

Analyzing Probabilistic Models in Hierarchical BOA on Traps and Spin Glasses

Mark Hauschild

Missouri Estimation of Distribution Algorithms
Laboratory (MEDAL)
Dept. of Math and Computer Science, 320 CCB
University of Missouri – St. Louis
One University Blvd., St. Louis MO 63121
mwh308@admiral.umsl.edu

Martin Pelikan

Missouri Estimation of Distribution Algorithms
Laboratory (MEDAL)
Dept. of Math and Computer Science, 320 CCB
University of Missouri – St. Louis
One University Blvd., St. Louis MO 63121
pelikan@cs.umsl.edu

Claudio F. Lima

Informatics Laboratory (UALG-ILAB)
Department of Electronics and Computer
Science Engineering
University of Algarve
Campus de Gambelas, 8000-117 Faro, Portugal
clima@ualg.pt

Kumara Sastry

Illinois Genetic Algorithms Laboratory (ILLiGAL)
Department of Industrial and Enterprise Systems
Engineering
University of Illinois at Urbana-Champaign
ksastry@uiuc.edu

ABSTRACT

The hierarchical Bayesian optimization algorithm (hBOA) can solve nearly decomposable and hierarchical problems of bounded difficulty in a robust and scalable manner by building and sampling probabilistic models of promising solutions. This paper analyzes probabilistic models in hBOA on two common test problems: concatenated traps and 2D Ising spin glasses with periodic boundary conditions. We argue that although Bayesian networks with local structures can encode complex probability distributions, analyzing these models in hBOA is relatively straightforward and the results of such analyses may provide practitioners with useful information about their problems. The results show that the probabilistic models in hBOA closely correspond to the structure of the underlying problem, the models do not change significantly in subsequent iterations of BOA, and creating adequate probabilistic models by hand is not straightforward even with complete knowledge of the optimization problem.

Categories and Subject Descriptors

I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search; I.2.6 [Artificial Intelligence]: Learning; G.1.6 [Numerical Analysis]: Optimization

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO '07, July 7–11, 2007, London, England, United Kingdom.
Copyright 2007 ACM 978-1-59593-697-4/07/0007 ...\$5.00.

General Terms

Algorithms, Performance

Keywords

Bayesian optimization algorithm, hierarchical BOA, estimation of distribution algorithms, model accuracy

1. INTRODUCTION

The hierarchical Bayesian optimization algorithm (hBOA) [22, 21, 25] can solve the broad class of nearly decomposable and hierarchical problems in a robust and scalable manner and a number of efficiency enhancement techniques have been proposed to further improve hBOA performance [35, 29, 16, 32, 36]. While both the performance of hBOA as well as the effectiveness of various efficiency enhancement techniques for hBOA crucially depend on the quality of probabilistic models used in hBOA to guide exploration of the space of potential solutions, very little work has been done in analyzing the structure and complexity of these models with respect to the structure of the underlying optimization problem [16, 40, 15].

The purpose of this paper is to analyze the structure and complexity of probabilistic models in hBOA on two common test problems: (1) concatenated traps and (2) two-dimensional $\pm J$ Ising spin glasses with periodic boundary conditions. Concatenated traps are used as an example of a separable problem of bounded order with subproblems that cannot be further decomposed [8, 1]. Two-dimensional Ising spin glasses, on the other hand, cannot be decomposed into subproblems of bounded order [18] and they provide a challenge for most optimization methods because they contain an extremely large number of local optima and the candidate solutions between different local optima are often of very low quality [3, 17, 9, 7, 28]. The results show that (1) probabilistic models obtained by hBOA closely correspond to the structure of the underlying optimization problem, (2) the

models do not change significantly between subsequent iterations of hBOA, and (3) creating adequate probabilistic models by hand is far from straightforward even with complete knowledge of the structure of the underlying problem. The proposed techniques for model analysis in hBOA can be adapted to study model structure and model complexity for other problems and other estimation of distribution algorithms, providing the practitioners with a powerful tool for learning about their problems. The obtained knowledge can be used to extend the class of problems that can currently be solved tractably by hBOA.

The paper is organized as follows. Section 2 outlines hBOA. Section 3 discusses concatenated traps and trap-5 in particular, and then analyzes probabilistic models obtained with hBOA when solving trap-5. Section 4 describes the problem of finding ground states of 2D $\pm J$ Ising spin glasses with periodic boundary conditions, and presents the analysis of hBOA models in this class of problems. Finally, section 5 concludes the paper and outlines future research.

2. HIERARCHICAL BOA (HBOA)

Estimation of distribution algorithms (EDAs) [2, 19, 14, 27]—also called probabilistic model-building genetic algorithms (PMBGAs) [27, 21] and iterated density estimation algorithms (IDEAs) [5]—replace standard crossover and mutation operators of genetic and evolutionary algorithms by building a probabilistic model of selected solutions and sampling the built model to generate new candidate solutions. The hierarchical Bayesian optimization algorithm (hBOA) [22, 24, 21] is an EDA that uses Bayesian networks to represent the probabilistic model and incorporates restricted tournament replacement [12] for effective diversity maintenance. This section outlines hBOA and briefly discusses Bayesian networks, which are used to guide the exploration of the search space in hBOA.

