Evolutionary Approaches to Optimization Problems in Chimera Topologies

Roberto Santana
University of the Basque Country (UPV/EHU)
San Sebastián, Spain
roberto.santana@ehu.es

Zheng Zhu
Texas A&M University
College Station, Texas, US
zzwtgts@tamu.edu

Helmut Katzgraber
Texas A&M University
College Station, Texas, US
katzgraber@physics.tamu.edu

ABSTRACT

Chimera graphs define the topology of one of the first commercially available quantum computers. A variety of optimization problems have been mapped to this topology to evaluate the behavior of quantum enhanced optimization heuristics in relation to other optimizers, being able to efficiently solve problems classically to use them as benchmarks for quantum machines. In this paper we investigate for the first time the use of Evolutionary Algorithms (EAs) on Ising spin glass instances defined on the Chimera topology. Three genetic algorithms (GAs) and three estimation of distribution algorithms (EDAs) are evaluated over 1000 hard instances of the Ising spin glass constructed from Sidon sets. We focus on determining whether the information about the topology of the graph can be used to improve the results of EAs and on identifying the characteristics of the Ising instances that influence the success rate of GAs and EDAs.

CCS Concepts

•**Applied computing** → **Physics;** •**Mathematics of computing** → **Discrete mathematics;** *Combinatorial optimization;* •**Computing methodologies** → *Artificial intelligence; Discrete space search;*

Keywords

quantum computing, Ising model, estimation of distribution algorithm, genetic algorithms, instance analysis

1. INTRODUCTION

Quantum computation (QC) aims to profit from the quantum properties of elementary particles to devise new, more efficient ways for representing and manipulating information. A number of paradigms have been proposed as promising candidates for QC. One of these paradigms is adiabatic quantum computing [4, 23] and perhaps the best known implementation of this approach for practical computation is the D-Wave quantum machine [6].

In very simple terms, the D-Wave architecture consists of an array of qubits coupled following a predefined topology (Chimera graph). Qubits have associated weights and each coupler has an associated strength. Current D-Wave quantum annealing processors are designed to minimize the energy of an Ising spin configuration whose pairwise interactions lie on the edges of a Chimera graph. D-Wave computers have served as an excellent testbed for investigating whether QC is actually possible. In particular, many studies have focused on the behavior of the quantum annealing (QA) optimization method [5,13]. QA is similar to simulated annealing (SA) but transitions between states (solutions) do not depend on thermodynamic fluctuations. These transitions are determined by quantum fluctuations.

Although other optimization problems of arbitrary pairwise interaction structure and manageable size can be embedded into the D-Wave topology, a more straightforward approach to study the performance of this QC paradigm is to address problems whose structure has been defined on the Chimera graph. Understanding which are the characteristics of problems defined on Chimera graphs is thus a very important issue. The behavior of a number of algorithms have been investigated for problems with Chimera graph structure. On these problems, QA has been compared to optimization methods such as Tabu search [17], SA [14] and parallel tempering (PT) [28].

In this paper we propose a through investigation of evolutionary algorithms (EAs) for Ising spin glass problems defined on Chimera graphs. Our goal is to determine to what extent the search principles in which EAs are based on can be valuable for the solution of these problems. We argue that EAs can and should be used not only as solvers but also as a tool to better understand the characteristics of relevant problems. In particular, identifying sources of problem difficulty that are common to optimization approaches with completely different search strategies can advance the understanding of the problems and of the optimizers. We show how EAs can be used with these goals in mind and provide evidence of the benefits of this approach.

To study the target problems, we apply hybrid genetic algorithms (GAs) and variants of estimation of distribution algorithms (EDAs) based on factorizations [3, 18, 26]. This strategy allows us to investigate EAs that are blind to the problem structure and other variants that exploit the information about this structure for a more efficient search. In addition to elucidate which is the difficulty that Chimera-graph based spin glass instances pose to EAs, we expect to obtain insights from the behavior of EAs on these in-

GECCO '16, July 20-24, 2016, Denver, CO, USA
© 2016 ACM. ISBN 978-1-4503-4206-3/16/07. . . $15.00
DOI: http://dx.doi.org/10.1145/2908812.2908914

stances. To the knowledge of the authors problems defined on Chimera graphs have not been previously addressed using EAs.

