2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

A Deterministic Annealing Approach to Combinatorial Library Design for Drug Discovery

Puneet Sharma, Srinivasa Salapaka and Carolyn Beck

Abstract— **In this paper we propose a new algorithm for the design of a lead generation library for the purpose of drug discovery. The algorithm is based on a modification of the deterministic annealing algorithm used for clustering and locational optimization problems. The algorithm addresses two key criteria namely, diversity and representativeness of compounds in the obtained library. Thus the time-intensive process of finding a lead compound for drug discovery can be accelerated using this technique of library design.**

I. INTRODUCTION

Combinatorial libraries consist of extremely large collections of chemical compounds, which when tested on target or test cells (say cancer cells) identify required properties, such as structural or bioaffinity properties of potential drugs for new treatments. Recent advances in high throughput screening such as micro/nanoarrays [4] have enabled the large scale investigation of compounds for drug discovery. These advances have made it possible to explore interactions between *individual* molecules - this specificity with which the interactions can be studied promises discovery of drugs with similar level of specificity. However with the current technology, it is impossible to test the various interactions between all the potential bio-molecules as the combinatorial nature of the problem leads to an unmanageable number of pairs to be tested.

To address this problem, tools from combinatorial optimization have been employed to design libraries consisting of subsets of representative compounds which can be synthesized and subsequently tested for relevant properties, such as structural activity, bioaffinity, aqueous solubility etc. In order to increase the overall efficiency of the drug discovery process, these libraries should be designed such that they are *representative* of all the potential compounds and at the same time contain a manageable number of compounds for testing.

More specifically, the selection problem in drug discovery, that is the process of selecting a *representative* subset of chemical compounds that also *covers* the possible range of solutions, is a form of resource allocation problem often referred to as a *locational optimization* problem. Locational optimization algorithms arise in a number of contexts in control; for example, motion coordination algorithms, coverage control [12] and mobile sensing networks [3]. These problems share the fundamental goals of aiming to determine an optimal partition of the underlying domain in which they are defined (e.g., a library of compounds for drug discovery, an unknown area of interest for coverage control), and an optimal assignment of values, or elements, from a finite set to each *cell* in the partition space.

These problems are typically computationally complex and time intensive. For example, choosing 30 representative compounds from an array of 1000 compounds will result in approximately 3×10^{25} possibilities. Another factor which adds to the complexity of such problems is their inherent non-convex nature. Thus we require an efficient selection algorithm that does not get stuck in the local minima.

Although all the aforementioned locational optimization problems share similar basic optimization goals, there are a number of features and criteria which are specific to each problem and thus distinguish one from the other. These differentiating characteristics include different distance metrics in the definition of *coverage*, the number and types of constraints on the resources in the problem formulation, the necessity of computing global versus local optima, the possibility of elements and resources that exhibit dynamical behavior, and the size and scale of the feasible domain. For example, in the drug discovery problem, there are scenarios where we have capacity constraints on the experimental resources, thus leading to a *multi-capacity constraint problem* similar to one which arises in the optimal strategic positioning of UAVs [12]. Motion control problems are satisfactorily solved by local optimization solutions, i.e., *distributed* coordination algorithms utilizing only nearest-neighbor information are more relevant and are preferred to global coordination schemes, whereas in the drug discovery scenario we are primarily interested in global solutions.

In this paper, we propose algorithms based on the concept of *deterministic annealing (DA)* to cater to the specific constraints and demands of combinatorial chemistry. The main distinctive feature of the DA algorithm is that it aims at avoiding local minima. At the same time, it is faster than the simulated annealing algorithm [1], [7]. Taking into account the huge size of combinatorial libraries, it becomes necessary to consider the scaling issues involved with the DA algorithm. We are currently in the process of studying numerical issues of the DA algorithm for large libraries.

This paper is organized as follows. In Section II, we provide background information on some of the specifics of combinatorial library design, and state the main problem we consider. The underlying approach we employ for the solution of the selection problem, i.e. the DA algorithm, is

This work is partially supported by AFOSR URI F49620-01-1-0365

The first and third authors are with the Coordinated Science Lab and the Department of General Engineering and the second author is with the Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign

Emails: psharma2@uiuc.edu, salapaka@uiuc.edu, beck3@uiuc.edu

0-7803-9098-9/05/$25.00 ©2005 AACC

described in Section III. We then discuss modifications we have made to the DA algorithms for our specific focus in Section IV. In Section V we present a number of simulated data set results. Finally we conclude the paper by recapping the important results and discussing some ongoing research for addressing the scaling issues of the DA algorithm.

