Pacific Symposium on Biocomputing 5:585-596 (2000)

PICCOLO: A TOOL FOR COMBINATORIAL LIBRARY DESIGN VIA MULTICRITERION OPTIMIZATION

WEIFAN ZHENG, SUNNY T. HUNG, JOEL T. SAUNDERS, AND GEORGE L. SEIBEL

Cheminformatics Department, SmithKline Beecham Pharmaceuticals, UW 2940, 709 Swedeland Road, King of Prussia, PA 19406

Combinatorial library design is by nature a multicriterion problem. These criteria often include reagent diversity, product similarity to lead compounds and product novelty with respect to a corporate compound bank. More recently, developability and druglikeness have also attracted much attention in library design practices. To address this multicriterion design problem, we have developed a computer program (PICCOLO) that simultaneously optimizes all the factors under consideration using a weighted sum optimization technique. In this paper, we describe the overall design of this program and the formulation of individual penalty functions that characterize the underlying design criteria. We also give an example to illustrate the process and the result of a library design using this program.

1. Introduction

Combinatorial library synthesis has evolved from what was initially described as a shotgun approach to a recent one often based on computational planning. Early computational technologies of library design can be classified as one of two major types, diverse and targeted. In diverse design, one tries to select a subset of reagents that (a) are as representative as possible of the full reagent set, and (b) ensure the novelty of the resulting library with respect to the corporate compound bank. The former is the diversity analysis problem and the latter is the hole-filling problem. Many authors have employed clustering technologies,[1,2] direct optimization techniques[3,4,5,6] and cell-based approaches[7] to diversity analysis while the hole-filling problem has best been addressed by cell based methods. Several groups have also advocated pharmacophore diversity in compound selection.[8] Several reviews have appeared recently on the topic of library design and diversity analysis.[9,10,8,11]

In targeted design, on the other hand, one tries to select reagents so that a higher percentage of library compounds satisfies a predefined objective. The objective functions are often similarity of library compounds to lead molecules,[12,13] activities predicted using a pre-constructed QSAR model[14] or shape complementarity to a receptor binding site.[15]

The early computational techniques have separately addressed either diversity or probability for binding, and achieved their respective goals. Whether combinatorial libraries are being synthesized for lead finding or for lead optimization purposes, ultimately developable druglike compounds are being sought. Therefore, criteria

other than diversity and likelihood of binding should also be considered. Most notably, developability parameters are becoming more and more important in the practice of drug design.[16] Empirical rules regarding developability have been proposed with the most frequently cited being the "rules of 5" suggested by Lipinski.[17] A related topic, druglikeness, has attracted the interests of many and has led to several publications.[18,19] These factors and ultimately, compound solubility, membrane permeability and cytochrome P450 activities have to be considered together in comprehensive library design practice. Furthermore, when a mixture is being synthesized, mass spectroscopic redundancy becomes important for efficient deconvolution.

Since many factors need to be considered in combinatorial library design, we have taken a multicriterion optimization approach to this problem. Our method simultaneously optimizes important factors that include similarity to lead molecules, reagent diversity, product novelty, Lipinski parameters, mass redundancy and reagent prices. The list of factors is being expanded to include cytochrome P450 activity classifiers, permeability models, solubility predictors as well as other ADME models being developed in our group. Some authors have taken similar approaches to library design problems,[20,21,22] but to our knowledge, our program is the most comprehensive integrated tool that allows chemists to conduct their own computational experiments. We have named this program PICCOLO, which stands for reagent **PIC**king by **CO**mbinatorial **L**ibrary **O**ptimization.

In this paper, we first define the problem and the scope of this work and then discuss the overall design of this computational tool. We then detail the individual penalty functions. Finally, a simple example is given to illustrate the process as well as the result of a typical PICCOLO library design.