2.1 Basic hBOA Procedure

hBOA evolves a population of candidate solutions. The population is initially generated at random according to a uniform distribution over all n -bit strings. Each iteration starts by selecting a population of promising solutions using any common selection method of genetic and evolutionary algorithms, such as tournament and truncation selection. We use truncation selection with the threshold $\tau = 50\%$. New solutions are generated by building a Bayesian network with decision trees [6, 10] for the selected solutions and sampling the built Bayesian network. The new candidate solutions are incorporated into the original population using restricted tournament replacement (RTR) [12]. We use RTR with window size $w = \min\{n, N/20\}$ as suggested in [21]. The run is terminated when termination criteria are met.

2.2 Bayesian Networks

Bayesian networks [20, 13] combine graph theory, probability theory and statistics to provide a flexible and practical tool for probabilistic modeling and inference. BOA and hBOA use Bayesian networks to model promising solutions found so far and sample new candidate solutions. A Bayesian network consists of two components: (1) *Structure*, which is defined by an acyclic directed graph with one node per variable and the edges corresponding to conditional dependencies between the variables, and (2) *parameters*, which

consist of the conditional probabilities of each variable given the variables that this variable depends on.

Mathematically, a Bayesian network with n nodes encodes a joint probability distribution of n random variables X_1, X_2, \dots, X_n :

$$p(X_1, X_2, \dots, X_n) = \prod_{i=1}^n p(X_i | \Pi_i), \quad (1)$$

where Π_i is the set of variables from which there exists an edge into X_i (members of Π_i are called parents of X_i). In addition to encoding direct conditional dependencies, a Bayesian network may also encode a number of conditional independence assumptions [20, 13].

hBOA uses Bayesian networks with local structures in the form of dependency trees [6, 10]. That means that the conditional probabilities for each variable are stored in a decision tree, allowing a more efficient representation of conditional dependencies and a more powerful model-building procedure. For more details on learning and sampling Bayesian networks with local structures, see [6, 21].

3. HBOA MODELS FOR SEPARABLE PROBLEMS

We start the analysis of hBOA models on separable problems of bounded difficulty; specifically, we consider concatenated traps of order 5. There are several reasons for analyzing hBOA models on concatenated traps. First of all, many real-world problems are believed to be nearly decomposable [37], and separable problems of bounded difficulty represent a broad class of such problems [11]. Second, there is a solid body of theory that defines what a good probabilistic model should look like in order to provide scalable performance on separable problems of bounded difficulty [11, 31, 21]. Finally, concatenated traps bound the class of decomposable problems of bounded difficulty because they use fully deceptive subproblems [8], which cannot be further decomposed, and any model that fails to accurately represent the problem decomposition is expected to fail to scale up polynomially with problem size [8, 39, 11, 21].

Our analysis will focus on answering two primary questions:

1. Do hBOA models accurately represent problem decomposition for separable problems?
2. How do the models change over time (from generation to generation)?

Answering the above questions is important to better understand how hBOA works on separable problems. But even more importantly, these results provide important information for developing theory and efficiency enhancement techniques for hBOA and nearly decomposable problems of bounded difficulty. Our primary focus will be on the second question as an in-depth analysis of model accuracy in hBOA on traps can be found in [15].

3.1 Concatenated 5-bit Trap

In concatenated 5-bit traps [1, 8], the input string is partitioned into disjoint groups of 5 bits each. This partitioning is unknown to the algorithm, and it does not change during the run. A 5-bit trap function is then applied to each of the groups and the contributions of all the groups are added

together to form the fitness, which we want to maximize. The 5-bit trap function is defined as follows:

$$\text{trap}_5(u) = \begin{cases} 5 & \text{if } u = 5 \\ 4 - u & \text{otherwise} \end{cases}, \quad (2)$$

where u is the number of ones in the input string of 5 bits. An n -bit trap-5 has one global optimum in the string of all ones and $2^{n/5} - 1$ other local optima (the number of local optima grows exponentially fast with problem size).

Trap-5 necessitates that all bits in each trap partition (linkage group) are treated together, because statistics of lower order mislead the search away from the global optimum [8], leading to highly inefficient performance [39, 11].

3.2 A Perfect Model for Trap-5

Since the 5-bit trap described above is fully deceptive [8], to solve concatenated traps in polynomial time, it is necessary that the probabilistic model encodes conditional dependencies between all or nearly all pairs of variables within each trap partition. On the other hand, to maximize the mixing and minimize model complexity, it is desirable that we do not discover any dependencies between the different trap partitions. Therefore, a “perfect” model for solving trap-5 would contain all dependencies between the bits within each trap partition but it would not contain any dependencies between the different trap partitions.

In the remainder of this section, we call the dependencies that we would like to discover the *necessary* dependencies whereas the dependencies that we would like to avoid in order to maximize the mixing will be called *unnecessary*.

3.3 Experimental Setup

To make sure that our results correspond to the actual hBOA performance on trap-5 and are not biased by either a too large or too small population size, we first use bisection [33] to ensure that the population size is large enough to obtain reliable convergence to the global optimum in 30 out of 30 independent runs. The results are then averaged over the 30 successful runs with the population size obtained by bisection. The number of generations is upper bounded according to preliminary experiments and hBOA scalability theory [31] by $2\sqrt{n}$ where n is the number of bits in the problem. Each run of hBOA is terminated when the global optimum has been found (success) or when the upper bound on the number of generations has been reached without discovering the global optimum (failure). To look at the relationship between problem size and the probabilistic models, we consider traps of sizes $n = 15$ to $n = 210$ bits.