II. LIBRARY DESIGN

Library design refers to the process of screening and then selecting a subset of compounds from a vast given array of similar or distinct compounds for the purpose of drug discovery [6]. The main aim of library design is to reduce the number of compounds for testing without compromising on the diversity of the library. Optimal design of libraries (containing diverse compounds) facilitates the process of drug discovery by replacing the testing of all compounds by a much smaller set of representative compounds.

Based on the current state of development in the drug discovery process, library design can be broadly classified into two main categories, namely lead generation and lead optimization. The main purpose of these libraries and their design criteria are discussed below.

A. Lead Generation Library

The development of a lead generation library usually involves the design and synthesis of a large number of chemical compounds. It is required that these compounds are varied and hence diverse from each other. The library containing these diverse compounds is then tested against a host of different biological agents. The main objective in designing such libraries is to obtain structurally diverse compounds so as to cover the chemical space efficiently. Keeping these requirements in mind, 'diversity' is generally used as the principal screening criterion for designing lead generation libraries. This criterion does not necessarily give intended results, and may encourage library design that contains compounds so diverse, i.e. singletons, which are not 'representative' of any group other than themselves. Hence the criterion of 'representativeness' should also be considered along with 'diversity'. This issue is addressed in this paper by modifying the deterministic annealing algorithm in such a way as to allow us to measure the extent of representativeness of the prospective leads.

B. Lead Optimization Library

Lead optimization libraries are usually designed at a later stage of the drug discovery process when it is required to select a subset of compounds that are 'similar' to a given lead compound(s). This results in a array of compounds which are structurally and chemically 'similar' to the lead. This criterion of similarity is generally used for designing targeted or focussed libraries, which mostly focus on a single therapeutic target. Thus the design of lead generation library precedes that of the lead optimization library.

This paper deals with the problem of designing a library of compounds for the purpose of lead generation. The most common method used to obtain such a library is to maximize the diversity of the overall library. It is based on the strategy that the more diverse the set of compounds, the better the chance to obtain a lead compound with desired characteristics. As was noted earlier, such a design strategy suffers from an inherent problem which occurs due to the fact that using diversity as the only criterion may result in a set of compounds which are 'exceptions' or singletons. Fig. 1 shows such a scenario. The X's denote the compounds chosen according to the maximum diversity principle. As can be seen from the figure, the cluster in the middle is not adequately represented in the final selection. This algorithm focusses more on distant compounds, which may be viewed as exceptions. An algorithm using such a criterion gives equal importance to the middle cluster and the exception compounds.

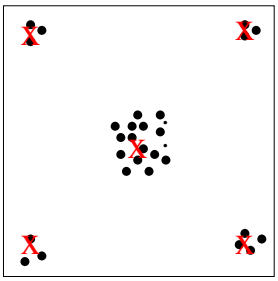

Fig. 1. A scenario depciting the inherent problems with the 'diversity' only criterion for lead generation library design

From a drug discovery point of view, it is desirable for the lead generation library to contain more compounds from the middle cluster (so as to adequately represent all the compounds) or at least know how representative they are in order to be able to make decisions on how much experimental resources to devote to these lead compounds. To address this problem, we propose a modification to the DA algorithm for selecting an optimal subset of compounds from the given combinatorial array. This modification leads to an optimization problem where the objective function tackles both the issues of diversity and representativeness effectively. The process involves identifying different compound locations in an iterative fashion, and is discussed in the following sections. The iterations are carried out as dictated by the DA algorithm.

In order to quantify the different criteria for designing libraries (namely 'diversity' and 'similarity'), it is required to define appropriate molecular descriptors which define the various compounds. Its has been previously shown that a 2-d molecular descriptor space exhibits a proper 'neighbourhood behaviour' which is essential for characterizing 'similarity' and 'diversity' properties between two compounds [1]. In all our simulations, we consider a 2-d molecular descriptor space in which the all compounds reside. The Euclidean distance between two points in the space provides a good approximation of the degree of 'similarity' or 'diversity' between these two compounds. Thus in this scenario of a 2-d descriptor space, the close

neighbours of an active compound will also be active. On the other hand, two compounds which are far apart (in terms of the Euclidean distance) can be labeled as diverse. The 2-d descriptor space provides a means of quantifying the properties of different compounds.