2. Problem Definition and Scope

Most combinatorial chemical libraries can be represented as a template and a set of R-groups (Ri, where i = 1 to *Nr*) and *Nr* is the number of substituents, which is usually between 1 and 4. The template may have up to *Nr* attachment points. The R-groups are usually attached to the template and can be bonded to each other. The template can be null, in which case the R-groups must be bonded to each other.

Let *Ni* be the number of reagents that are available for an R group Ri. Then the number of compounds that could be synthesized or enumerated is:

$$\prod_{i=1}^{Nr} N_i$$

For instance, one billion compounds could be synthesized in a combinatorial fashion when 1000 reagents are available for each reagent list in a three-R-group case. This is much more than one would like to synthesize. Rather, a smaller library is usually synthesized by selecting a subset of *Ki* reagents for each R group (*Ki* … *Ni*). Therefore, the problem becomes which *Ki* out of *Ni* reagents should be selected in order to get the best library. As mentioned in the first section, library design is a multi-objective optimization problem. This nature of library design entails the formulation of a goodness criterion (or a penalty function) that combines the penalty scores for all the objectives under consideration. This will be detailed in Section 4.

One can select {*Ki*} reagents and synthesize a library in a full combinatorial fashion, (i.e., each reagent is combined with every other) in which case a full combinatorial library design is needed. The {*Ki*} are usually specified according to the requirement of a particular project, but they can be determined using an algorithm to optimize the "library shape". Libraries can also be generated in a non-combinatorial fashion, which requires a partial combinatorial design or a "cherry picking" design. Here we limit the scope of this paper to address only full combinatorial library design when {*Ki*} are specified. The shape optimization and partial combinatorial optimization problems will be addressed elsewhere in subsequent papers.

3. Overall Design of the Algorithm

There are many ways of choosing *Ki* out of *Ni* reagents to make combinatorial libraries, and each of these ways is a potential solution to the library design problem. The number of all possible solutions for a full combinatorial library design is given by:

$$\prod_{i=1}^{i=Nr} \frac{Ni!}{Ki!(Ni-Ki)!}$$

That is the product of the numbers of all the combinations for choosing *Ki* from *Ni* over all the R groups. For instance, there are $1.54*10^{62}$ potential solutions when one is selecting 20 reagents out of 100 for every R group position for a 3-R-group library. The size of the solution space becomes even larger when more reagents are available. The discrete and combinatorial nature of the problem entails the use of an algorithm that can sample the solution space efficiently to find the global or near-global optimal solutions. Simulated annealing (SA)[23] is known to be such an

algorithm and has been employed in this work. Other stochastic algorithms such as genetic algorithms (GA)[24] and Taboo search[25] can also be applied.

Let *E(S)* represent the penalty score of a solution *S*. Let t_0, r_0, μ be the initial temperature, number of iterations in an annealing series, and temperature reducing factor, respectively. The general framework of the SA algorithm can then be described as follows.

1. Generate an initial solution (S_0) randomly and calculate $E(S_0)$.
2. Set $t = t_0$, $r = r_0$ and flag = FALSE. Set also current solution $S_c = S_0$, the best solution $S_b = S_0$, $E(S_c) = E(S_0)$, and $E(S_b) = E(S_0)$.
3. Generate a trial solution S_t by perturbing the current solution S_c. Calculate $\Delta E = E(S_t) - E(S_c)$. If $\Delta E \ldots 0$, then execute step 5; otherwise execute step 4.
4. Compute $P = \exp(-\Delta E/t)$ and compare it with a random value *y* from a uniform distribution in [0,1]. If y — P, then execute step 6; otherwise execute step 5.
5. Set $S_c = S_t$. If $E(S_t) < E(S_b)$, then set $S_b = S_t$. If $\Delta E < 0$, then set flag = TRUE.
6. Set r = r - 1. If r > 0, then return to step 3.
7. If flag is TRUE, then set flag = FALSE, $r = r_0$, $t = \mu$ t and repeat step 3; otherwise, stop with S_b as the best solution obtained.

