accordingly. If increasing ε becomes computationally infeasible, revisit the constraints to further restrict the chemical space.

CONCLUSIONS

In silico design of novel molecules with desired properties has the potential to greatly decrease both the time and cost of trial-and-error procedures. Unfortunately, deterministic enumeration for computer-aided molecular design is often dismissed due to the combinatorial complexity of the problem. Since the chemical space is in principle infinite, an exhaustive enumeration of all possible molecules is impossible. However, it is important to note that forward QSAR formulations are empirical. There is a limited domain under which the predictions are accurate, and therefore, only a subset of the chemical space should be considered for enumeration. This key observation, made previously in Brown et al.,[20] led us to develop an enumerative algorithm focused on identifying compounds in the subset of chemical space where QSAR predictions are likely to be accurate, namely the chemical space near the QSAR training set.

Our new approach supersedes our previous work,[20,21,23] replacing the computation of a Hilbert basis with lattice enumeration using the Fincke–Pohst algorithm. In this paper, we implemented and compared the new approach with the original approach on three previously studied data sets. We have demonstrated the superiority of the lattice enumeration method in both speed and accuracy. Indeed, a particular advantage of the new approach is an accuracy guarantee dependent on the user provided ε neighborhood of the training data. The ε parameter is an important improvement in the algorithm and provides the simultaneous ability to affect solution time, number of solutions requested, and QSAR accuracy.

Although the new approach improves the previous approach, there are can still be a huge number of potential molecules generated by the enumeration. Future work toward inverse design using the signature descriptor might therefore include methods for further reducing the number of candidate solutions. In addition to the QSAR screens already available, constraints could be added based on other graph theoretic conditions, such as increased signature height. As signature height increases, the solution set would approach the training set for a given ε. Other constraints might include application specific conditions, such as fragments that are required to be present in any solution. This would be beneficial in case a chemistry team was interested in modifying specific portions of a molecule. With such additional constraints, we could reduce the number of candidate solutions and allow enumeration of larger ε neighborhoods.

Finally, it is worth noting that the method proposed here is not specific to molecular design. It is in fact a generic method for enumerating certain solutions to high dimensional linear systems of Diophantine equations. Specifically, it is an alternative to the standard approach (via the Hilbert basis) in situations where there are a large number of variables. In addition to molecular design, such equations arise in various computer science applications such as Petri nets and associative-commutative unification.[49]

AUTHOR INFORMATION

Corresponding Author

*Phone: +64 3 479-7950. Fax: +64 3 479-8529. E-mail: smartin@cs.otago.ac.nz.

Notes

The authors declare no competing financial interest.

REFERENCES

(1) Winkler, D. A. The role of quantitative structure--activity relationships (QSAR) in biomolecular discovery. *Briefings Bioinf.* **2002**, *3* (1), 73–86.

(2) Reddy, A. S.; Pati, S. P.; Kumar, P. P.; Pradeep, H. N.; Sastry, G. N. Virtual screening in drug discovery -- a computational perspective. *Curr. Protein Pept. Sci.* **2007**, *8* (4), 329–51.

(3) Derringer, G. C.; Markham, R. L. A Computer-Based Methodology for Matching Polymer Structure with Required Properties. *J. Appl. Polym. Sci.* **1985**, *30*, 4609–4617.

(4) Brignole, E. A.; Bottlini, S.; Gani, R. A Strategy for Design and Selection of Solvents for Separation Processes. *Fluid Phase Equilib.* **1986**, *29*, 125–132.

(5) Gani, R.; Brignole, E. A. Molecular Design of Solvents for Liquid Extraction Based on UNIFAC. *Fluid Phase Equilib.* **1983**, *13*, 331–340.

(6) Gani, R.; Nielsen, B.; Fredenslund, A. A Group Contribution Approach to Computer-Aided Molecular Design. *AIChE J.* **1991**, *37* (9), 1318–1332.

(7) Kier, L. B.; Lowell, H. H.; Frazer, J. F. Design of Molecules from Quantitative Structure-Activity Relationship Models. 1. Information Transfer between Path and Vertex Degree Counts. *J. Chem. Inf. Comput. Sci.* **1993**, *33* (1), 143–147.

(8) Skvortsova, M. I.; Baskin, I. I.; Slovokhotova, O. L.; Palyulin, V. A.; Zefirov, N. S. Inverse Problem in QSAR/QSPR Studies for the Case of Topological Indices Characterizing Molecular Shape (Kier Indices). *J. Chem. Inf. Comput. Sci.* **1993**, *33* (4), 630–634.