In addition to 'similarity' and 'diversity', other criteria can also be used so that the set of compounds satisfy particular design objectives. 'Confinement' is often used as a criterion to quantify the degree to which the properties of a set of compounds lie between prescribed upper and lower ranges [2]. Another objective is maximizing the 'activity' of the set of compounds against some predefined targets. This 'activity' is usually measured in terms of the quantitative structure of the given set. The presence of these multiple (and often conflicting) design objectives make the library design problem a multi-objective optimization problem.

III. Deterministic Annealing Algorithm

In its prototypical form, the problem of selecting representative elements for the purpose of library design can be stated as:

Given a distribution $p(x)$ of the elements x in a descriptor space $\mathcal{D}$, find the best *set of M representative elements r_j that solves the following minimization problem:*

$$\min_{r_j,\ 1\le j\le M} \int_{\mathcal{D}} p(x) \left\{ \min_{1\le j\le M} d(x, r_j) \right\} dx. \quad (1)$$

Here $d(x, r_j)$ represents an appropriate *distance* metric between the representative element r_j and the element x. Alternatively, this problem can also be formulated as finding an *optimal* partition of the descriptor space $\mathcal{D}$ into M cells R_j and assigning to each cell R_j a representative element r_j such that the following cost function is minimized

$$\sum_j \int_{R_j} d(x, r_j) p(x) dx.$$

Realistic objective functions have unpredictable surfaces with many local minima, and thus require design algorithms that avoid them. The DA algorithm is suited for this purpose since it is specifically designed to avoid local minima.

This algorithm can be viewed as a modification of another algorithm called Lloyd's algorithm [8], [5]. Lloyd's algorithm is an iterative method which identifies two necessary conditions of the optimal solution and then ensures that at each iteration, the partition of domain and the representative elements satisfy these conditions:

1) Nearest Neighbor condition (Voronoi partitions): The partition of the domain is such that each element in the domain is associated to the nearest representative element.
2) Centroid condition: The representative elements are such that the location r_j is in centroid of the jth cell R_j.

In this algorithm, the initial step consists of randomly choosing locations of representative elements and then successively iterating between the steps: (1) forming Voronoi partitions, and (2) moving the representative elements to respective centroids of cells till the sequence of locations of representative elements converge. It should be noted that the solution depends substantially on the initial allocation as in the successive iterations the locations are influenced only by 'near' points of the domain and are virtually independent of 'far' points. As a result the solutions from this algorithm 'typically' get stuck to local minima.

The DA algorithm [11], [10] does away with this local influence of domain elements by allowing each element $x \in \mathcal{D}$ to be associated with every representative element r_j through a weighting parameter $p(r_j|x)$. Thus this algorithm does away with the hard partitions of Lloyd's Algorithm. The DA formulation includes a modified distortion term

$$D = \int_{\mathcal{D}} p(x) \sum_j d(x, r_j) p(r_j|x) dx,$$

which is similar to the cost function in Equation 1. It also includes an entropy term

$$H = -\int_{\mathcal{D}} p(x) \sum_j p(r_j|x) \log p(r_j|x) dx,$$

which measures the randomness of distribution of the associated weights. This entropy is the highest when the distribution of weights over each representative element is the same ($p(r_j|x) = 1/M$) for each x, i.e., when all x have the same influence over every representative element. This algorithm solves the following optimization problem

$$\min_{r_j} \min_{p(r_j|x)} \underbrace{D - T_k H}_{:=F}$$

at the kth iteration where T_k is a parameter called *temperature* which tends to zero as k tends to infinity. The cost function F is called *Free Energy* as this formulation has a close parallel in statistical physics [9]. Clearly for large values of T_k, we mainly attempt to maximize the entropy. As T_k is lowered we trade entropy for the reduction in distortion, and as T_k approaches zero, we minimize D directly to obtain a hard (non random) solution. Minimizing the Free Energy term F with respect to the association probabilities $p(r_j|x)$ is straightforward and gives the *Gibbs* distribution