3.4 Model Accuracy for Trap-5

We first look at the accuracy of the probabilistic models in hBOA on trap-5; specifically, we are interested in finding out how many of the necessary and unnecessary dependencies are covered. Since numerous empirical results confirmed that hBOA can solve trap-5 in a number of evaluations that grows polynomially with problem size [26, 21], we expect that the necessary dependencies will be discovered; otherwise, the deceptiveness of the trap partitions should lead to poor, exponential scalability [39, 11, 21]. On the other hand, good scalability does not directly necessitate that the number of unnecessary dependencies is relatively small compared to the overall number of dependencies. Quite the contrary, [16] showed that with tournament selection, hBOA often

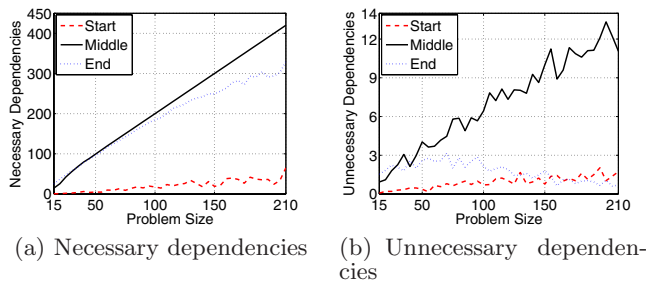


Figure 1: The average number of necessary and unnecessary dependencies with respect to problem size on trap-5. Three model snapshots were taken in each run (the first, middle, and last generation).

finds many unnecessary dependencies when solving trap-5. Nevertheless, the results with truncation selection show that the models closely resemble what we consider the perfect model for trap-5, where trap partitions are fully connected while there are almost no connections between the different traps.

Figure 1 shows the number of necessary and unnecessary dependencies for trap-5 of sizes 15 to 210. For each problem size, the results show the average number of necessary and unnecessary dependencies over the 30 runs using the population size determined by the bisection method. Since the models change over time, we consider three snapshots. The first snapshot shows the model in the first generation, the second snapshot shows the model in the middle of the run, and the last snapshot shows the final model learned before terminating the run.

The results show that initially, the probabilistic model covers only a few of the necessary dependencies. Nonetheless, the model improves over time and the second snapshot shows that in the middle of the run, all or nearly all necessary dependencies are covered (the number of necessary dependencies for trap-5 is $2n$). Finally, late in the run, the model covers many but not all necessary dependencies.

The results from figure 1 are quite intuitive. At the beginning, only some subproblems can be expected to be discovered because the initial population is generated at random and the statistical information from one round of selection does not provide enough information to identify all subproblems. However, after several more iterations, the collateral noise [11] in the population decreases, the overall quality of solutions in the population increases, and the statistical information provides enough input to enable hBOA to learn better models. Finally, at the end of the run, even though accurate problem decomposition should still be important, the dependencies between the subproblems become easy enough to cover with short-order conditional probabilities (because we expect each trap partition to be assigned to either of the two local optima, that is, 00000 or 11111).

We illustrate the argument that in the end of the run our models can significantly simplify without affecting the encoded distribution with the following example. Consider a population of 5-bit binary strings with only two alternative candidate solutions: 00000 and 11111. Late in the run of hBOA on trap-5, every trap partition is expected to have the same or at least a very similar

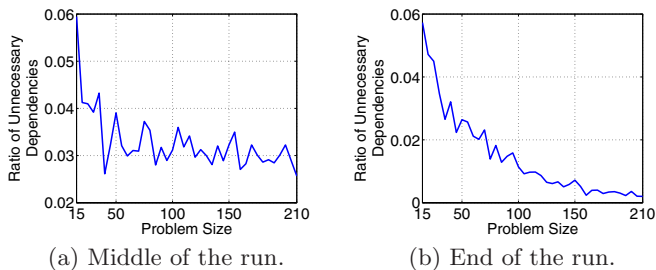


Figure 2: Ratio of the number of unnecessary dependencies to the total number of dependencies over the lifetime for trap-5 of sizes 15 to 210.

distribution to this. Clearly, the value of any bit depends on the value of the remaining bits. Nonetheless, to fully encode this probability distribution, we can use a simple chain model defined as $p(X_1, X_2, X_3, X_4, X_5) = p(X_1)p(X_2|X_1)p(X_3|X_2)p(X_4|X_3)p(X_5|X_4)$. Therefore, as the population diversity decreases and some partial solutions are eliminated, many of the necessary dependencies become unnecessary.

In addition to showing good coverage of necessary dependencies, the results from figure 1 show that the number of unnecessary dependencies is significantly smaller than the number of necessary dependencies. This fact is also exemplified in figure 2, which shows that the ratio of the number of unnecessary dependencies to the number of necessary dependencies decreases with problem size. This is great news—not only does hBOA discover the dependencies we need to solve trap-5 scalably, but it is also capable of avoiding the discovery of a significant number of unnecessary dependencies, which may slow down the mixing.

One of the most surprising results obtained in this study is that the model structure in hBOA significantly depends on the selection method used to select the populations of promising solutions. More specifically, the work on hybridization of hBOA [16] as well as the work on accuracy of hBOA models with tournament selection [15] showed that with tournament selection the number of unnecessary dependencies is relatively significant and that the ratio of the number of unnecessary dependencies to the total number of dependencies increases with problem size. Furthermore, it was shown that with tournament selection nearly all necessary dependencies are discovered at the beginning of the run. Therefore, based on [16, 15], one may conclude that with tournament selection, hBOA models tend to be overly complex. On the contrary, the results presented in figures 1 and 2 show that for truncation selection, the models contain only a handful of unnecessary dependencies, but it takes several generations to discover all the necessary dependencies. Although hBOA scalability is asymptotically the same in both cases [26, 21], studying the influence of the selection method on performance of hBOA and other EDAs appears to be an important topic for future research that has been somewhat neglected in the past.

