Missouri Estimation of Distribution Algorithms Laboratory

Obtaining Ground States of Ising Spin Glasses via Optimizing Bonds Instead of Spins

Martin Pelikan and Alexander K. Hartmann

MEDAL Report No. 2007002

January 2007

Abstract

Frustrated Ising spin glasses represent a rich class of challenging optimization problems that share many features with other complex, highly multimodal optimization and combinatorial problems. This paper shows that transforming candidate solutions to an alternative representation that is strongly tied to the energy function simplifies the exploration of the space of potential spin configurations and that it significantly improves performance of evolutionary algorithms with simple variation operators on Ising spin glasses. The proposed techniques are incorporated into the simple genetic algorithm, the univariate marginal distribution algorithm, and the hierarchical Bayesian optimization algorithm.

Keywords

Genetic algorithm, hierarchical BOA, univariate marginal distribution algorithm, spin glass, problem transformation, hybrid evolutionary algorithms, estimation of distribution algorithms, scalability.

Missouri Estimation of Distribution Algorithms Laboratory (MEDAL)
Department of Mathematics and Computer Science
University of Missouri–St. Louis
One University Blvd., St. Louis, MO 63121
E-mail: medal@cs.umsl.edu
WWW: http://medal.cs.umsl.edu/

Obtaining Ground States of Ising Spin Glasses via Optimizing Bonds Instead of Spins

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

Alexander K. Hartmann
Institut für Theoretische Physik
Universität Göttingen
Friedrich-Hund-Platz 1
37077 Göttingen, Germany
hartmann@physik.uni-goettingen.de

Abstract

Frustrated Ising spin glasses represent a rich class of challenging optimization problems that share many features with other complex, highly multimodal optimization and combinatorial problems. This paper shows that transforming candidate solutions to an alternative representation that is strongly tied to the energy function simplifies the exploration of the space of potential spin configurations and that it significantly improves performance of evolutionary algorithms with simple variation operators on Ising spin glasses. The proposed techniques are incorporated into the simple genetic algorithm, the univariate marginal distribution algorithm, and the hierarchical Bayesian optimization algorithm.

Keywords: Genetic algorithm, hierarchical BOA, univariate marginal distribution algorithm, spin glass, problem transformation, hybrid evolutionary algorithms, estimation of distribution algorithms, scalability.

1 Introduction

Ising spin glasses are prototypical models for disordered systems and have played a central role in statistical physics during the last three decades (Binder & Young, 1986; Mezard, Parisi, & Virasoro, 1987; Fischer & Hertz, 1991; Young, 1998). In optimization, spin glasses provide a rich class of test problems that represent a challenge for most optimizers (Naudts & Naudts, 1998; Mühlenbein & Mahnig, 1999; Hartmann, 2001; Hartmann & Rieger, 2001; Hartmann & Rieger, 2004; Pelikan & Goldberg, 2003; Höns, 2005; Santana, 2005; Pelikan & Hartmann, 2006; Shakya, McCall, & Brown, 2006). Spin glasses are interesting from the optimization perspective mainly because of the large number of local optima and the existence of rough energy landscapes (Barahona et al., 1982; Hartmann, 1998; Hed et al., 2001; Pelikan & Goldberg, 2003; Dayal et al., 2004; Höns, 2005; Pelikan & Hartmann, 2006).

This paper describes how to transform spin configurations into vectors of Boolean variables, one for each coupling constant ($S \to C$), and how to transform the Boolean vectors for couplings back into spin configurations ($C \to S$). We show that performance of the simple genetic algorithm improves by applying transformation $S \to C$ before applying the variation operators and using transformation $C \to S$ to incorporate the configurations into the original population. Although the transformation $S \to C$ increases the number of bits processed by the variation operators, the proposed approach leads to speedups that grow with problem size. Nonetheless, when applying the more advanced hierarchical Bayesian optimization algorithm (hBOA) (Pelikan & Goldberg, 2001; Pelikan, 2005), problem transformation does not improve performance. That indicates that problem transformation simplifies the problem for simple variation operators, but that the problem transformation does not lead to positive effects with advanced variation operators that are capable of identifying and exploiting important problem regularities automatically.

The paper is organized as follows. Section 2 outlines the evolutionary algorithms discussed in this paper. Section 3 describes the problem of finding ground states of Ising spin glasses. Section 4 outlines the methods that can be used to implement the transformations $S \to C$ and $C \to S$, which will be used to temporarily transform candidate solutions for the variation operators. Section 5 presents experimental results. Section 6 outlines future work. Finally, Section 7 summarizes and concludes the paper.

2 Algorithms

This section outlines the optimization algorithms discussed in this paper: (1) the genetic algorithm (GA) (Holland, 1975; Goldberg, 1989), (2) the univariate marginal distribution algorithm (UMDA), and (3) the hierarchical Bayesian optimization algorithm (hBOA) (Pelikan & Goldberg, 2001; Pelikan, 2005). GA, UMDA and hBOA differ in the way they process selected solutions to generate new candidate solutions. Additionally, the section describes the deterministic hill climber (DHC) (Pelikan & Goldberg, 2003), which is incorporated into all compared algorithm to improve their performance. In all compared algorithms, candidate solutions are represented by binary strings of n bits.