$$p(r_j|x) = \frac{e^{-d(x,r_j)/T}}{Z}, \quad \text{where} \quad (2)$$

$$Z := \sum_i e^{-d(x,r_i)/T} \quad (3)$$

is called the *partition function*. The corresponding minimum of F is obtained by substituting for $p(r_j|x)$ from Equation 2,

$$\hat{F} = -T \int_{\mathcal{D}} p(x) \log Z. \quad (4)$$

To minimize $\hat{F}$ with respect to the representative elements $\{r_j\}$, we set the corresponding gradients equal to zero i.e.,

($\frac{\partial \hat{F}}{\partial r_j} = 0$); this yields the corresponding implicit equations for the locations of representative elements

$$r_j = \int_{\mathcal{D}} p(x|r_j)x dx \quad 1 \leq j \leq M \tag{5}$$

where

$$p(x|r_j) = \frac{p(x)p(r_j|x)}{\int_{\mathcal{D}} p(x')p(r_j|x')dx'}. \tag{6}$$

Note that $p(x|r_j)$ denotes the posterior probability calculated using Bayes's rule and the above equations clearly convey the 'centroid' aspect of the solution.

The DA algorithm consists of minimizing $\hat{F}$ with respect to $\{r_j\}$ starting at high values of T_k and tracking its minimum while lowering T_k. The steps at each k are

1) fix $\{r_j\}$ and use Equation 2 to compute the new weights $\{p(r_j|x)\}$.
2) fix $\{p(r_j|x)\}$ and Equation 5 to compute the representative elements $\{r_j\}$.

IV. Modified Deterministic Annealing Algorithm

In the original DA formulation, different resources are indistinguishable since each of them carry equal weight. However, there are situations where the capacity constraints distinguish one resource from another. In order to account for such capacity constraints, it is necessary to modify the DA algorithm. For the specific problem of library design, such a scenario occurs when we want to address the issue of 'representativeness' of individual clusters in the final library design. In order to constrain the size of each cluster, it becomes necessary to distinguish between the various locations of representative elements. Moreover, constraints on the experimental resources also call for a modification of the DA algorithm. In this section, we present two modifications of the original DA algorithm for tackling the issues faced in library design.

A. Incorporating the representativeness criterion

We present a modified version of the DA algorithm for a capacity constrained problem. This modification is necessary in order to incorporate the criterion of 'representativeness' in the algorithm. As discussed in Section II, such a criterion when used along with 'diversity' effectively addresses the problem of avoiding 'singletons' in the library.

This can be achieved by choosing multiple compounds at each representative element location. The multiplicity λ_j at each location of the representative element r_j can be regarded as number of compounds that should be chosen from that location. According to the nature of the problem, the capacity (W_j) at each location of representative element can be constrained. Taking this constraint into account, we get the following constrained minimization problem:

$$\min_{r_j, 1\leq j\leq M} \int_{\mathcal{D}} p(x) \left\{ \min_{1\leq j\leq M} d(x, r_j) \right\} dx$$

such that

$$\lambda_j = W_j \quad 1 \leq j \leq M$$

Note here that a large value of λ_j implies that compounds near the location r_j need to be given more weight than those at other locations. It should also be noted that although the constraints do not seem to occur in the cost function explicitly, they can be interpreted as the multiplicity (λ_j) at each location. In terms of the DA algorithm, this can be thought of as multiple units of resources at each location. Taking this into account, the partition function in Equation 3 can be rewritten as

$$Z := \sum_i \lambda_i e^{-d(x,r_i)/T}$$

This leads to a modified Gibbs distribution and Free Energy term. Following a procedure similar to that in the previous section, we finally get

$$p(r_j|x) = \frac{\lambda_j e^{-d(x,r_j)/T_k}}{\sum_i \lambda_i e^{-d(x,r_i)/T_k}}$$

This modified algorithm effectively addresses the issue of 'diversity' and 'representativeness' at the same time. The value of W_j can be interpreted as the relative importance of the clusters which we want to choose in the lead generation library. Thus using this algorithm will not result in the scenario portrayed by Fig.1. This algorithm gives us an option for choosing the amount of 'diversity' and 'representativeness' that we want in a particular library.