The paper is organized as follows: The next section presents some necessary background on Chimera graphs and the Ising model. Section 3 discusses related work on the relevant questions treated in the paper. Section 4 presents the EAs used to optimize the instances. Section 5 introduces the experimental framework used to investigate our hypothesis, presents the numerical results and discusses our findings. Section 6 concludes the paper.

2. BACKGROUND

2.1 Ising model

A Hamiltonian function serves to describe how the energy of a physical process depends on the state of the system's particles and their interactions. The generalized Ising model is described by the Hamiltonian shown in Equation (1) where L is the set of sites called a lattice. Each spin variable σ_i at site $i \in L$ either takes the value 1 or -1. One specific choice of values for the spin variables is called a configuration. The constants J_{ij} are the interaction coefficients or couplings. The ground state is the configuration with minimum energy.

$$H = -\sum_{<i,j> \in L} J_{ij}\sigma_i\sigma_j - \sum_{i \in L} h_i \sigma_i \qquad (1)$$

where the first sum is over pairs of spins adjacent in the lattice L.

Spin couplings can take values from an a priori defined set of values and both the choice of the set and the distribution with which the couplings are sampled from it influence the characteristics of the problem and its difficulty for the optimizers. In this paper we use couplings selected from Sidon sets of the type $U_{5,6,7} = \{\pm 5, \pm 6, \pm 7\}$ as advocated in [14]. The rationale behind this choice of the couplings is explained in Section 5.1. We consider instances for which $h_i = 0,\ \forall i \in L$.

2.2 D-Wave architecture and Chimera graphs

The D-Wave architecture follows a regular repeating pattern to tile out a processor [6]. The building blocks of the Chimera graph are symmetric subgraphs referred to as $K_{4,4}$ graphs. The eight qubits included in the $K_{4,4}$ graph are connected as bipartite graphs. One example of two connected $K_{4,4}$ graphs is shown in Figure 1. In addition to bipartite connections, in each group, the four left nodes are also connected to their respective north/south grid neighbors and the 4 right nodes are connected to their east/west neighbors. Thus internal nodes have degree 6 and boundary nodes have degree 5. The Chimera graph is formed by n^2 $K_{4,4}$ subgraphs.

3. RELATED WORK

The questions treated in this paper are related to different lines of research in the fields EAs and also to studies on the behavior of different types of optimizers for Chimera graphs. Therefore, we organize the review of previous work according to these relevant topics.

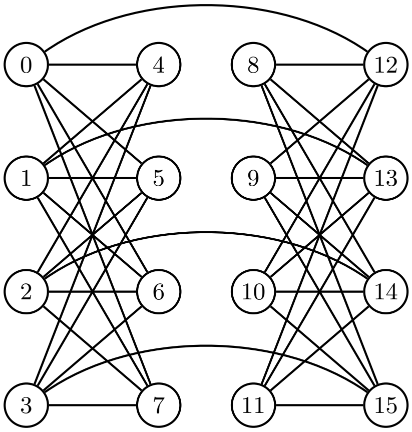


Figure 1: Detail of a Chimera graph. Two connected $K_{4,4}$ blocks.

3.1 Previous EA approaches to Ising and other problems from Physics

Ising spin glasses are not only relevant because of the role they play in Physics. They represent a formidable problem to investigate the capabilities of different EAs and as such they have been investigated in a number of papers.

GAs were early applied to Ising models [1, 15, 21]. Some of these works used knowledge about the topology of the problem to design the genetic operators. In [1], a block-crossover, able to exploit the 2-dimensional structure of the instances was proposed. EDAs have been also used to solve Ising models [19, 22, 25]. In [19], the Bayesian optimization algorithm (BOA) was applied to instances with $\{\pm 1\}$ and Gaussian couplings. A deterministic hill climber based on single-bit flips was used to solve problems of up to 400 spins. To solve bigger instances of 512 and 1024 spins, a more sophisticated cluster exact approximation method was required. In general, successful EDAs applications to large Ising models are in fact hybrid algorithms that incorporate local search methods.

While Ising problems addressed with EAs have been generally defined on regular 2-d and 3-d grids, other topologies have been also used. In [20], the behavior of EAs was investigated on one-dimensional spin glass with power-law interactions. In [9], the impact of different network topologies (e.g. grids, small-world networks and random graphs) of the underlying problem on the models learned by EDAs was investigated. We did not find previous reports on the application of EAs to Ising models defined on Chimera graphs. It is important to notice that the Chimera topology considerably departs from regular grids.