2.1 Genetic algorithm (GA)

The genetic algorithm (GA) (Holland, 1975; Goldberg, 1989) evolves a population of candidate solutions represented by fixed-length binary strings. The first population is generated at random. Each iteration starts by selecting promising solutions from the current population. We use binary tournament selection, specifically restricted tournament replacement (RTR) (Harik, 1995). New solutions are created by applying variation operators to the population of selected solutions. Specifically, crossover is used to exchange bits and pieces between pairs of candidate solutions and mutation is used to perturb the resulting solutions. Here we use two-point crossover and bit-flip mutation (Goldberg, 1989). To ensure useful diversity maintenance, the new candidate solutions are incorporated into the original population using. The run is terminated when termination criteria are met, see below.

2.2 Univariate Marginal Distribution Algorithm (UMDA)

The univariate marginal distribution algorithm (UMDA) (Mühlenbein & Paaß, 1996) also evolves a population of candidate solutions represented by binary strings, starting with a random population.

Each iteration starts by selection. Then, the probability vector is learned that stores the proportion of 1s in each position of the selected population. Each bit of a new candidate solution is then set to 1 with the probability equal to the proportion of 1s in this position; otherwise, the bit is set to 0. Consequently, the variation operator of UMDA preserves the proportions of 1s in each position while decorrelating different string positions. The new candidate solutions are incorporated into the original population using RTR. The run is terminated when termination criteria are met.

2.3 Hierarchical BOA (hBOA)

The hierarchical Bayesian optimization algorithm (hBOA) (Pelikan & Goldberg, 2001; Pelikan, 2005) also 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. New solutions are generated by building a Bayesian network with decision trees (Chickering, Heckerman, & Meek, 1997; Friedman & Goldszmidt, 1999) 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) (Harik, 1995). The run is terminated when termination criteria are met.

UMDA and hBOA are estimation of distribution algorithms (EDAs) (Baluja, 1994; Mühlenbein & Paaß, 1996; Larrañaga & Lozano, 2002; Pelikan, Goldberg, & Lobo, 2002), also called probabilistic model-building genetic algorithms (PMBGAs) (Pelikan, Goldberg, & Lobo, 2002) and iterated density estimation algorithms (IDEAs) (Bosman & Thierens, 2000). EDAs replace standard variation operators of genetic algorithms (GAs) such as crossover and mutation by building a probabilistic model of promising solutions and sampling the built model to generate new candidate solutions.

2.4 Deterministic Hill Climber (DHC)

The deterministic hill climber (DHC) is incorporated to GA, UMDA and hBOA to improve their performance. DHC takes a candidate solution represented by an n-bit binary string on input. Then, it performs one-bit changes on the solution that lead to the maximum improvement of solution quality (maximum decrease in energy). DHC is terminated when no single-bit flip improves solution quality and the solution is thus locally optimal. Here, DHC is used to improve every solution in the population before the evaluation is performed. The hybrids created by incorporating DHC into GA, UMDA and hBOA are referred to as GA+DHC, UMDA+DHC and hBOA+DHC, respectively.

3 Ising Spin Glasses

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. Each edge has a real value $J_{i,j}$ 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 elements in each row along each dimension.

For the classical Ising model, each spin s_i can be in one of two states: $s_i = +1$ or $s_i = -1$. Note that this simplification corresponds to highly anisotropic systems, which do indeed exist in some experimental situations. Nevertheless, the two-state Ising model comprises the most-important effects also found in models with more degrees of freedom. A specific set $\{J_{i,j}\}$ of coupling constants defines a spin glass instance. Each possible setting $C = \{s_i\}$ of all spins is called a spin configuration.

Given a set of coupling constants $J_{i,j}$, and a configuration of spins C, the energy can be computed as

$$E(C) = \sum_{\langle i,j \rangle} s_i J_{i,j} s_j \ , \tag{1}$$

where the sum runs over all couplings $\langle i, j \rangle$. For a given spin configuration, couplings which have $s_i J_{i,j} s_j$ are called *satisfied*, unsatisfied else.

Given a set of coupling constants, the usual task in statistical physics is to integrate a known function over all possible configurations of spins, assuming the Boltzmann distribution of spin configurations; that means, the probability of each configuration C is proportional to $\exp\left(-E(C)/T\right)$ where T is temperature. From the physics point of view, it is also interesting to know the ground states (configurations associated with the minimum possible energy). Finding extremal energies then corresponds to sampling the Boltzmann distribution with temperature approaching 0 and thus the problem of finding ground states is simpler *a priori* than integration over a wide range of temperatures. However, most of the conventional methods based on sampling the above Boltzmann distribution fail to find the ground states because they get often trapped in a local minimum (Dayal, Trebst, Wessel, Würtz, Troyer, Sabhapandit, & Coppersmith, 2004).

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 to obtain a measure of computational complexity. In many cases one has to study different system sizes up to large sizes to obtain a correct view on the spin-glass behavior (Hartmann & Moore, 2003), hence sophisticated optimization algorithms are needed. 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. This leads to *frustration*, i.e. it is impossible to find a spin configuration, which satisfies all bonds simultaneously. Due to frustration and because of the highly multimodal and rough energy landscape, $\pm J$ Ising spin glasses are much more difficult to optimize than the simple Ising model, in which all couplings are set to the same value (Barahona, Maynard, Rammal, & Uhry, 1982; Hartmann & Rieger, 2001; Pelikan & Goldberg, 2003; Hartmann & Rieger, 2004).