Two of the most important aspects of our program are the formulation of the penalty function and the perturbation scheme used to generate the trial solutions during simulated annealing. These are explained in detail in the following section.

4. Computational Details

4.1. Virtual Library File (RG File)

We have adopted the RG file format[26] as the representation of virtual libraries for input to PICCOLO. This format allows the encoding of templates and R group members as connection tables, along with attachment point information. R group members may be attached to the template or to each other with one or more attachments. The template may be null leading to considerable representational freedom.

4.2. Perturbation method

We consider three aspects of the sampling task in this application. 1) The choice of which R group to be sampled on a given iteration; 2) The choice of reagent to be picked from the reagent pools; 3) The choice of reagent to be ejected from the current library solution. The first sampling question is addressed by considering the relative number of reagents used in each R group, as well as the size of the reagent pool for each R group. The R groups are sampled randomly with probability

determined by the average of the ratios of the size of a pool *Ni* to the total number of reagents in all pools and the number of selected reagents *Ki* to the total number of selected reagents. This empirical rule biases the sampling toward the R groups that need more sampling, while still ensuring that each R group is sampled reasonably. The second sampling decision is handled by a uniform random sampling approach. The reagent pool is randomized at the start of the optimization, with reagents selected in order. After the pool has been fully sampled, the sampling begins again from the start of the list. This sampling method is more efficient than a purely random approach, and converges faster. Finally, the reagent that is ejected from the chosen R group of the library is selected purely at random.

4.3. Enumeration of Molecules for a Given Solution

Molecular structures of library compounds need to be enumerated to calculate each individual penalty score for a solution. For the initial solution (S_0) in a SA optimization, all molecules have to be enumerated and their properties calculated. However, not all compounds of a perturbed solution are different from those of the previous one. To improve the computational efficiency, molecular features and other properties of all the molecules of a solution are stored in an internal data structure. When a trial solution (S_t) is being generated from a current solution (S_c), all the features and other parameters of S_c are copied into S_t. When S_t is then perturbed (Cf. Section 4.2), only the structures of new compounds are enumerated and their parameters are calculated.

4.4. Objective Function

Our current objective function contains terms related to Diversity, Developability, Focussing, and Practicality. There are two diversity terms: The first, Reagent Diversity, describes the degree of self-similarity among the reagents at each position. The second term, Product Novelty, depends on the similarity between the products of the library and our existing large collection of compounds. Developability terms include molecular weight, lipophilicity, and hydrogen bond donor/acceptor counts. Focussing in this version of PICCOLO is implemented by computing an average similarity between the library and one or more leads. Other terms related primarily to practical issues include Mass Spectral Redundancy, reagent price, and product flexibility. The overall objective function *E(S)* of a solution is defined as the weighted sum of penalty scores $E_i(S)$ for all the terms under consideration. That is,

$$E(S) = \sum w_i * E_i(S)$$

Where w_i is the weight given to the *i*th term. Each term is described below.

Reagent diversity A simple method is employed to penalize the selected library for excessive self-similarity. An S-optimality criterion is computed for the reagents of each R group based on the Daylight fingerprint[27] Tanimoto distances between members of that R group. This has the effect of minimizing the maximum similarity between members of an R group. A uranium atom is connected to each R group member at the point at which it attaches to the template. Fingerprints are computed with the uranium atom included in order to encode the position of attachment.

Product novelty An important goal of many library designs is to augment our large screening collection with compounds that populate previously unexplored regions of chemical space. In order to avoid the time-consuming pairwise comparison of library products with hundreds of thousands of compounds at each iteration of the optimization, we chose to implement a low dimensional cell-based method. In this approach, compounds from the screening collection are represented in a 4 to 6 dimensional feature space, mapped onto a grid with 20 divisions on each axis. The cell occupancies are stored in memory. On each iteration, features are calculated for each new product, its location on the grid is determined in an extremely rapid lookup, and the cell count at that location is incremented. The average cell count for the library is minimized. We have implemented a smoothing function that "bleeds" density from occupied cells into adjacent cells, as a way of partially compensating for the errors introduced by the discrete nature of the cells. The results reported here used a 4D principal components space derived from topological indices calculated with Molconn-Z.[30] More recently we have implemented a 6D feature space optimized against the simultaneous separation of actives from inactives in 20 high throughput screens.