B. Incorporating experimental resources constraints

In this section, we provide further modification to the DA algorithm in order to account for constraints on experimental resources. In this scenario, we classify the library into q types corresponding to experimental resources required by the compounds for testing, i.e., the nth type of compound requires the nth resource for testing. The algorithm modification described in this section addresses the fact that there are, necessarily, constraints on each of these resources in the experimental process. This translates into direct constraints on each of the representative elements, for example, the jth representative element can avail only W_{jn} amount of the nth resource. This type of a constraint is generally referred to as multi-capacity constraint. [12].

The modified optimization problem is given by

$$\min \; D = \sum_n \int_{\mathcal{D}} p_n(x^n) \sum_{j=1}^{M} d(x^n, r_j) p(r_j|x^n) dx^n$$

such that

$$\lambda_{jn} = W_{jn} \quad 1 \leq j \leq M, \;\; 1 \leq n \leq q$$

where $p_n(x^n)$ is the probability distribution of the compound corresponding to location x^n, needing nth type of resource and W_{jn} is the amount of nth resource that the jth representative element can avail.

We proceed along the same lines as the DA algorithm by defining the entropy term as

$$H = -\sum_n \int_{\mathcal{D}} p_n(x^n) \sum_{j=1}^{M} p(r_j|x^n) log p(r_j|x^n) dx^n$$

and minimizing the Free Energy given by $F = D - T_k H$.

After some computations, we obtain the corresponding Gibbs distribution as

$$p(r_j|x^n) = \frac{\lambda_{jn} e^{-d(x^n, r_j)/T_k}}{\sum_i \lambda_{in} e^{-d(x^n, r_i)/T_k}}$$

Adding the constraints to this equation, we derive the new Lagrangian given by

$$F^{'} = -1/T \sum_n \int_{\mathcal{D}} \log(\sum_j \lambda_{jn} e^{-d(x^n, r_j)/T}) p_n(x^n) dx^n +$$

$$\sum_j \sum_n q_{jn}(\lambda_{jn} - W_{jn}),$$

where $q_{jn} 1 \leq n \leq q, 1 \leq j \leq M$ are Lagrange multipliers. Finally, the optimal location of representative elements is obtained by setting $\frac{\partial F^{'}}{\partial r_j} = 0$. This gives the following set of equations

$$r_j = \frac{\sum_n p_n(x^n) p(r_j|x^n) x^n dx^n}{\sum_n p_n(x^n) p(r_j|x^n) dx^n}$$

where $p(r_j|x^n)$ has been calculated as above.

V. SIMULATIONS AND RESULTS

A. Simulated Data Sets

For the purpose of simulation, the DA algorithm was tested on two different types of data sets. The first set was obtained as follows. 15 random locations were identified in a square region of size 200×200. These locations were then chosen as the cluster locations. Next, the size of each of these clusters was chosen and all points in the cluster were generated by a normal distribution of randomly chosen variance. A total of 450 points comprised this data set. Choosing 10 points from this data set results in a total of 8.4×10^{19} different possibilities.

The second data set was specifically designed to address the problem of 'diversity' and 'representativeness' in the lead generation library. In this set, cluster locations were chosen so that a minority of the clusters were quite 'diverse', with the majority of cluster locations chosen close together in a given region.

B. Simulation Results

Simulations were carried out (in MATLAB) on both data sets described above. The results for data set 1 are shown in Fig. 2. The DA algorithm is a hierarchical algorithm, and thus it identifies natural clusters in the population at each step. As the 'temperature' is decreased, the algorithm

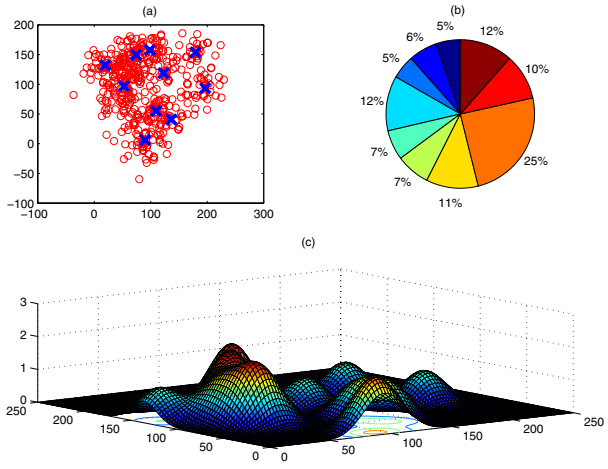

Fig. 2. Simulation results for data-set 1; (a) The locations of different compounds and representative elements in the 2-d descriptor space. (b) The weights associated with different locations of representative elements. (c) The probability distribution of different compounds.