4 Transforming Spin Configurations to Boolean Coupling Vectors

The most straightforward approach to applying optimization algorithms to the problem of finding ground states of Ising spin glasses is to encode spin configurations as binary vectors (spin $+1$ is encoded by 1, spin -1 is encoded by 0), and set the fitness of each candidate spin configuration to grow inversely proportionally to the energy of this configuration. Then, any popular optimization algorithm for binary vectors can be applied to find ground states.

This section describes a modified approach, in which the exploration of the space of candidate solutions is performed using an alternative representation of candidate solutions. Specifically, we transform promising spin configurations to coupling vectors (transformation $S \to C$) and then perform exploration in the space of coupling vectors. Newly discovered coupling vectors are then transformed back into spin configurations ($C \to S$) in order to obtain their energy and eliminate conflicting constraints. The purpose of performing exploration in the modified domain is to make exploration easier and consequently enhance the efficiency of optimization techniques for solving the spin glass problem with the special focus on evolutionary algorithms studied in this paper.

4.1 Motivation

The problem of finding ground states for a given set of couplings connecting the neighbors in the underlying grid can be formulated as a *weighted constraint satisfaction problem*. More specifically, each coupling can be seen as a constraint on the spins it connects. A positive coupling $J_{i,j} > 0$ between two spins indicates that, to minimize the energy, hence satisfy the coupling, spins s_i and s_j should be different (either $s_i = +1$ and $s_j = -1$, or $s_i = -1$ and $s_j = +1$). A negative coupling $J_{i,j} < 0$ indicates that the two spins s_i and s_j should have the same values. For the general case, each constraint is weighted by the absolute value of the corresponding coupling constant, which is just $|J_{i,j}| = 1$ here. The task of finding a spin configuration with minimum energy then corresponds to finding a spin configuration that maximizes the weighted sum of satisfied constraints (or minimizes the weighted sum of unsatisfied constraints).

The main source of difficulty in solving spin glasses is that low-energy configurations are often far from each other in terms of the number of matching spins (hamming distance) and that configurations that are located between the local optima are often of very low quality. Consequently it becomes difficult to explore the space of the local optima and the search for the global optimum becomes intractable for many standard optimization methods (Dayal, Trebst, Wessel, Würtz, Troyer, Sabhapandit, & Coppersmith, 2004; Pelikan & Goldberg, 2003).

Nonetheless, while different with respect to the values of different spins, low-energy configurations can be expected to share many similarities in terms of the coupling constraints that are satisfied (or unsatisfied) (Houdayer & Martin, 1999; Hartmann & Rieger, 2001; Hartmann & Rieger, 2004). Since variation operators in genetic and evolutionary algorithms work most effectively if high-quality solutions share many similarities, it can be hypothesized that if variation was processing regularities in the space of couplings as opposed to the space of spins, the exploration of the space of local optima would become more effective and the performance would improve. This paper verifies this hypothesis and provides empirical evidence that variation does indeed become more effective if it processes information on the constraints (couplings) instead of spin values.

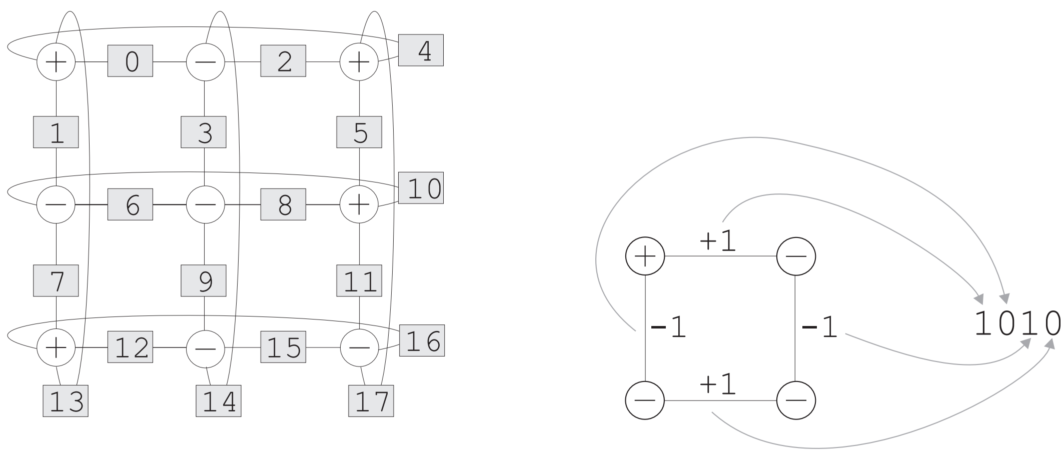
4.2 From Spins to Couplings

Let us assume that the total number of spins is n and the number of couplings between the spins is $m \geq n$. For 2D spin glasses with periodic boundary conditions, $m = 2n$, whereas for 3D spin glasses with periodic boundary conditions, $m = 3n$. The Boolean vectors for couplings contain m bits, where each bit corresponds to a unique coupling constraint between a pair of spins. Let's denote the index of the bit corresponding to the coupling $J_{i,j}$ by $a(i,j)$.