3.5 Model Dynamics for Trap-5

Knowing that hBOA models for traps are accurate is great, but that’s only one piece of the puzzle and there still remain many important questions. How do the models

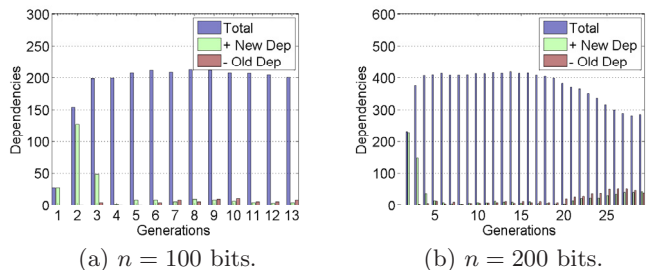


Figure 3: Dependency changes in trap-5 for two different problem sizes.

change over time? Are the models in subsequent generations similar? When do most changes in models occur - is it early in the run or late in the run? All these questions are important, especially for the design of efficiency enhancement techniques, such as sporadic and incremental model building [32] and hybridization [16, 34, 23].

To better understand model dynamics in hBOA, we analyze changes in hBOA models in subsequent generations of hBOA. In each generation, we record the number of dependencies that were not present in the previous generation but have been added in the current one, and analogically we record the number of dependencies that were present in the previous generation but that have been eliminated in the current one. Figure 3 shows the obtained results for trap-5 of size 100 and 200 (the results for other problem sizes look similar). The results clearly indicate that most dependencies are added in the first few generations of hBOA. After the correct or approximately correct model for the underlying problem is discovered, the models do not change significantly.

The stability of hBOA models on trap-5 is great news for sporadic and incremental model building [32], which are efficiency enhancement techniques that focus on improving efficiency of model building in hBOA. More specifically, both sporadic and incremental model building lead to the highest gains in performance when models in subsequent generations have similar structure. Our results indicate that this is indeed the case, at least for separable problems of bounded difficulty.

While stability of hBOA models is important for incremental and sporadic model building, the rapid learning of an accurate model is also an important precondition for efficient hBOA hybrids using specialized local operators that are based on the learned models [16, 34].

4. HBOA MODELS FOR 2D SPIN GLASSES

We have shown on a difficult separable problem of bounded order that hBOA learns the adequate problem decomposition quickly and accurately, but what happens when we move to a harder problem that cannot be broken up into subproblems of bounded order? To answer this question, we consider the problem of finding ground states of 2D Ising spin glasses [4, 17, 9, 41] where the task is to find spin configurations that minimize the energy of a given spin glass instance. While the structure of the energy function for 2D Ising spin glasses is relatively easy to understand, the problem of finding spin-glass ground states represents a great challenge for most optimization techniques because

(1) the energy landscape contains an extremely large number of local optima, (2) the local optima in the energy landscape are often separated by configurations with very high energy (low-quality regions), (3) any decomposition of bounded order is insufficient for solving the problem, and (4) the problem is still solvable in polynomial time using analytical methods. This is why most standard and advanced optimization techniques fail to solve this problem in polynomial time, including classical genetic algorithms [23] and state-of-the-art Markov chain Monte Carlo (MCMC) methods such as the flat-histogram MCMC [7]. Despite the difficulty of the Ising spin glass problem, hBOA is able to solve 2D Ising spin glasses in polynomial time, achieving empirical asymptotic performance of the best analytical methods without any problem-specific knowledge [23].

4.1 2D Ising Spin Glass

A very simple model to describe a finite-dimensional Ising spin glass is typically arranged on a regular 2D or 3D grid where each node i corresponds to a spin s_i and each edge $\langle i, j \rangle$ corresponds to a coupling between two spins s_i and s_j . For the classical Ising model, each spin s_i can be in one of two states: $s_i = +1$ or $s_i = -1$. Each edge $\langle i, j \rangle$ has a real value (coupling) $J_{i,j}$ associated with it that defines the relationship between the two connected spins. To approximate the behavior of the large-scale system, periodic boundary conditions are often used that introduce a coupling between the first and the last element along each dimension. In this paper we consider the 2D Ising spin glass with periodic boundary conditions.

Given a set of coupling constants $J_{i,j}$, and a configuration of spins $C = \{s_i\}$, the energy can be computed as $E(C) = \sum_{\langle i,j \rangle} s_i J_{i,j} s_j$, where the sum runs over all couplings $\langle i, j \rangle$. Here the task is to find a spin configuration given couplings $\{J_{i,j}\}$ that minimizes the energy of the spin glass. The states with minimum energy are called *ground states*. The spin configurations are encoded with binary strings where each bit specifies the value of one spin (0 for +1, 1 for -1).

In order to obtain a quantitative understanding of the disorder in a spin glass system introduced by the random spin-spin couplings, one generally analyzes a large set of random spin glass instances for a given distribution of the spin-spin couplings. For each spin glass instance, the optimization algorithm is applied and the results are analyzed. Here we consider the $\pm J$ spin glass, where each spin-spin coupling constant is set randomly to either +1 or -1 with equal probability. All instances of sizes up to 18×18 with ground states were obtained from S. Sabhapandit and S. N. Coppersmith from the University of Wisconsin who identified the ground states using flat-histogram Markov chain Monte Carlo simulations [7]. The ground states of the remaining instances were obtained from the Spin Glass Ground State Server at the University of Cologne [38].