identifies the various clusters and sub-clusters in a hierarchical fashion. The pie chart in Fig. 2 shows the relative weight that should be assigned to each representative element. Representative elements with a large weight signify that more compounds are chosen from that area. As was required, the algorithm gave higher weights at locations which had larger numbers of similar compounds. Thus different relative weights at each resource location address the issue of 'representativeness' in the library. At the same time, it should be noted that the key issue of 'diversity' is not at all compromised in any sense. This is due to the fact that the algorithm inherently recognizes the 'natural' clusters in the population. Thus a 'unique' compound (one which is maximally diverse from all the others) will be identified , but with a very small weight. This issue will be further dealt with in the results for data set 2. Hence the DA algorithm chooses a set of compounds which are diverse (from each other) as well as representative for the entire library. Fig. 3 shows the results for data set 2. The main idea behind choosing such a data set was to demonstrate the fact that 'diversity' alone cannot be used as an objective for the design of lead generation libraries. As is seen from the figure, the DA algorithm identifies all cluster locations. The two cluster locations which were quite diverse from the rest of the compounds are also identified albeit with a smaller weight. As can be seen from the accompanying pie chart in Fig. 3, the exception cluster was assigned a weight of 2%, while the central lump was assigned a significant weight of 22%.

Another significant feature of the modified DA algorithm is the flexibility it provides in dealing with 'unique' (i.e. far-away) clusters. By properly assigning the relative importance of clusters apriori, we can choose whether to include or reject these 'unique' clusters in the library design. Though immediate rejection of 'unique' clusters compromises the diversity of the library, there can be scenarios

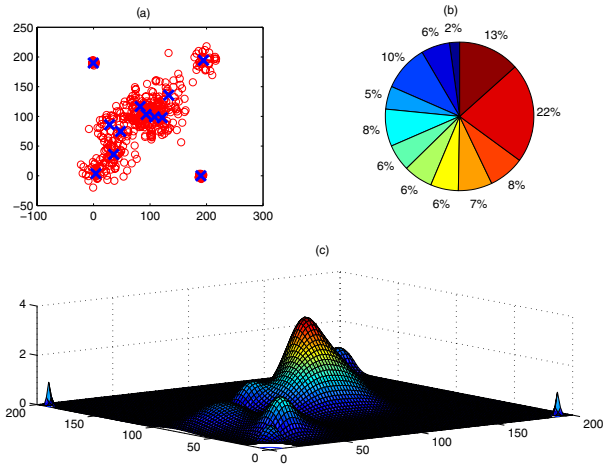

Fig. 3. Simulation results for data-set 2; (a) The locations of different compounds and representative elements in the 2-d descriptor space. (b) The weights associated with different locations of representative elements. (c) The probability distribution of different compounds.

where the properties of these 'unique' cluster compounds are totally undesired in the lead generation library. Thus the modified DA approach gives us a means to deal with such scenarios effectively.

Fig. 4 shows the results obtained by using the modified algorithm for the case where constraints were imposed due to the different experimental resources. The compounds were classified into three types (denoted by 'green', 'red' and 'pink' dots in Fig. 4). The algorithm was used to select a total of 15 representative element locations. The 'blue' crosses denote the locations chosen by the algorithm. This