To transform a specific spin configuration $\{s_i\}$ into a Boolean coupling vector $\{c_i\}$, for each coupling $J_{i,j}$ connecting spins s_i and s_j we set

$$c_{a(i,j)} = \begin{cases} 1 & \text{if } \; s_i J_{i,j} s_j < 0 \\ 0 & \text{otherwise} \end{cases}$$

For 2D spin glasses, the order in which the couplings are encoded in the coupling vectors is determined by iterating through spins from the top-left corner to the bottom-right corner in row-major order, and in each iteration considering connections going to the right and to the bottom of the current spin (periodic boundaries are included in the right-most and bottom-most elements). This encoding of couplings ensures that couplings that are located close to each other in the spin



(a) Indexing the couplings before transformation. (b) Example of spin-glass transformation.

Figure 1: Transformation of a spin glass into a Boolean vector for coupling constants proceeds from top-left to bottom-right in row-major order. Part (a) shows how the couplings are indexed (index is shown in shaded rectangles). Part (b) shows the transformation for one sample elementary plaquette. The signs inside the circles represent spin orientations, while the values adjacent to the edges represent the couplings.

glass are going to be located relatively close after the transformation. Every bit of the transformed solutions corresponds to a unique coupling between two spins. See Figure 1 for an example of transforming a small system of 2×2 spins (for simplicity, the example does not include periodic boundary conditions).

4.3 From Couplings to Spins

After variation, new candidate solutions encode whether or not the different coupling constraints are satisfied, but they do not directly encode spin values. To find a corresponding spin configuration for a given Boolean vector for coupling satisfiability, we can start by assigning a randomly selected spin to a random value. In each iteration, we assign one of the neighbors of the spins, which have already been assigned, according to the couplings to the already assigned spins. Since we know the sign of the product of each coupling and its two neighbors (elements of the coupling vector), one of the two spins of any coupling can be used to determine the value of the other.

Ideally, we can create a spin configuration that fully agrees with the coupling vector. However, variation can sometimes create coupling vectors that are contradictory and not all constraints (couplings) can be satisfied or unsatisfied as specified in the new candidate solution. For example, for the spin glass instance shown in Figure 1(b), a coupling vector 1011 represents conflicting couplings and regardless of the way we set the spins, the corresponding coupling vector will not be 1011.

To resolve conflicts, we proceed by using a simple greedy algorithm that starts by setting a randomly selected spin to a random value and then in each iteration it sets the spin that will satisfy most of the coupling constraints as specified in the coupling vector. All ties are resolved randomly.

More advanced algorithms can be used for the transformation from coupling vectors to spins.

However, this paper shows that even this simple approach can lead to significant speedups for simple variation operators.

4.4 Algorithm Modifications

The procedures for transforming spin configurations into coupling vectors and the other way around can be incorporated into any evolutionary algorithm in a straightforward manner. The goal of the transformation is to improve the effectiveness of the variation operators. Before applying variation operators, the selected population of candidate solutions represented by spin configurations is transformed into coupling vectors. Variation is applied to the coupling vectors, creating new coupling vectors (with potential conflicts). The new coupling vectors are then mapped back into spin configurations. The transformation of one candidate solution can be completed in $O(n)$ steps (in both directions).

5 Experiments

This section presents and discusses empirical results. First, test problems and experiments are described. Next, the results are presented and discussed.

5.1 Test Instances

We considered 2D Ising spin glass instances with $\pm J$ couplings and periodic boundary conditions. To analyze scalability, instances of size 6×6 ($n = 36$ spins) to 18×18 ($n = 324$ spins) have been considered; 1000 random instances have been used for each problem size. All instances 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 (Dayal, Trebst, Wessel, Würtz, Troyer, Sabhapandit, & Coppersmith, 2004).

5.2 Description of Experiments

Binary tournament selection is used in all algorithms to select promising solutions. RTR is used to incorporate new candidate solutions into the original population; the window size for RTR is $w = \min\{n, N/20\}$ where n is the number of bits and N is the population size (Pelikan & Goldberg, 2001). Population size for each algorithm and problem instance is determined using the bisection method (Sastry, 2001) to ensure that the algorithm finds the global optimum in 5 out of 5 independent runs. Each run is terminated when the ground state has been found.

For each problem size, the results obtained with each compared algorithm are averaged over 1000 random problem instances; the results for each algorithm and each problem size thus correspond to 5000 successful runs.

In GA, two-point crossover is used to recombine pairs of selected solutions and the probability of applying the crossover is $p_c = 0.6$. GA also uses standard bit-flip mutation. By default, the probability of flipping each bit by mutation is set to $p_m = 1/n$ where n is the number of bits in the recombined solutions. Thus, by default, p_m for the coupling-vector representation is smaller than p_m for the original representation by vectors of spin configurations. However, other mutation rates were tested to eliminate potential effects of modifying p_m due to the change in problem size after transforming spin configurations into coupling vectors. Specifically, we also consider the performance of GA without any mutation at all and we consider the performance of GA with the same mutation rate whether or not the problem transformation is used.

To quantify the effects of the proposed problem transformation, we record both the number of evaluations with and without the transformation as well as the speedup, which is defined as the ratio of the number of function evaluations without and with the transformation. The higher the speedup, the more significant the benefits of using the problem transformation.