3.2 Exploiting the problem structure

Using problem information to improve EAs has been always a relevant issue in evolutionary computation (EC). One relevant question is whether and how knowledge about the underlying topology of the optimization problem can be used to increase the efficiency of the EA approaches to this problem. In this paper, we investigate this issue in three different ways: 1) Using the Chimera structure to design building-block wise crossover operators where the building blocks corresponds to groups of variables in the same $K_{4,4}$ subgraph. 2) Using factorized EDAs in which the factors comprise variables in each $K_{4,4}$ subgraph. 3) Biasing the search for tree-based EDAs to include dependencies between variables that are connected in the Chimera structure.

Building-block wise crossover is at the core of successful probabilistic model building GAs like the extended compact GA [11]. EDAs that explicitly use to different extent the regular grid structure of the Ising problems have been previously proposed in [22, 25]. EDAs that bias the learning of probabilistic trees using a priori information about the problem has shown to be more efficient [2].

Other methods that use information about the structure of the graphs to solve Ising problems defined on Chimera graphs have been also proposed in the field of Physics. In [29], a heuristic method that finds optimal configurations of local clusters of spins as a way to reach the ground state is presented. The target clusters are those that are strongly coupled to each other and more weakly coupled to the rest of the system. Although the factorized EDA we use in our comparisons use information about clusters of solutions, these clusters depend on the Chimera graph and not on the strength of the couplings.

The Hamze-de Freitas-Selby (HFS) algorithm [10,24], uses a subgraph-based sampling method for Ising-type models with frustration. A collection of subgraphs induced by the original topology are used instead of single spins for a more efficient sampling with GS and PT. In Selby [24], experiments on Chimera graphs were done using trees as the induced subgraphs. In this paper, we use EDAs based on tree models. However, the trees are learned from the selected solutions by applying statistical methods.

3.3 Fitness landscape analysis and investigation of fitness difficulty

A number of papers [7,17] have investigated the behavior of classical optimizers on Ising instances defined on Chimera graphs. In [17], three conventional software solvers: CPLEX, a variant of Tabu, and a branch-and-bound search, were compared to QA in Chimera-structured problems.

In [12], different variants of SA were evaluated on Ising problems defined on Chimera graphs. However, the focus of the paper was not on the comparison of SA with other algorithms but on the development of fast SA implementations. In [14], an extensive comparison between SA and QA is presented for Ising problems with Chimera topology. This work identified the important effect that the choice of the couplings for the Ising instances could have in the comparison between optimizers for problems defined on Chimera-graphs. Instances generated using Sidon sets[1] were proposed as harder to optimize. Also in [28], a particular class of Sidon instances ($J_{ij} = \{5, 6, 7\}$) was used as a test-bed for evaluating the behavior of QA which was compared to PT.

4. EVOLUTIONARY OPTIMIZATION APPROACHES

Let $\mathbf{X} = (X_1, \ldots, X_n)$ denote a vector of discrete random variables where n is the number of nodes of the Chimera graph. We use $\mathbf{x} = (x_1, \ldots, x_n)$ to denote an assignment to the variables. In our problem representation, each binary variable represents one spin configuration of a node in the Chimera graph. In the chromosome representation, variables are ordered according to the position of the nodes in the Chimera graph. For instance, the first 16 variables would correspond to the nodes shown in Figure 1. Therefore, the variables that represent nodes that are connected in the Chimera graph are, in most of the cases, relatively close in the chromosome. The fitness function used to evaluate the solutions is the one represented by Equation (1).

4.1 Benchmarked algorithms

Algorithm 1 shows the general pseudocode of all the EAs tried in the paper. Population size $N = 512$ and truncation selection with parameter $T = 0.5$ were used.

Algorithm 1: **EA**

```
Set $t \Leftarrow 0$. Generate an initial population $D_0$ of $N \gg 0$ random solutions.
do {
    For each solution, apply a greedy local search method and output local optimum.
    Select from population $D_t$ a set $D_t^S$ of $k \leq N$ points using truncation selection.
    Generate a new population $D_{t+1}$ from $D_t^S$ applying the variator operator of choice.
    Apply random bit-flip mutation to solutions in $D_{t+1}$.
    $t \Leftarrow t + 1$
} until Termination criteria are met.
```

In the design of EAs, certain assumptions about the structure of the optimization problem are usually implicitly or explicitly made. The extent to which these assumptions agree with the characteristics of the problem usually determine the success of the EA. We have selected six EAs by considering how their mechanisms to explore the space of solutions may be related with the structure of the Chimera graphs.