(9) Fujiwara, H.; Wang, J.; Zhao, L.; Nagamochi, H.; Akutsu, T. Enumerating treelike chemical graphs with given path frequency. *J. Chem. Inf. Model.* **2008**, *48* (7), 1345–57.

(10) Churi, N.; Achenie, L. E. K. A Novel Mathematical Programming Model for Computer Aided Molecular Design. *Ind. Eng. Chem. Res.* **1996**, *35* (10), 3788–3794.

(11) Klein, J. A.; Wu, D. T.; Gani, R. In Computer-Aided Mixture Design with Specified Property Constraints. *European Symposium on Computer-Aided Process Engineering-ESCAPE-1*, Elsinore, Denmark, June 14–17; Gani, R., Ed. 1992; pp 229–236.

(12) Ostrovsky, G. M.; Achenie, L. E. K.; Sinha, M. A Reduced Dimension Branch-and-Bound Algorithm for Molecular Design. *Comput. Chem. Eng.* **2003**, *27* (4), 551–567.

(13) Raman, V. S.; Maranas, C. D. Optimization in Product Design with Properties Correlated with Topological Indices. *Comput. Chem. Eng.* **1998**, *22* (6), 747–763.

(14) Brown, N.; McKay, B.; Gilardoni, F.; Gasteiger, J. A graph-based genetic algorithm and its application to the multiobjective evolution of median molecules. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (3), 1079–1087.

(15) Douguet, D.; Thoreau, E.; Grassy, G. A genetic algorithm for the automated generation of small organic molecules: Drug design using an evolutionary algorithm. *J. Comput.-Aided Mol. Des.* **2000**, *14* (5), 449–466.

(16) Kvasnicka, V.; Pospichal, J. Simulated annealing construction of molecular graphs with required properties. *J. Chem. Inf. Comput. Sci.* **1996**, *36* (3), 516–526.