To improve the performance of hBOA, we incorporated a deterministic local search—deterministic hill climber (DHC)—to improve quality of each evaluated solution [23, 21]. DHC proceeds by making single-bit flips that improve the quality of the solution most until no single-bit flip improves the candidate solution. DHC is applied to every solution in the population before it is evaluated.

4.2 Perfect Model for 2D Ising Spin Glass

With spin glasses it is not clear what dependencies are

really necessary to solve the problem scalably and neither is it clear what dependencies are unnecessary and should be avoided to maximize the mixing. In fact, to some degree, every spin (bit) depends on every other spin either directly through one connection (coupling) or through a chain of connections. It has been argued [18] that to fully cover all necessary dependencies, the order of dependencies in the probabilistic model must grow at least as fast as $\Omega(\sqrt{n})$. Of course, since hBOA has been shown to solve 2D Ising spin glasses in polynomial time and due to the initial-supply population sizing [11] the population size in hBOA is lower bounded by $\Omega(2^k)$ (where k is the order of dependencies covered by the probabilistic model), the models in hBOA cannot encode dependencies of order $\Omega(\sqrt{n})$ or more.

Nonetheless, while it is unclear what a “perfect” model for a 2D Ising spin glass looks like, it is clear that the interactions between immediate neighbors should be strongest and the interactions should decrease in magnitude with the distance of spins measured by the number of links between these spins. This hypothesis will be confirmed with the experimental results presented shortly.

4.3 Experimental Setup

Most hBOA parameters were set the same as in the experiments for concatenated traps. Since the difficulty of spin glass instances varies significantly depending on the couplings [7, 23], we generated 100 random instances for each considered problem size. On each spin glass instance we used the population size obtained by bisection but since we used 100 random instances for each problem size, we required hBOA to converge only in 5 out of 5 independent runs.

4.4 Model Dynamics and Structure for 2D Ising Spin Glass

Given our initial understanding of the structure of 2D Ising spin glass, we start by analyzing the grid distances of the spins directly connected by a dependency in the probabilistic model, where the distance of two spins is defined as the minimum number of links in the 2D grid that we must pass to get from one spin to the other one. The minimum distance, one, is between any immediate neighbors in the grid. Due to the periodic boundary conditions, the maximum distance is *not* between the opposite corners of the grid. Instead, the maximum distance is, for example, between the top-left spin and the spins around the middle of the grid; for instance, for 2D spin glasses of size 20×20 the maximum distance is 20 (10 links to the right, 10 links to the bottom). Also of note is that the number of possible dependencies of any given distance increases to a maximum at half of the maximum distance and then decreases to having only one possible dependency at the maximum distance. For example, any particular spin in a 2D spin glass of size 20×20 could have 38 possible dependencies of distance 10 but only 1 dependency possible of length 20. To analyze distances of spins that are connected in the model, we created a histogram of the average number of dependencies for each distance value at different stages of the hBOA run on 100 randomly generated $\pm J$ Ising spin glasses. We expected that early in the run, the majority of the dependencies would be between immediate neighbors, whereas later in the run, the number of dependencies between farther spins would increase. We obtained four snapshots of the models at equally distributed time intervals to cover the entire run of hBOA.

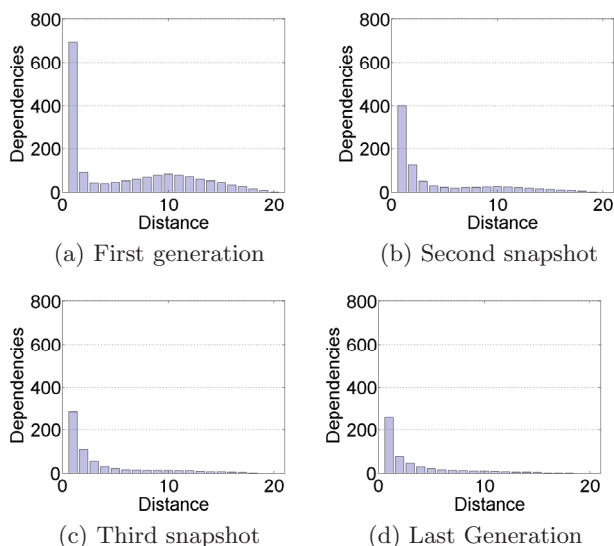


Figure 4: Distribution of dependencies with respect to spin distances for hBOA with DHC.

Figure 4 shows that in all stages of the run, the number of dependencies between immediate neighbors is as large as the number of the remaining dependencies. Since the influence of each spin on the distribution of another spin decreases significantly with the number of links between the spins, we believe that (at least initially) the dependencies of spins that are located far away are not necessary and are only a result of the strong problem regularities and noisy information of the initial population. However, as the run proceeds, long-range dependencies may become more meaningful as the search progresses to locate the candidate solutions in the most promising regions of the search space. Further analysis of the obtained results remains an important topic for future work; more insight could be obtained by analyzing the energy landscape from the perspective of the populations at different stages of the hBOA run.

One of the surprising results is that the number of dependencies at any distance decreases over time and therefore in each generation of hBOA, the model becomes simpler. We believe that while the effective order of dependencies encoded by the model increases over the run, since these dependencies become simpler (they cover fewer alternative partial solutions), they can be covered with structures composed of short-order dependencies. Also of note is the increased number of mid-range dependencies that exist early on. We believe this is a result of the increased total number of dependencies possible at these distances and is caused by noise early on in the run.