1. GA with 1-point crossover (1PCX-GA): Simple GA algorithm with a crossover probability of 1 and where a crossover point is randomly selected between positions 2 and $n - 1$. The two offspring are created by taking one segment from each parent.

2. GA with uniform CX (uCX-GA): Similar to 1PCX-GA but the alleles of the offspring are randomly taken from each of the two parents with probability 0.5.

3. GA with bit-wise CX (BWCX-GA): Similar to uCX-GA but instead of single bits, blocks of variables are taken from each parent. These blocks correspond to variables that are related by the Chimera graph. In particular, there is one block for each subgraph $k_{4,4}$ involving 8 variables and for each edge joining these subgraphs.

4. Factorized distribution algorithm (FDA): An EDA that uses the same structure as BWCX-GA. Marginal probabilities for all configurations of the blocks are learned from the selected solutions and new solutions are generated sampling from a junction tree [18] constructed from this factorization.

[1] In a Sidon set, the sum of two members of the set gives a number that is not part of the set

5. Tree-based EDA (Tree-EDA): An EDA that learns the structure of the graphical model from the matrix of mutual information between *all possible pairs* of variables [3]. The pairs with the strongest mutual information are used to build a tree structure from which solutions are sampled.

6. Tree-based EDA with a priori information (Tree-EDA^r): Idem to Tree-EDA but only pairs of variables connected in the Chimera topology are used to learn the probabilistic model.

4.2 Efficient local search

In step 3 of Algorithm 1 a greedy local search method was added as part of the evaluation process. Previous works [19, 22] have shown that without the inclusion of local optimizers, EAs are not expected to be competitive algorithms for large Ising spin glass problems.

The details of the greedy local search implemented are shown in Algorithm 2. One main characteristic of the method is that the local fitness value of the current solution are stored in memory and only local computations are needed to evaluate the effect of bit flips. Therefore, only a fraction of computations are needed to evaluate the n possible bit-flips possible from the current solution. In addition, when the initial solution is a local optimum and no bit-flip improves the current fitness value, the local optimizer allows a predefined number of transitions (5) to solutions with lower fitness. In these cases, the bit-flip that decreases the fitness the least is selected. This mechanism, similar to one of the components of Tabu search was designed as a way to help the algorithm to escape from local optima. However, the local optimizer does not implement a memory or any other type of more sophisticated components. The local optimizer stops when the current solution can not be improved or the maximum number of transitions to solutions of lower fitness has been consumed.

Algorithm 2: **Greedy search**

```
Set current solution to initial solution.
Initialize number of accepted negative $n_{neg\_moves} = 0$.
Evaluate the 2 possible assignments for the $n$ variables of the initial solution keeping other variables fixed.
do {
    Select the variable $X_i$ whose bit-flip improves the fitness the most
    If improvement is negative, initial solution was already improved or $n_{neg\_moves} = max_{neg\_moves}$
        Return current best solution
    Else if improvement is negative, initial solution has not been improved, and $n_{neg\_moves} < max_{neg\_moves}$.
        $n_{neg\_moves} \rightarrow n_{neg\_moves} + 1$
    Bit-flip $X_i$ in current solution.
    Update local fitness values for variable $X_i$ and all its neighbors in the Chimera graph.
} while
```

After solutions have been evaluated a random bit-flip mutation operator was applied. Extensive preliminary experiments showed that this way of infusing diversity in the population was a required ingredient for avoiding early convergence to poor solutions. More remarkably, a high mutation rate of $p_m = 0.2$ showed to produce the best results. Notice, that all the EAs are compared in the same conditions. The only difference between the implementation of the six EAs is in step 5 of Algorithm 1. This step comprises the recombination operator for the GAs and the learning and sampling step for EDAs. The termination criteria for all algorithms was a maximum of 1000 generations or reaching a low diversity in the population (10 or less genotypically different individuals).