(17) Lin, B.; Chavali, S.; Camarda, K.; Miller, D. C. Computer-aided molecular design using Tabu search. *Comput. Chem. Eng.* **2005**, *29*, 337−347.
(18) Marcoulaki, E. C.; Kokossis, A. C. Molecular Design Synthesis Using Stochastic Optimization as a Tool for Scoping and Screening. *Comput. Chem. Eng.* **1998**, *22*, S11−18.
(19) Venkatasubramanian, V.; Chan, K.; Caruthers, J. M. Evolutionary Design of Molecules with Desired Properties Using the Genetic Algorithm. *J. Chem. Inf. Comput. Sci.* **1994**, *35*, 188−195.
(20) Brown, W. M.; Martin, S.; Rintoul, M. D.; Faulon, J. L. Designing novel polymers with targeted properties using the signature molecular descriptor. *J. Chem. Inf. Model.* **2006**, *46* (2), 826−35.
(21) Churchwell, C. J.; Rintoul, M. D.; Martin, S.; Visco, D. P., Jr.; Kotu, A.; Larson, R. S.; Sillerud, L. O.; Brown, D. C.; Faulon, J. L. The signature molecular descriptor. 3. Inverse-quantitative structure-activity relationship of ICAM-1 inhibitory peptides. *J. Mol. Graph Modell.* **2004**, *22* (4), 263−73.
(22) Jackson, J. D.; Weis, D. C.; Visco, D. P., Jr. Potential glucocorticoid receptor ligands with pulmonary selectivity using I-QSAR with the signature molecular descriptor. *Chem. Biol. Drug Des.* **2008**, *72* (6), 540−50.
(23) Weis, D. C.; Faulon, J.-L.; LeBorne, R. C.; Visco, D. P. The Signature Molecular Descriptor. 5. The Design of Hydrofluoroether Foam Blowing Agents Using Inverse-QSAR. *Ind. Eng. Chem. Res.* **2005**, *44* (23), 8883−8891.
(24) Helgee, E. A.; Carlsson, L.; Boyer, S. A method for automated molecular optimization applied to Ames mutagenicity data. *J. Chem. Inf. Model.* **2009**, *49* (11), 2559−63.
(25) Martin, S.; Brown, W. M.; Faulon, J.-L.; Weis, D.; Visco, D.; Kenneke, J. In Inverse Design of Large Molecules using Linear Diophantine Equations. *IEEE Computational Systems Bioinformatics (CSB) Workshop and Poster Abstracts*, Stanford, CA, Aug 8−11, 2005; IEEE: Stanford, CA, 2005; pp 11−14.
(26) Hawkins, D. M. The problem of overfitting. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (1), 1−12.
(27) Sheridan, R. P.; Feuston, B. P.; Maiorov, V. N.; Kearsley, S. K. Similarity to molecules in the training set is a good discriminator for prediction accuracy in QSAR. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (6), 1912−28.
(28) Fincke, U.; Pohst, M. Improved Methods for Calculating Vectors of Short Length in a Lattice, Including a Complexity Analysis. *Math. Comput.* **1985**, *44* (170), 463−471.
(29) Faulon, J. L.; Visco, D. P., Jr.; Pophale, R. S. The signature molecular descriptor. 1. Using extended valence sequences in QSAR and QSPR studies. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (3), 707−20.
(30) Todeschini, R.; Consonni, V. *Molecular Descriptors for Chemoinformatics*: Wiley-VCH: Weinheim, Germany, 2009; Vol. *41*.
(31) Faulon, J. L.; Misra, M.; Martin, S.; Sale, K.; Sapra, R. Genome scale enzyme-metabolite and drug-target interaction predictions using the signature molecular descriptor. *Bioinformatics* **2008**, *24* (2), 225−33.
(32) Weis, D. C.; Visco, D. P., Jr.; Faulon, J. L. Data mining PubChem using a support vector machine with the Signature molecular descriptor: classification of factor XIa inhibitors. *J. Mol. Graphics Modell.* **2008**, *27* (4), 466−75.
(33) Contejean, E.; Devie, H. An Efficient Incremental Algorithm for Solving Systems of Linear Diophantine Equations. *Inf. Comput.* **1994**, *113* (1), 143−172.
(34) Clausen, M.; Fortenbacher, A. Efficient Solution of Linear Diophantine Equations. *J. Symbolic Comput.* **1989**, *8* (1), 201−216.
(35) Hermann, M.; Juban, L.; Kolaitis, P. G., On the Complexity of Counting the Hilbert Basis of a Linear Diophantine System. In *Proceedings of the 6th International Conference on Logic Programming and Automated Reasoning*, Sept. 6−10; Springer-Verlag: Tbilisi, Georgia, 1999; Vol. *13*.
(36) Pasechnik, D. V. On Computing Hilbert Bases via the Eliot-MacMahon Algorithm. *Theor. Comput. Sci.* **2001**, *263* (1−2), 37−46.
(37) Faulon, J. L. On using graph-equivalent classes for the structure elucudation of large molecules. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 338−348.
(38) Faulon, J. L.; Churchwell, C. J.; Visco, D. P., Jr. The signature molecular descriptor. 2. Enumerating molecules from their extended valence sequences. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (3), 721−34.
(39) Colbourn, C. J.; Read, R. C. Orderly Algorithms for Generating Restricted Classes of Graphs. *J. Graph Theory* **1979**, *3*, 187−195.
(40) Balaban, A. Chemical Graphs: Looking Back and Glimpsing Ahead. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 339−350.
(41) Dimitrov, S.; Dimitrova, G.; Pavlov, T.; Dimitrova, N.; Patlewicz, G.; Niemela, J.; Mekenyan, O. A Stepwise Approach for Defining the Applicability Domain of SAR and QSAR Models. *J. Chem. Inf. Model.* **2005**, *45* (4), 839−849.
(42) Guha, R.; Jurs, P. C. Determining the validity of a QSAR model--a classification approach. *J. Chem. Inf. Model.* **2005**, *45* (1), 65−73.
(43) Storjohann, A. Near optimal algorithms for computing Smith normal forms of integer matrices. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation (ISAAC)*, July 24−26; ACM: Zurich, Switzerland, 1996; pp 267−274.
(44) Lazebnik, F. On Systems of Linear Diophantine Equations. *Math. Mag.* **1996**, *69* (4), 261−26.
(45) Lenstra, A. K.; Lenstra, H. W.; Lovasz, L. Factoring Polynomials with Rational Coefficients. *Math. Ann.* **1982**, *261*, 515−534.
(46) Shannon, J. P.; Silva, M. V.; Brown, D. C.; Larson, R. S. Novel cyclic peptide inhibits intercellular adhesion molecule-1-mediated cell aggregation. *J. Pept. Res.: Off. J. Am. Pept. Soc.* **2001**, *58* (2), 140−50.
(47) Sillerud, L. O.; Burks, E. J.; Brown, W. M.; Brown, D. C.; Larson, R. S. NMR solution structure of a potent cyclic nonapeptide inhibitor of ICAM-1-mediated leukocyte adhesion produced by homologous amino acid substitution. *J. Pept. Res.: Off. J. Am. Pept. Soc.* **2004**, *64* (4), 127−40.
(48) Sekiya, A.; Misaki, S. The potential of hydrofluoroethers to replace CFCs, HCFCs and PFCs. *J. Fluorine Chem.* **2000**, *101* (2), 215−221.
(49) Tomas, A. P. On solving linear Diophantine constraints. University of Porto, 1997.