Developability penalties In his well known 1997 paper[17], Lipinski pointed out the importance to the developability of a compound of molecular weight, logP, and hydrogen bond donor / acceptor counts. He also introduced a convenient mnemonic known as the "Rule of 5", which states that compounds associated with good developability properties have MW less than 500, logP less than 5, and no more than 5 donors or 10 acceptors. We took these four terms as our initial set of developability parameters, in each case taking the term to be minimized as the fraction of the total number of molecules in the library that fall outside of the limit for each term. Lipinski's values for each term are used as defaults, but all are variables under the control of the chemist. We have also incorporated a lower bound for logP, not present in Lipinski's rules but having a well-known significance to permeability.[28] The default lower limit is a logP of –1. LogP is calculated with the

Daylight / BioByte clogP.[29] More recently we have implemented neural network classifier modules with more sophisticated developability models.

Similarity to leads One or more lead molecules may be used as a focussing target. Similarity metrics include either Daylight fingerprint Tanimoto similarity or the Euclidean distance in a principle component space derived from topological indices computed using Molconn-Z.[30] The penalty score for each compound in the library is defined as the distance between it and the nearest lead molecule. The penalty score for the library is the average of the individual compound penalty scores. We have also incorporated neural network classifiers for focussing as well as developability biasing.

Mass Spectral Redundancy A library synthesized using a split and mix protocol without the use of tagging schemes results in a mixture of polymer beads in which only the identity of the last R group added is known. When a compound cleaved from one of these beads is found to be active in an assay, it is subjected to mass spectroscopic analysis in which the parent ion mass is determined. All expected products from the library that fall in the range of the mass of the molecular ion +/- the resolution of the instrument must be resynthesized for confirmation. It is therefore beneficial to minimize mass spectral redundancies in the library. Our algorithm takes into account the different combinations of chlorine and bromine atoms that can be identified by characteristic isotopic mass patterns. It is parameterized with the resolution of the mass spectrometer. The library is divided into sublibraries according to the last substituent added in the synthesis. Products are sorted by mass and redundancies are counted. The average number of redundancies per library is used as the penalty function.

Reagent Price The price of reagents, when available, is taken from the Available Chemicals Directory (ACD). Although extremely expensive reagents can be filtered out prior to optimization, we find price optimization beneficial. The quantity minimized is the average reagent price in dollars per equivalent.

5. Experiments

5.1. The Reaction Scheme and Reagent Lists

A published synthetic scheme[31] (Figure 1) for a four component Ugi reaction has been adopted as a design example in this paper. Since two of the four components are fixed in the scheme, only two diversity sites remain for optimization. These two sites come from primary amines (R_1NH2) and aldehydes (R_2CHO), respectively. We have collected from ACD (Available Chemical Directory) structures of primary amines and aldehydes available from ALDRICH and LANCASTER. Following our normal practice, we have removed compounds with synthetically incompatible functional groups. For example, compounds with multiple amino groups, multiple aldehyde groups, amines with {-CHO and/or -NC groups}, and aldehydes with {-NH2 and/or -NC groups} have been removed. Compounds with other reactive or unstable structural patterns were also removed. As a result, 779 primary amines and 246 aldehydes were considered and their structures were put into an R-group file for the PICCOLO optimization.