4.3 Structural hypothesis and EAs

The choice of the EAs has been made with the aim of evaluating different questions about the best EA approach to problems with a Chimera structure. We briefly state these questions and their relation to the algorithms.

- How relevant is the impact of potential building block disruption? 1PCX-GA and uCX-GA differ only in the type of crossover mechanism they use. Uniform crossover is a more disruptive operator. By evaluating the difference in the performance of these two algorithms we can have a rough idea of the importance of respecting the interactions of the problem.

- Which is the added gain of considering the underlying structure of the Chimera graph in the design of the crossover operator? By comparing BWCX-GA to 1PCX-GA we can measure if a more informed choice of the information to exchange between the parents has a strong effect in the performance of the GAs.

- Can structured-informed factorized approximations of the selected solutions make a difference over "blind" and more intelligent crossover operators? By comparing the FDA, which uses the blocks of variables related in the Chimera graph as factors, to all other GAs we can determine if using problem information the way EDAs do makes a difference for this problem.

- Should EDAs learn the structure of the interactions from the data instead of inferring them from the Chimera structure? Are pair-wise interactions sufficient to solve the problem? By comparing Tree-EDA to FDA and GAs we can find answers to these questions.

- Are the pair-wise interactions determined by the Chimera graph sufficient to solve the problem? The comparison between the performance of Tree-EDA and Tree-EDA^r will help to answer this question.

5. EXPERIMENTS

The goal of the experiments is twofold. 1) We want to evaluate the difference in the behavior of the evolutionary algorithms and whether these differences offer clues about the characteristics of the Chimera instances. 2) We would like to identify which characteristics of the instances (descriptors) have an impact in the performance of EAs.

5.1 Experimental benchmark

To evaluate the behavior of the algorithms, 1000 Ising instances defined on the Chimera graph were used. These instances were proposed in [14,28] and investigated to evaluate

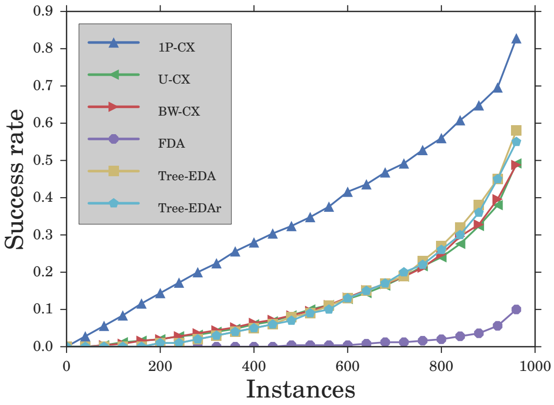


Figure 2: EAs success rate for the 1000 instances considered. For each algorithm, instances have been sorted from lowest to highest success rate.

the behavior of QA in the D-Wave quantum annealer. Instances were generated with couplings defined in the $U_{5,6,7} = \{\pm5, \pm6, \pm7\}$ Sidon set.

There are a number of reasons why the Sidon set instances have been particularly useful for studies in Physics [14,27,28] and relevant for us: 1) All Instances have unique (two if spin reversal symmetry is considered) ground states and all spins have nonzero local fields (by carefully selecting the combination of bonds), these features make instances harder than random bi-modal instances. 2) With Sidon interactions, there is full control of the number of ground states and low lying excited sates, even energy gaps, so it is possible to study resilience of instances with more precision. 3) With full control of many parameters of the instances, it would be possible to identify the important ones that are related to the behavior of different algorithms.

For these instances, the ground states and first exited states with equal probabilities have been found by applying the isoenergetic cluster algorithm [27] and running millions of Monte Carlo Sweeps for each instance. Recent results [16] show that the isoenergetic cluster algorithm is one of the few algorithms that scales better than quantum annealing on Google instances [8].

We use a number of descriptors that characterize the instances. These descriptors are derived from an analysis of the landscape of the spinglass order parameter distribution $P(q)$ for the instances. $P(q)$ overlap distributions is the proxy to the complexity of energy landscape of an instance. It measures the average Hamming distance between two solutions randomly sampled from the problem's low-temperature Boltzmann distribution. This approach can be used to identify the instances with tall and thin energy barriers [16]. We considered the following descriptors:

1. GRstate: Fitness value of the ground state solution.
2. dJ+dh_RES: Resilience of instance to coupler and qubit noise (upper bound success probability). Quantifies the robustness of ground-state configurations to noise in the D-Wave device.
3. PEAK_1, PEAK_2: Position of the highest and second highest peaks in the $P(q)$ distribution plot.