5.3 Results

Figure 2 shows the influence of the proposed problem transformation on the performance of GA. Regardless of the strategy for setting p_m, the problem transformation leads to significant speedups that grow approximately linearly with problem size. For instances of 256 spins, this leads to speedups of 2 or more, depending on the settings. The number of evaluations required by GA with or without the transformation appears to grow polynomially with problem size; nonetheless, due to the detailed analysis presented elsewhere (Pelikan, Goldberg, Ocenasek, & Trebst, 2003) it can be expected that the performance is slightly worse than polynomial.

Figure 3 shows the influence of the proposed problem transformation on the performance of UMDA. Since UMDA without using the problem transformation was not able to solve instances of 144 spins or more in reasonable time, the speedup is displayed only for problem sizes from 36 to 100 spins. The speedups obtained with UMDA are much more dramatic than those obtained with the GA; already for the problem of size 100, the speedup of more than 720 has been obtained. Furthermore, while UMDA without the problem transformation was practically incapable of solving instances of 144 spins or more, UMDA with the transformed problem is easily able to solve even problems of 256 bits or more. The bad news is that the scalability of both UMDA variants appears to grow exponentially fast with problem size; nonetheless, the growth is much slower with the problem transformation than without it.

Figure 4 shows the influence of the proposed problem transformation on the performance of hBOA. Unlike GA and UMDA, hBOA does *not* benefit from the problem transformation at all; instead, we see that the problem transformation increases the number of function evaluations and the decrease in the performance gets more significant as problem size grows. The reason is that the number of bits of the transformed representation is twice as high as for the original system, while hBOA seems to be capable of obtaining the information gained by the transformation already for the original representation, i.e. without the transformation. Comparing the performance of hBOA without the problem transformation with any other compared algorithm reveals that hBOA retains the best performance among all the algorithms.

To provide an overview of the relative performance of the different algorithms tested in this paper, Figure 5 shows a comparison of the best variants of GA, UMDA and hBOA; both GA and UMDA used the proposed problem transformation while hBOA is applied to the original problem. The comparison clearly shows that hBOA performs best both in terms of the absolute performance as well as the asymptotic growth of the number of function evaluations.

To summarize, the proposed problem transformation leads to speedups for GA and UMDA which range from moderate to very high, whereas for hBOA it actually worsens the performance. The results obtained with GA and especially those obtained with UMDA indicate that the problem transformation significantly simplifies the problem for simple recombination operators that are not capable of learning and exploiting linkage between different problem variables. However, this transformation still does not outperform the automatic model building and sampling procedures of hBOA, which is capable of solving 2D spin glasses faster than GA and UMDA regardless of whether we use the spin-configuration representation of coupling-vector representation.

6 Future Work

This work lead to many important questions. First of all, can we modify the problem transformation so that algorithms like GA and UMDA scale up polynomially and outperform or at least perform as well as hBOA? Succeeding in this challenge could allow us to solve extremely large spin glass instances and other important types of problems that are currently intractable, which is necessary to understand their behavior (Hartmann & Moore, 2003). The second important question is, can we use a similar approach to solve other important types of problems that can be formulated as constraint satisfaction problems, such as MAXSAT? Furthermore, it is important to look at other classes of spin glasses, such as the 3D spin glasses and spin glasses with Gaussian couplings.

One of the important avenues of future research in this area is to study model structure and model complexity used in hBOA when solving spin glasses. Some of the first steps in this area have recently been published (Hauschild, Pelikan, Lima, & Sastry, 2007), but many questions still remain unanswered. Since hBOA is so successful in solving spin glasses, understanding model structure and complexity in hBOA should allow us to better understand the properties of desired representation transformations that could lead to significant speedups. The lessons learned should be applicable to other important classes of difficult problems.

Studying the distribution of the coupling vectors and the models created by hBOA for the populations of coupling vectors corresponding to local optima in the spin glass energy landscape may provide important insights into the problem difficulty and the properties of the energy landscape of this class of problems. Since Ising spin glasses share many difficulties with other challenging, highly multimodal problems (Barahona, Maynard, Rammal, & Uhry, 1982; Hartmann & Rieger, 2001; Hartmann & Rieger, 2004; Pelikan & Goldberg, 2003) the gained insight should be useful in the design of robust optimization methods for solving difficult problems.

Finally, the proposed problem transformation is an efficiency enhancement technique which can be easily combined with other efficiency enhancement techniques (Sastry, 2001; Goldberg, 2002; Pelikan, 2005; Sastry, Pelikan, & Goldberg, 2006) such as parallelization (Cantú-Paz, 2000; Ocenasek, Cantú-Paz, Pelikan, & Schwarz, 2006) and cluster exact approximation (Hartmann, 1996).

7 Summary and Conclusions

This paper proposed a technique for transforming spin configurations to coupling vectors in order to simplify the problem of finding ground states of Ising spin glasses. The proposed method was tested in combination with various evolutionary algorithms, providing thorough empirical evidence that the transformation helps to improve simple variation operators but is unnecessary for advanced variation operators that outperform it. Specifically, we showed that using the proposed problem transformation improves results with standard two-point crossover and the univariate probabilistic recombination of the UMDA algorithm, sometimes by a factor of several hundreds or more. Nonetheless, the performance of hBOA on the original problem remains best both in terms of the absolute complexity as well as the asymptotic complexity with respect to the number of function evaluations until optimum.