To check whether the spin glass results are significantly influenced by DHC, we repeated the same experiments without DHC (see figure 5). The results without DHC indicate that the structure of the models looks similar to that found with DHC; however, the long-range dependencies become even more significant without DHC than with DHC. In fact, without DHC, the models are generally more complex.

Just as with trap-5, we next look at how stable individual dependencies are in subsequent generations. In each generation, we record the number of dependencies that were not present in the previous generation but have been added in

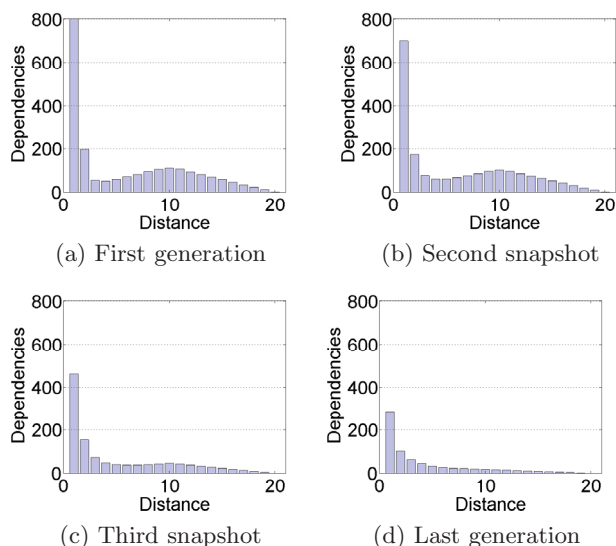


Figure 5: Distribution of dependencies with respect to spin distances for hBOA without DHC.

the current one, and analogically we record the number of dependencies that were present in the previous generation but that have been eliminated in the current one. Figure 6a shows the obtained results for one run of a 20×20 spin glass (the results for other runs are similar). The results seemed to indicate that we had a great deal of change, however in Figure 6b we repeated the experiment and only looked at neighbor dependencies. The results clearly show that the short-range dependencies are very stable and the changes seen in Figure 6a are due to the farther dependencies changing. Since many long-range dependencies can be expected to be unnecessary as is also supported by the results presented in the next section, the most important factor regarding the stability of hBOA models is the stability with respect to the short-range dependencies.

4.5 Restricting hBOA Models on Spin Glass

We have observed that most of the dependencies hBOA finds are the short dependencies between neighbors in the grid. So the following question arises, if restricted to only dependencies of distance one, would hBOA still solve the problem scalably and reliably? If not, could we at least restrict the dependencies to consider distances of at most two or three? If we could restrict hBOA models significantly without negatively affecting hBOA scalability, these restrictions could be used both to significantly speed up hBOA as well as to design powerful problem-specific recombination operators for spin glasses and other similar problems.

To answer the above question, we looked at the scalability of hBOA with the different restrictions on the dependencies using spin glass instances of sizes from $10 \times 10 = 100$ spins to $25 \times 25 = 625$ spins, using 100 instances for each problem size. For each problem instance and each hBOA variant, we used bisection to find an adequate population size and recorded the average number of evaluations until convergence. The results are shown in figure 7.

The results show that while performance of hBOA restricted to neighbor dependencies is comparable to the performance of the original hBOA on the smallest spin glass

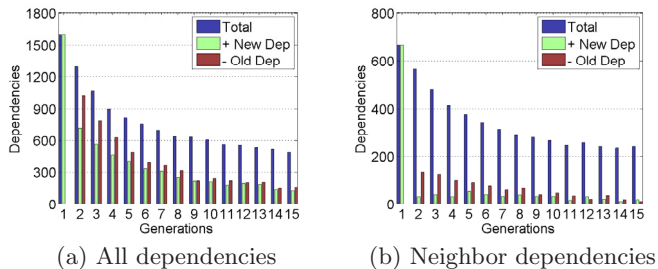


Figure 6: Dependency changes for one run of a 20×20 spin glass.

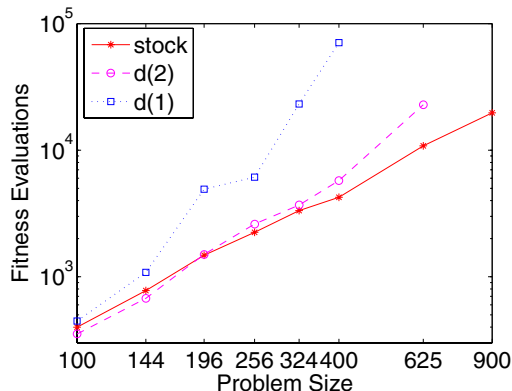


Figure 7: Scalability of hBOA with DHC with and without restrictions on model structure.

instances of size 10×10 , as the problem size increases, performance of hBOA with only neighbor dependencies deteriorates quickly and the number of evaluations grows exponentially. Performance of hBOA with dependencies restricted to spins at distances of at most two leads to performance comparable to that of the original hBOA up to the problems of size 18×18 . However, as the problem size is further increased, the performance of the restricted hBOA again starts to deteriorate. In particular, the harder instances were affected much more severely by the restrictions.