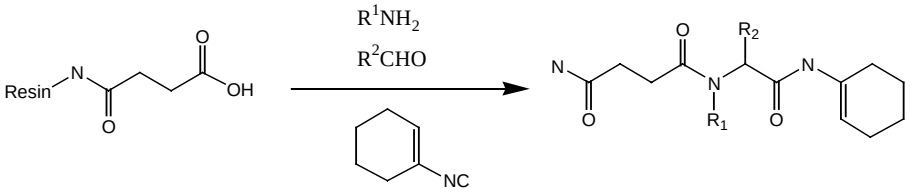

Figure 1. Reaction scheme for a Ugi library.

5.2. An 864-member Ugi Library

To demonstrate what parameter setup and the results look like in PICCOLO, an 864-member library is being designed with 24 primary amines (R1) and 36 aldehydes (R2). Table 1 shows the setup of a typical experiment. The default ranges for developability parameters are set according to Lipinski. The default weights have been selected based on the ranges of values observed in most cases.

Table 1. Parameter setup for a PICCOLO experiment.

Developability			
Terms	Lower limit	Upper limit	Weights
MW	0	500	1.0
ClogP	-1	5	1.0
H-bond donor	0	5	0.3
H-bond acceptor	0	10	0.3
Diversity & Hole-filling			
Reagent diversity	N/A	N/A	40.00
Hole-filling	N/A	N/A	0.01
Potential for Binding			
Similarity	N/A	N/A	0.0
Practicality			
Mass Spec	N/A	N/A	0.0
Price	N/A	N/A	0.0

6. Results and Discussions

PICCOLO records all the accepted solutions during the Simulated Annealing process. These solutions and their associated penalty scores are presented in a Spreadsheet. For the 864-member Ugi library, a spreadsheet showing the best three solutions and the initial random solution are given in Table 2. Note that the terms with zero weights except MS redundancy are not shown in Table 2. One can see that molecular weight and clogP penalties went from 70% and 39% for the initial solution down to 1.9% and 0.9% for the best solution. Hydrogen bond donor and acceptor counts penalties went from 0.4% and 28% down to 0.0% and 10.0%, respectively. Reagent diversity (Sdiv) penalty also went down from 73 to 2.1. Since we gave a very small weight to hole-filling (Hfil) and zero weight to mass spectral redundancy (MS) in this experiment, their penalty scores actually went up. This indicates that we can emphasize those terms that we care about most by giving larger weights to them and sacrifice those that we do not care by giving smaller weights. This example also indicates that we can reduce the penalty for multiple terms simultaneously. Chemists usually select several of the best-scoring solutions for further examination.

PICCOLO also displays the trajectory of each penalty score during the optimization process. Figure 2 shows an example of the trajectories for Lipinski penalty scores (MW, H-bond acceptor counts, and clogP).

Table 2. Spreadsheet in a PICCOLO run.

Solution #	Tot	Sdiv	Hfil	MW	HBD	HBA	LogP	MS
2004 (best)	91.9	2.1	334	1.9	0.0	10.0	0.9	0.34
1951	92.1	2.1	353	1.85	0.0	10.0	0.9	0.34
1847	92.13	2.1	359	1.85	0.0	10.0	0.9	0.34
…								
1 (initial)	3047	73.0	126	70	0.4	28	39	0.08