Table 1: Number of instances (out of 1000) for which the ground state was found in r% of the runs. For Tree-EDA and Tree-EDA^r, 100 runs were executed. For all other algorithms, 250.

r (%)	1P-CX	U-CX	BW-CX	FDA	Tree	Tree-r
0.25	991	935	940	543	821	832
1	982	881	878	322	821	832
10	868	480	471	45	474	462
25	645	188	199	4	218	212
50	274	37	38	1	65	63
75	54	6	4	0	12	13
90	13	1	0	0	3	2

4. HEIGHT_1,HEIGHT_2: Height of the highest and second highest peaks in the $P(q)$ distribution plot.
5. H_SCORE: Ratio between the height of second highest peak and the height of highest peak.
6. NOISE_SCR: Ratio between the height of third highest peak and the height of second highest peak.

250 runs were executed for all the algorithms except for Tree-EDA and Tree-EDA^r for which, due to the high computational time spent by the algorithms, only 100 runs were executed.

5.2 Behavior of EAs

Table 1 shows the number of instances for which the ground state was found in r percentage of the runs. For $r = 0.25$, the table shows the number of instances whose optima were found at least once in all the runs. It can be seen that the best results were achieved by GAs over EDAs, with the best absolute results achieved by 1PCX-GA. These results are better detailed in Figure 2 where instances are sorted according to the success rate reached by each algorithm. BWCX-GA and uCX-GA have a similar behavior, while these algorithms are more efficient than Tree-EDA and Tree-EDA^r for finding the ground state 10% of the time or less frequently, the EDAs find more instances with 25% and higher success rate, a fact that can be also observed in Table 1.

The success rate of the algorithms does not provide the whole picture about their behavior since some EAs may exhibit a high variability being able to achieve high quality or poor solutions depending on the instances. Therefore, for each instance, we applied a multiple comparison statistical test to look for significant differences between algorithms using the best solutions reached in 100 runs (i.e. not the number of times that the optimum was found in these runs). The Kruskal Wallis test was applied first, and by applying a posthoc test we looked for statistical differences between each pair of algorithms. A Bonferroni correction was added to compensate for multiple comparisons. All tests used as pvalue $\alpha = 0.01$.

Table 2 summarizes the results of the pair-wise tests for all instances. In the table, cell (r, c) indicates the number of instances for which algorithm in row r was significantly better than algorithm in column c. For example, algorithm 1PCX-GA was significantly better than FDA for the 1000 instances. From the analysis of Table 2 it is clear that 1PCX-GA significantly outperforms all other algorithms. There are not significant differences between the pair of algorithms (uCX-GA,BWCX-GA) and the pair (Tree-EDA,Tree-EDA^r). This

Table 2: Summary of the pair-wise statistical tests. Cell (r, c) indicates the number of instances for which algorithm in row r was significantly better than algorithm in column c.

Alg.	1P-CX	U-CX	BW-CX	FDA	Tree	Tree-r
1P-CX	-	862	858	1000	671	707
U-CX	0	-	0	973	33	31
BW-CX	0	0	-	974	39	41
FDA	0	0	0	-	0	0
Tree	2	201	196	980	-	0
Tree-r	2	182	173	971	0	-

seems to indicate that using information about the structure of the problem does not provide any advantage for the search, at least this is the case for GA with uniform crossover and Tree-EDA. Also, a conclusion from the analysis is that FDA is not a good choice for this problem.

Figure 3 shows the average computational time of all the algorithms across all instances. It can be appreciated that 1PCX-GA is also the fastest among all the EAs compared. It is slightly faster than the other two GAs because uniform crossover requires the systematic generation of random numbers during the crossover and in 1PCX-GA only one random number has to be generated for each crossover. All EDAs require a higher computational time than any GA. In particular Tree-EDA is approximately 8 times slower than 1PCX-GA. As expected, since Tree-EDAs need to learn the structure of the model from data, they are more computationally costly than FDA. Also, there is a clear gain in efficiency in Tree-EDA^r over Tree-EDA. This gain is due to considering for the construction of the tree only the pairwise relationships that exist in the Chimera graph.