There are several important consequences of this work. First of all, the results presented here show that even for difficult combinatorial problems that are intractable by an optimization algorithm, a relatively simple problem transformation may yield previously intractable problem instances tractable, as was the case with UMDA. Even more importantly, the work takes an important step toward designing better and more general approaches to problem transformation and representation adjustment for efficiency enhancement. The results of this work are closely related to the study of model structure and model complexity in multivariate EDAs and should help to reinforce some of the recent findings with additional evidence. Studying the populations of coupling vectors and their probability distribution—which is captured for example by models developed by hBOA with the transformed problem representation—can provide important insight into the properties of the energy landscape in spin glasses. Because of the inherent problem complexity of frustrated Ising spin glasses and because Ising spin glasses share many features with other difficult highly multimodal problems, studying this area should provide important lessons for the design of robust, scalable, and practical optimization techniques for solving challenging problems.

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 Materiel Command, USAF, under grant FA9550-06-1-0096, by the *VolkswagenStiftung* (Germany) within the program "Nachwuchsgruppen an Universitäten", 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.

The U.S. Government is authorized to reproduce and distribute reprints for government purposes notwithstanding any copyright notation thereon. Some experiments were done using the hBOA software developed by Martin Pelikan and David E. Goldberg at the University of Illinois at Urbana-Champaign and most experiments were performed on the Beowulf cluster maintained by ITS at the University of Missouri in St. Louis.

References

Baluja, S. (1994). *Population-based incremental learning: A method for integrating genetic search based function optimization and competitive learning* (Tech. Rep. No. CMU-CS-94-163). Pittsburgh, PA: Carnegie Mellon University.

Barahona, F., Maynard, R., Rammal, R., & Uhry, J. (1982). Morphology of ground states of a two dimensional frustration model. *J. Phys. A*, *15*, 673.

Binder, K., & Young, A. (1986). Spin-glasses: Experimental facts, theoretical concepts and open questions. *Rev. Mod. Phys.*, *58*, 801.

Bosman, P. A. N., & Thierens, D. (2000). Continuous iterated density estimation evolutionary algorithms within the IDEA framework. *Workshop Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000)*, 197–200.

Cantú-Paz, E. (2000). *Efficient and accurate parallel genetic algorithms*. Boston, MA: Kluwer.

Chickering, D. M., Heckerman, D., & Meek, C. (1997). *A Bayesian approach to learning Bayesian networks with local structure* (Technical Report MSR-TR-97-07). Redmond, WA: Microsoft Research.

Dayal, P., Trebst, S., Wessel, S., Würtz, D., Troyer, M., Sabhapandit, S., & Coppersmith, S. (2004). Performance limitations of flat histogram methods and optimality of Wang-Langdau sampling. *Physical Review Letters*, *92*(9), 097201.

Fischer, K., & Hertz, J. (1991). *Spin glasses*. Cambridge: Cambridge University Press.

Friedman, N., & Goldszmidt, M. (1999). Learning Bayesian networks with local structure. In Jordan, M. I. (Ed.), *Graphical models* (pp. 421–459). Cambridge, MA: MIT Press.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley.

Goldberg, D. E. (2002). *The design of innovation: Lessons from and for competent genetic algorithms*, Volume 7 of *Genetic Algorithms and Evolutionary Computation*. Kluwer Academic Publishers.

Harik, G. R. (1995). Finding multimodal solutions using restricted tournament selection. *Proceedings of the International Conference on Genetic Algorithms (ICGA-95)*, 24–31.

Hartmann, A. K. (1996). Cluster-exact approximation of spin glass ground states. *Physica A*, *224*, 480.

Hartmann, A. K. (1998). Are ground states of 3d $\pm j$ spin glasses ultrametric ? *Europhys. Lett.*, *44*, 249.

Hartmann, A. K. (2001). Ground-state clusters of two, three and four-dimensional +/-J Ising spin glasses. *Phys. Rev. E*, *63*, 016106.

Hartmann, A. K., & Moore, M. A. (2003). Corrections to scaling are large for droplets in two-dimensional spin glasses. *Phys. Rev. Lett.*, *90*, 12720.

Hartmann, A. K., & Rieger, H. (2001). *Optimization algorithms in physics*. Weinheim: Wiley-VCH.

Hartmann, A. K., & Rieger, H. (Eds.) (2004). *New optimization algorithms in physics*. Weinheim: Wiley-VCH.

Hauschild, M., Pelikan, M., Lima, C., & Sastry, K. (2007). *Analyzing probabilistic models in hierarchical boa on traps and spin glasses* (MEDAL Report No. 2007001). Missouri Estimation of Distribution Algorithms Laboratory, University of Missouri–St. Louis, St. Louis, MO.

Hed, G., Hartmann, A. K., Stauffer, D., & Domany, E. (2001). Spin domains generate hierarchical ground states. *Phys. Rev. Lett.*, *86*, 3148.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor, MI: University of Michigan Press.

Höns, R. (2005). *Estimation of distribution algorithms and minimum relative entropy*. Doctoral dissertation, University of Bonn, Germany.