The scalability analysis thus indicates that to solve 2D spin glasses scalably we cannot restrict the dependencies to be only between the immediate neighbors or spins located at the distance of at most two. This result agrees with the results presented in [42]. The result also shows that the hand-designed approximate Bayesian network structure proposed for 2D Ising spin glasses in [18] is unlikely to scale up because it relates only the spins at distance one or two. Nonetheless, the results show that we can restrict hBOA models significantly although the degree to which we can restrict the models appears to depend on the problem size, assuming that we do not want to affect hBOA scalability. Because of the important implications of restricting complexity of hBOA probabilistic models, this topic should be further investigated in future research.

The implications of these results go beyond the direct application of hBOA to spin glasses. Specifically, we show that none of the probabilistic models designed by hand for 2D Ising spin glasses in the past seems to ensure polynomially scalable performance. While the probabilistic model based

on the running intersection property leads to intractable performance due to the large order of dependencies it uses [18], the approximate probabilistic model based on statistical terms of short order [18] fails to scale up for large problems. Thus, in spite of the availability of the complete knowledge of the problem specifics, models obtained with automatic model building procedures of hBOA outperform hand-crafted models.

5. CONCLUSIONS AND FUTURE WORK

Since both the scalability of hBOA as well as the effectiveness of several efficiency enhancement techniques for hBOA crucially depend on the quality of the probabilistic models learned by hBOA, analysis of probabilistic models in hBOA is an important research topic. This paper analyzed the structure and complexity of hBOA probabilistic models on two common test problems: concatenated traps and 2D Ising spin glasses with periodic boundary conditions.

For concatenated traps, we know what an adequate probabilistic model looks like to ensure scalable and reliable convergence with hBOA. The results show that hBOA is able to learn such adequate probabilistic models quickly and that the learned problem decomposition corresponds closely to the underlying problem. The results also show that the models on concatenated traps do not change much from generation to generation, which is important especially for the effectiveness of sporadic and incremental model building.

While for 2D Ising spin glasses it is unclear what is an ideal probabilistic model, the probabilistic models obtained by hBOA are shown to correspond closely to the structure of the underlying problem. Furthermore, the results indicate that while it seems to be possible to restrict hBOA models significantly without affecting hBOA performance much, probabilistic models designed by hand may not lead to good scalability despite the robust and scalable performance of hBOA based on automatic model building procedures.

Although the analysis of Bayesian networks with local structures is a challenging task, this paper shows that the analysis of hBOA models is indeed possible and that such an analysis can reveal useful information about the underlying problem.

A direct extension of this work is analysis of probabilistic models in hBOA on other challenging problems, considering both artificial and real-world problems. Another important topic for future research is the development of techniques to effectively use information obtained from the analysis of hBOA models and the use of this information for the design of efficiency enhancement techniques.

Acknowledgments

This project was sponsored by the National Science Foundation under CAREER grant ECS-0547013, by the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant FA9550-06-1-0096, and by the University of Missouri in St. Louis through the High Performance Computing Collaboratory sponsored by Information Technology Services, and the Research Award and Research Board programs. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation, the Air Force Office of Scientific Research, or the U.S. Government.