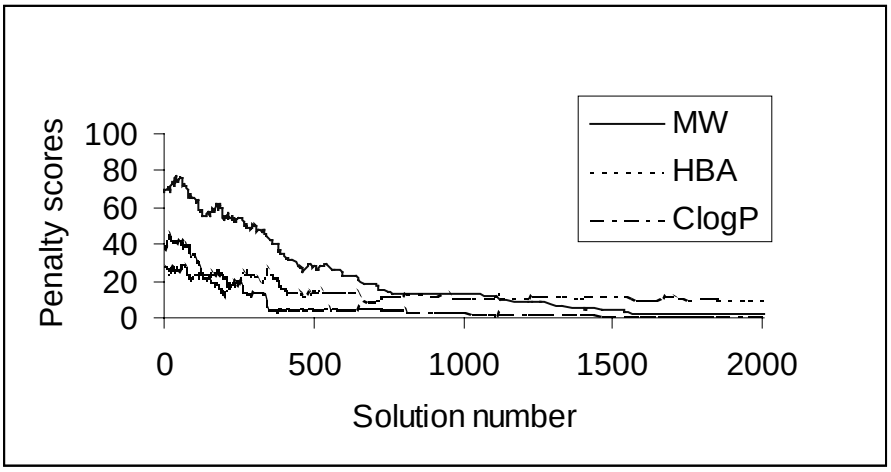

Figure 2. SA trajectories for MW, H-bond acceptor and clogP scores.

Finally, PICCOLO allows users to view the structures, prices, and vendor information for the reagents of any design solution. Users can observe the MW/clogP distribution of a solution as a scatter plot, with individual points hyperlinked to product structures (data not shown). All the aforementioned functionalities are accessible to users via a Web-based user interface.

One can use PICCOLO to compare two different library shapes and find out which design is better. For instance, a second 864-member Ugi library was designed using 36 R1 (primary amines) and 24 R2 (aldehydes). Compared with the 24-by-36 library, it has a better diversity (1.9 for 36-by-24 vs. 2.1 for 24-by-36), a better hole-filling (254 vs. 334), a similar clogP score (0.69 vs. 0.93) and a better H-bond acceptor score (8.0 vs. 10.0). It is, however, worse on MW (8.3 vs. 1.9). Chemists are encouraged to experiment in this manner.

One can bias the library towards any developability criterion. For instance, when a larger weight was given to clogP when designing a 36-by-24 Ugi library, a solution with 0.0% penalty on clogP was obtained, as opposed to 0.69 in the previous case. This is not a dramatic change due to the nature of this particular library, but it can make a huge difference in other situations. This solution is also better on MW (2.6 as opposed to 8.3), better on H-bond acceptor penalty (3.8 vs. 8.0). As expected, the diversity of the library (2.28 vs. 1.92) was sacrificed to achieve the above goals.

7. Conclusions and Future Work

We have developed an integrated computer program (PICCOLO) for library design that simultaneously optimizes all the factors under consideration. This program is accessible transnationally to all chemists at SmithKline Beecham. It has been used in a large number of library design problems of both the general screening and targeted variety. We encourage chemists to run "computational experiments" to test their design ideas.

The program has been designed so that additional objective functions can be added when more focusing criteria become available. Of particular importance are classifiers/predictors for developability parameters such as compound solubility, membrane permeability, cytochrome P450 activities and other ADME parameters. Pharnacophore models and receptor site directed docking scores could also be incorporated.

The perturbation scheme has a significant impact on the speed of convergence and the quality of the results. Thus, the study of various perturbation schemes is of both theoretical and practical interest. We have considered one of them in this paper (see Section 4.2). The way that a reagent is removed from a current solution could conceivably be based on a calculated "quality index" for each reagent. This and other variants of our perturbation schemes will be addressed elsewhere.

8. Acknowledgements

We thank Dr. Kenneth Kopple for his support and encouragement, and Drs. Jie Liang, Kenneth Kopple and Stephen Johnson for their useful discussions and proofreading the manuscript. We also thank Drs. Jian Jin, Todd Graybill, and Dennis Yamashita for their valuable input.

9. References

1. Shemetulskis N.E.; Dunbar J. B. Jr; Dunbar B. W.; Moreland D. W.; Humblet C. *J. Comput. Aided Mol. Des.* **1995**, 9(5), 407-416.
2. Brown R. D.; Martin Y.C. *SAR QSAR Environ. Res.* **1998**, 8(1-2), 23-39.