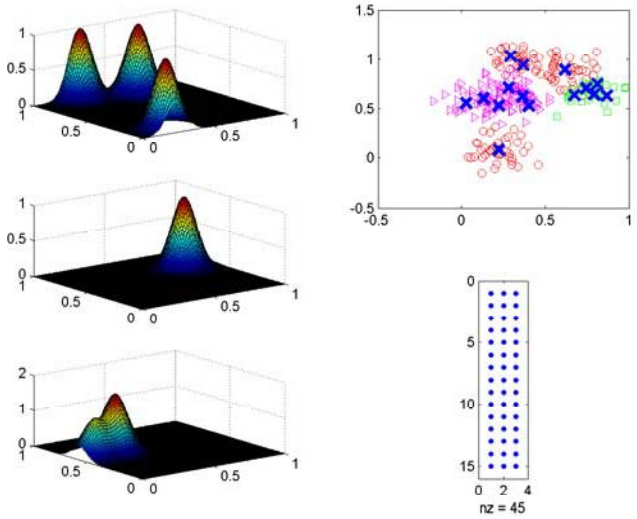

Fig. 4. Simulation results (constraints due to experimental resources). (a) Probability distributions of different types of compounds (based on experimental resources required). (b) The locations of different compounds and representative elements. (c) The capacities of each experimental resource to different types.

modified DA algorithm was used for many test cases. The essential feature of this algorithm is that it converges to the global optimum each time, even for complex scenarios.

VI. CONCLUSIONS

In this paper, we have presented a new algorithm for the design of a lead generation library for drug discovery, based on a modified version of the DA algorithm. The modified algorithm enables us to effectively and simultaneously address the key issues of 'diversity' and 'representativeness' in the chosen lead generation library. At the same time, the algorithm does not get stuck in local optima. Simulation results have been presented for different capacity constraints using three different simulated data sets.

Due to the inherently large size of present day combinatorial libraries, the scalability of this algorithm needs to be studied further. Since the present algorithm operates in a 2-d space, modifications are being evaluated for incorporating higher dimensions. The metric used for calculating distance between two points can be modified to include additional criteria in the optimization problem.

The DA algorithm effectively addresses the issue of locating 'inherent' clusters in a given population. The next step would be to identify the sub-clusters within inherent clusters and further, requiring a modification in the basic annealing algorithm so that it works in a recursive fashion. The issue of recursive clustering using the present DA algorithm is being considered.

The next step in drug discovery is usually the design of a lead optimization library, in which the main criterion is that of 'similarity' (to the lead compounds). The existing DA algorithm can be further modified to address this problem effectively. This issue is also being considered by the authors at present.

REFERENCES

[1] D.K Agrafiotis. Multiobjective optimization of combinatorial libraries. *IBM Journal of Research and Development*, 45(3):545–566, 2001.

[2] D.K. Agrafiotis and V. S. Lobanov. Ultrafast algorithm for designing focussed combinatorial arrays. *Journal of Chemical Information and Computer Sciences*, 40:1030–1038, 2000.

[3] J. Cortés, S. Martínez, T. Karatas, and F. Bullo. Coverage Control for Mobile Sensing Networks: Variations on a Theme. *Proc. Mediterranean Conference on Control and Automation*, July 2002.

[4] Linette Demers and Guy della Cioppa. Drug discovery: Nanotechnology to advance discovery R&D. 23(15), September 2003.

[5] A. Gersho and R. Gray. *Vector Quantization and Signal Compression*. Kluwer, Boston, Massachusetts, 1st edition, 1991.

[6] E.M. Gordon, R.W. Barrett, W.J. Dower, S.P.A. Fodor, and M.A. Gallop. Applications of combinatorial technologies to drug discovery. 2. combinatorial organic synthesis, library screening strategies, and future directions. *Journal of Med. Chem.*, 37(10):1385–1401, 1994.

[7] S. Kirkpatrick, C.D. Gelatt, and M.P. Vechhi. Optimization by simulated annealing. *Science*, 220:671–680, 1983.

[8] S.P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[9] K. Rose. Statistical mechanics and phase transitions in clustering. *Physics Review Letters*, 65(8):945–948, 1990.

[10] K. Rose. Constrained Clustering as an Optimization Method. *IEEE Trans. Pattern Anal. Machine Intell.*, 15:785–794, August 1993.

[11] K. Rose. Deterministic Annealing for Clustering, Compression, Classification, Regression and Related Optimization Problems. *Proceedings of the IEEE*, 86(11):2210–39, November 1998.

[12] S. Salapaka and A. Khalak. Constraints on Locational Optimization Problems. *Proc. IEEE Control and Decision Conference*, pages 1741–1746, December 2003.