6. REFERENCES

- [1] D. H. Ackley. An empirical study of bit vector function optimization. *Genetic Algorithms and Simulated Annealing*, pages 170–204, 1987.
- [2] S. Baluja. Population-based incremental learning: A method for integrating genetic search based function optimization and competitive learning. Tech. Rep. No. CMU-CS-94-163, Carnegie Mellon University, Pittsburgh, PA, 1994.
- [3] F. Barahona. On the computational complexity of Ising spin glass models. *Journal of Physics A: Mathematical, Nuclear and General*, 15(10):3241–3253, 1982.
- [4] K. Binder and A. Young. Spin-glasses: Experimental facts, theoretical concepts and open questions. *Rev. Mod. Phys.*, 58:801, 1986.
- [5] P. A. N. Bosman and D. Thierens. Continuous iterated density estimation evolutionary algorithms within the IDEA framework. *Workshop Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000)*, pages 197–200, 2000.
- [6] D. M. Chickering, D. Heckerman, and C. Meek. A Bayesian approach to learning Bayesian networks with local structure. Technical Report MSR-TR-97-07, Microsoft Research, Redmond, WA, 1997.
- [7] P. Dayal, S. Trebst, S. Wessel, D. Würtz, M. Troyer, S. Sabhapandit, and S. Coppersmith. Performance limitations of flat histogram methods and optimality of Wang-Langdau sampling. *Physical Review Letters*, 92(9):097201, 2004.
- [8] K. Deb and D. E. Goldberg. Analyzing deception in trap functions. IlliGAL Report No. 91009, University of Illinois at Urbana-Champaign, Illinois Genetic Algorithms Laboratory, Urbana, IL, 1991.
- [9] K. Fischer and J. Hertz. *Spin Glasses*. Cambridge University Press, Cambridge, 1991.
- [10] N. Friedman and M. Goldszmidt. Learning Bayesian networks with local structure. In M. I. Jordan, editor, *Graphical models*, pages 421–459. MIT Press, 1999.
- [11] D. E. Goldberg. *The design of innovation: Lessons from and for competent genetic algorithms*. Kluwer, 2002.
- [12] G. R. Harik. Finding multimodal solutions using restricted tournament selection. *International Conference on Genetic Algorithms (ICGA-95)*, pages 24–31, 1995.
- [13] R. A. Howard and J. E. Matheson. Influence diagrams. In R. A. Howard and J. E. Matheson, editors, *Readings on the principles and applications of decision analysis*, volume II, pages 721–762. Strategic Decisions Group, Menlo Park, CA, 1981.
- [14] P. Larrañaga and J. A. Lozano, editors. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*. Kluwer, Boston, MA, 2002.
- [15] C. F. Lima et al. Structural accuracy of probabilistic models in BOA. Technical report, University of Algarve, 2007.
- [16] C. F. Lima, M. Pelikan, K. Sastry, M. V. Butz, D. E. Goldberg, and F. G. Lobo. Substructural neighborhoods for local search in the Bayesian optimization algorithm. *Parallel Problem Solving from Nature*, pages 232–241, 2006.
- [17] M. Mezard, G. Parisi, and M. Virasoro. *Spin glass theory and beyond*. World Scientific, Singapore, 1987.
- [18] H. Mühlenbein, T. Mahnig, and A. Ochoa-Rodriguez. Schemata, distributions and graphical models in evolutionary optimization. *Journal of Heuristics*, 5:215–247, 1999.
- [19] H. Mühlenbein and G. Paaß. From recombination of genes to the estimation of distributions I. Binary parameters. *Parallel Problem Solving from Nature*, pages 178–187, 1996.
- [20] J. Pearl. *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann, San Mateo, CA, 1988.
- [21] M. Pelikan. *Hierarchical Bayesian optimization algorithm: Toward a new generation of evolutionary algorithms*. Springer-Verlag, 2005.
- [22] M. Pelikan and D. E. Goldberg. Escaping hierarchical traps with competent genetic algorithms. *Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 511–518, 2001.
- [23] M. Pelikan and D. E. Goldberg. Hierarchical BOA solves Ising spin glasses and MAXSAT. *Genetic and Evolutionary Computation Conference (GECCO-2003)*, II:1275–1286, 2003.
- [24] M. Pelikan and D. E. Goldberg. A hierarchy machine: Learning to optimize from nature and humans. *Complexity*, 8(5):36–45, 2003.
- [25] M. Pelikan and D. E. Goldberg. Hierarchical Bayesian optimization algorithm. In M. Pelikan, K. Sastry, and E. Cantú-Paz, editors, *Scalable optimization via probabilistic modeling: From algorithms to applications*, pages 63–90. Springer, 2006.
- [26] M. Pelikan, D. E. Goldberg, and E. Cantú-Paz. BOA: The Bayesian optimization algorithm. *Genetic and Evolutionary Computation Conference (GECCO-99)*, I:525–532, 1999.
- [27] M. Pelikan, D. E. Goldberg, and F. Lobo. A survey of optimization by building and using probabilistic models. *Computational Optimization and Applications*, 21(1):5–20, 2002.
- [28] M. Pelikan and A. K. Hartmann. Searching for ground states of Ising spin glasses with hierarchical BOA and cluster exact approximation. In M. Pelikan, K. Sastry, and E. Cantú-Paz, editors, *Scalable optimization via probabilistic modeling: From algorithms to applications*, pages 333–349. Springer, 2006.
- [29] M. Pelikan and K. Sastry. Fitness inheritance in the Bayesian optimization algorithm. *Genetic and Evolutionary Computation Conference (GECCO-2004)*, 2:48–59, 2004.
- [30] M. Pelikan, K. Sastry, M. V. Butz, and D. E. Goldberg. Performance of evolutionary algorithms on random decomposable problems. *Parallel Problem Solving from Nature (PPSN IX)*, pages 788–797, 2006.
- [31] M. Pelikan, K. Sastry, and D. E. Goldberg. Scalability of the Bayesian optimization algorithm. *International Journal of Approximate Reasoning*, 31(3):221–258, 2002.
- [32] M. Pelikan, K. Sastry, and D. E. Goldberg. Sporadic model building for efficiency enhancement of hierarchical BOA. pages 405–412, 2006.
- [33] K. Sastry. Evaluation-relaxation schemes for genetic and evolutionary algorithms. Master’s thesis, University of Illinois at Urbana-Champaign, Department of General Engineering, Urbana, IL, 2001.
- [34] K. Sastry and D. E. Goldberg. Designing competent mutation operators via probabilistic model building of neighborhoods. *Genetic and Evolutionary Computation Conference (GECCO-2004)*, pages 114–125, 26-30 2004.
- [35] K. Sastry, D. E. Goldberg, and M. Pelikan. Don’t evaluate, inherit. *Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 551–558, 2001.
- [36] K. Sastry, M. Pelikan, and D. E. Goldberg. Efficiency enhancement of estimation of distribution algorithms. In M. Pelikan, K. Sastry, and E. Cantú-Paz, editors, *Scalable Optimization via Probabilistic Modeling: From Algorithms to Applications*, pages 161–185. Springer, 2006.
- [37] H. A. Simon. *The Sciences of the Artificial*. The MIT Press, Cambridge, MA, 1968.
- [38] Spin Glass Ground State Server. http://www.informatik.uni-koeln.de/lis_juenger/research/sgs/sgs.html, 2004. University of Köln, Germany.
- [39] D. Thierens and D. E. Goldberg. Mixing in genetic algorithms. *International Conference on Genetic Algorithms (ICGA-93)*.
- [40] H. Wu and J. L. Shapiro. Does overfitting affect performance in estimation of distribution algorithms. pages 433–434, 2006.
- [41] A. Young, editor. *Spin glasses and random fields*. World Scientific, Singapore, 1998.
- [42] T.-L. Yu. *A Matrix Approach for Finding Extreme: Problems with Modularity, Hierarchy, and Overlap*. PhD thesis, University of Illinois at Urbana-Champaign, Urbana, IL, 2006.