3. Agrafiotis, D. K. *J. Chem. Inf. Comput. Sci.*, **1997**, 37(5), 841.
4. Hassan M.; Bielawski, J. P.; Hempel, J. C.; Waldman, M. *Mol. Divers.* **1996**, 2(1-2), 64-74.
5. Chapman, D. *J. Comput. Aided Mol. Des.* **1996**, 10(6), 501-12.
6. Zheng, W.; Cho, J.; Waller, C.L.; and Tropsha, A. *J. Chem. Inf. Comput. Sci*., **1999**, 39(4).
7. Pearlman, R.S.; Smith, K.M. *J. Chem. Inf. Comput. Sci*.,**1999**, 39(1), 28.
8. Mason J.S.; Hermsmeier, M. A. *Curr. Opin. Chem. Biol*., **1999**. 3(3), 342-9.
9. Blaney, J. M.; Martin, E. J. *Curr. Opin. Chem. Biol*., **1997**, 1(1), 54-9.
10. Bures M. G.; Martin, Y. C. *Curr. Opin. Chem. Biol.* **1998**, 2(3), 376-80.
11. Agrafiotis, D. K.; Myslik J. C.; Salemme F. R. *Mol Divers.* **1998-99**, 4(1), 1-22.
12. Sheridan, R. P.; Kearsley, S. K. *J. Chem. Inf. Comput. Sci.*, **1995**, 35(2), 310.
13. Zheng W.; Cho S. J.; Tropsha, A. *J. Chem. Inf. Comput. Sci.* **1998**, 38(2), 251-8.
14. Cho S. J.; Zheng, W.; Tropsha, A. *J. Chem. Inf. Comput. Sci.*, **1998**, 38(2), 259-68.
15. Sun Y.; Ewing T. J.; Skillman A. G.; Kuntz I.D. *J. Comput. Aided Mol. Des.* **1998**, 12(6), 597-604.
16. Tarbit, M. H.; Berman, J. *Curr. Opin. Chem. Bio.l* **1998**, 2(3), 411-6.
17. Lipinski C. A.; Lombardo, F.; Dominy, B. W.; Feeney, P. J. *Advanced Drug Delivery Reviews* **1997**, 23, 3-25.
18. Ajay, A.; Walters, W. P.; Murcko, M. A. *J. Med. Chem.* **1998**,, 41(18), 3314-24.
19. Sadowski, J.; Kubinyi, H. A. *J. Med. Chem.* **1998**, 41(18), 3325-9.
20. Gillet, V. J.; Willett, P.; Bradshaw, J.; and Green, D. V. S. *J. Chem, Inf. Comput. Sci*. **1999**, 39, 169-177.
21. Good A. C.; Lewis, R. A. *J. Med. Chem.* **1997**, 40(24), 3926-36.
22. Ghose, A. K.;Viswanadhan, V. N.; and Wendoloski, J. J *J. Comb. Chem.* **1999**, 1(1), 55-68.
23. Kirkpatrick, S.; Gelatt, C. D. Jr.; Vecchi, M. P. *Science* **1983**, 220, 671-680.
24. Forrest, S. *Science* **1993**, 261, 872-878.
25. Cvijovic, D.; Klinowski, J. *Science* **1995**, 267, 664.
26. CT File Formats, August 1998, MDL Information Systems, Inc., 14600 Catalina Street, San Leandro, CA 94577.
27. Daylight Chemical Information Software, version 4.51, Daylight Chemical Information Systems, Inc., 27401 Los Altos, Suite #370, Mission Viejo, CA 92691.
28. Camenisch, G.; Folkers, G.; Waterbeemd, van de H. *Eur. J. Pharm. Sci.* **1998**, 6(4), 325-333.
29. Daylight clogP, v4.51, Daylight Chemical Information Systems, Inc., 27401 Los Altos, Suite #370, Mission Viejo, CA 92691.
30. MolConn-Z, version 3.10; eduSoft, P.O.Box 1811, Ashland, VA 23005.
31. 4 Component Ugi Reaction. Advanced ChemTech Handbook of Combinatorial & Solid Phase Organic Chemistry. Advanced ChemTech. P.65.