Houdayer, J., & Martin, O. C. (1999). Renormalization for discrete optimization. *Phys. Rev. Lett.*, *83*, 1030.

Larrañaga, P., & Lozano, J. A. (Eds.) (2002). *Estimation of distribution algorithms: A new tool for evolutionary computation*. Boston, MA: Kluwer.

Mezard, M., Parisi, G., & Virasoro, M. (1987). *Spin glass theory and beyond*. Singapore: World Scientific.

Mühlenbein, H., & Mahnig, T. (1999). FDA – A scalable evolutionary algorithm for the optimization of additively decomposed functions. *Evolutionary Computation*, *7*(4), 353–376.

Mühlenbein, H., & Paaß, G. (1996). From recombination of genes to the estimation of distributions I. Binary parameters. *Parallel Problem Solving from Nature*, 178–187.

Naudts, B., & Naudts, J. (1998). The effect of spin-flip symmetry on the performance of the simple GA. In Eiben, A. E., Bäck, T., Schoenauer, M., & Schwefel, H.-P. (Eds.), *Parallel Problem Solving from Nature* (pp. 67–76). Berlin Heidelberg: Springer Verlag.

Ocenasek, J., Cantú-Paz, E., Pelikan, M., & Schwarz, J. (2006). Design of parallel estimation of distribution algorithms. Springer.

Pelikan, M. (2005). *Hierarchical Bayesian optimization algorithm: Toward a new generation of evolutionary algorithms*. Springer-Verlag.

Pelikan, M., & Goldberg, D. E. (2001). Escaping hierarchical traps with competent genetic algorithms. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, 511–518. Also IlliGAL Report No. 2000020.

Pelikan, M., & Goldberg, D. E. (2003). Hierarchical BOA solves Ising spin glasses and maxsat. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2003)*, *II*, 1275–1286. Also IlliGAL Report No. 2003001.

Pelikan, M., Goldberg, D. E., & Lobo, F. (2002). A survey of optimization by building and using probabilistic models. *Computational Optimization and Applications*, *21*(1), 5–20. Also IlliGAL Report No. 99018.

Pelikan, M., Goldberg, D. E., Ocenasek, J., & Trebst, S. (2003). *Robust and scalable black-box optimization, hierarchy, and Ising spin glasses* (IlliGAL Report No. 2003019). Urbana, IL: Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign.

Pelikan, M., & Hartmann, A. K. (2006). Searching for ground states of Ising spin glasses with hierarchical BOA and cluster exact approximation. In Cantú-Paz, E., Pelikan, M., & Sastry, K. (Eds.), *Scalable optimization via probabilistic modeling: From algorithms to applications* (pp. ?–?). Springer.

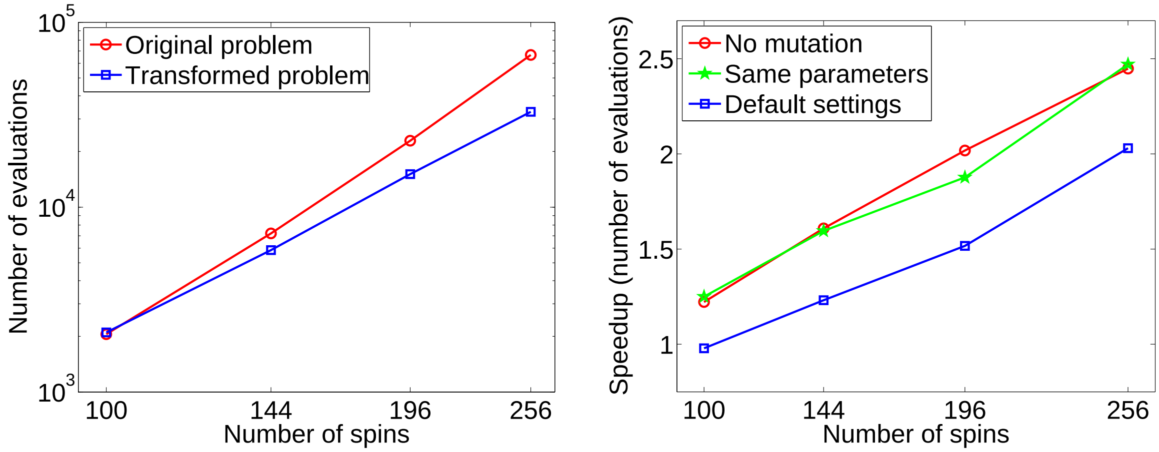
Santana, R. (2005). Estimation of distribution algorithms with Kikuchi approximations. *Evolutionary Computation*, *13*(1), 67–97.

Sastry, K. (2001). *Evaluation-relaxation schemes for genetic and evolutionary algorithms*. Master's thesis, University of Illinois at Urbana-Champaign, Department of General Engineering, Urbana, IL. Also IlliGAL Report No. 2002004.

Sastry, K., Pelikan, M., & Goldberg, D. E. (2006). Efficiency enhancement of estimation of distribution algorithms. In Pelikan, M., Sastry, K., & Cantú-Paz, E. (Eds.), *Scalable Optimization via Probabilistic Modeling: From Algorithms to Applications* (pp. ?–?). Springer.

Shakya, S. K., McCall, J. A., & Brown, D. F. (2006). Solving the Ising spin glass problem using a bivariate EDA based on Markov random fields. *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2006)*, 908– 915.

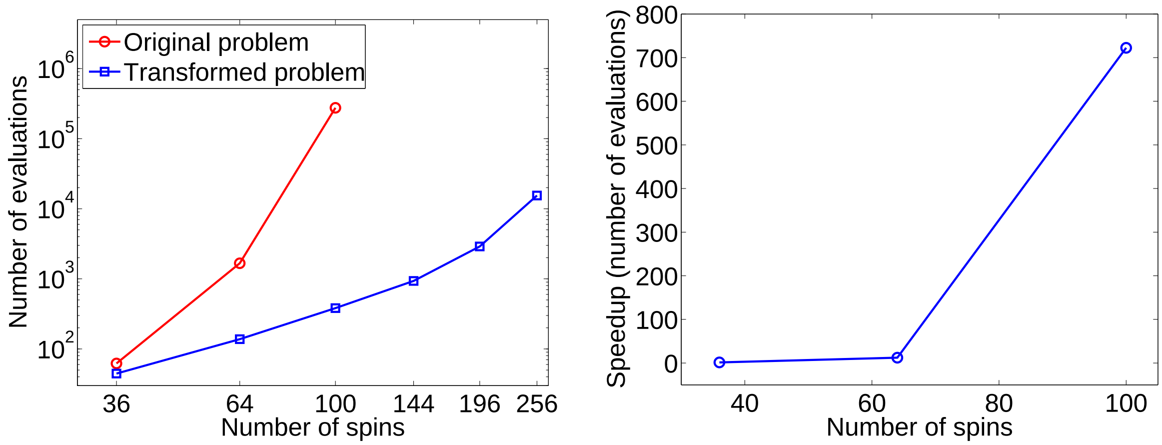
Young, A. (Ed.) (1998). *Spin glasses and random fields*. Singapore: World Scientific.



(a) GA scalability with the default $p_m = 1/n$.

(b) GA speedup.

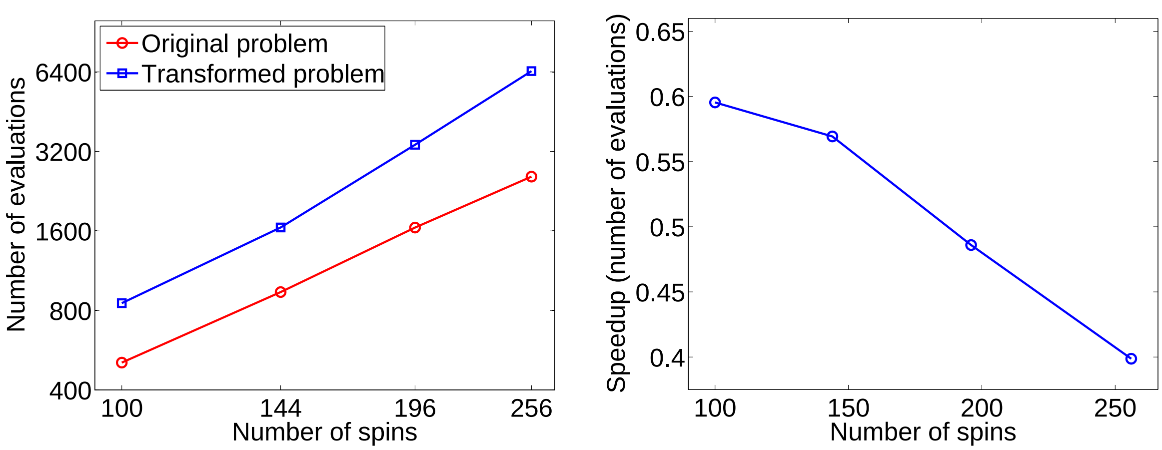
Figure 2: Speedup and scalability of GA (two-point crossover) with various settings of p_m. Lines are guides to the eyes only.



(a) UMDA scalability.

(b) UMDA speedup.

Figure 3: Speedup and scalability of UMDA. Lines are guides to the eyes only.



(a) hBOA scalability.

(b) hBOA speedup.

Figure 4: Speedup and scalability of hBOA. Lines are guides to the eyes only.

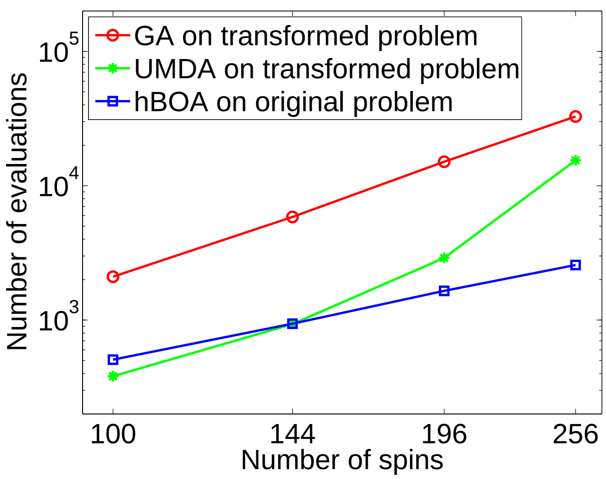


Figure 5: Comparison of the best variants of GA, UMDA and hBOA.