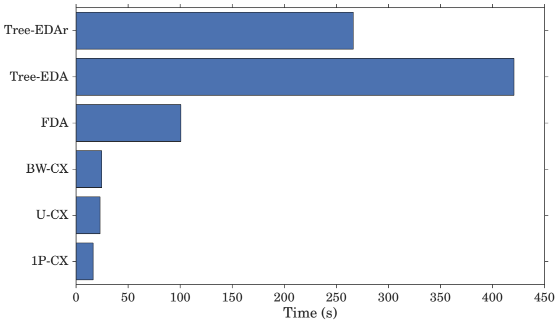


Figure 3: Average computational time of the algorithms in solving all the instances.

5.3 Relationship with instance descriptors

Another important question is to determine whether and how is the performance of the EAs related to the characteristics of the instances. This issue is particularly relevant since the Sidon instances were originally engineered to investigate the behavior of QA on the D-Wave computers. Unveiling this type of relationship could help to find links between the behavior of EAs and optimizers that use completely different search mechanisms. In addition, by investigating this relationship we can determine whether the instance descriptors have a similar impact on all EAs or some of the descriptors are better signatures of behavior for some algorithms than for others.

Table 3: Correlation between the descriptors of the instance and success rate of the EAs. Correlations that were *not* found significant for $p = 0.01$ are underlined.

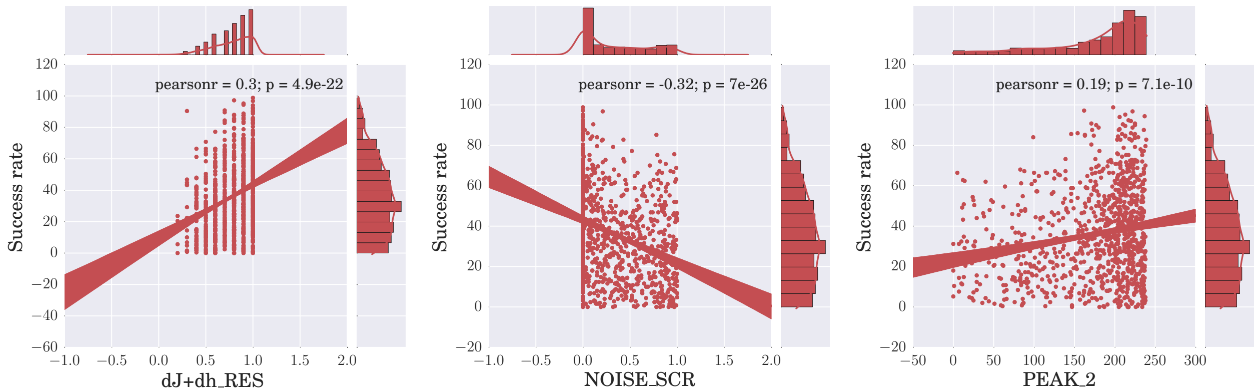
Descriptors	1P-CX	U-CX	BW-CX	FDA	Tree	Tree-r
GRstate	-0.21	-0.17	-0.17	-0.14	-0.17	-0.18
dJ+dh_RES	0.30	0.22	0.22	$\underline{0.07}$	0.16	0.16
PEAK_1	0.16	0.11	0.12	$\underline{0.03}$	0.12	0.12
HEIGHT_1	0.20	0.17	0.17	0.08	0.19	0.18
PEAK_2	0.19	0.13	0.12	-0.00	$\underline{0.07}$	$\underline{0.07}$
HEIGHT_2	$\underline{0.05}$	$\underline{0.02}$	$\underline{0.02}$	0.00	$\underline{-0.06}$	$\underline{-0.04}$
H_SCORE	-0.15	-0.13	-0.13	$\underline{-0.07}$	-0.17	-0.15
NOISE_SCR	-0.32	-0.27	-0.27	-0.09	-0.20	-0.19

We computed the Pearson's correlations between the success rates of EAs and the descriptors of the instances. This information is shown in Table 3 where only a small number of correlations were not statistically significant using $\alpha = 0.01$. Most of them involve algorithm FDA or descriptor $HEIGHT\_2$. The strongest correlations were found for descriptors dJ+dh_RES and NOISE_SCR. Figure 4 shows the patterns of the relationships between these descriptors and the success rate of 1PCX-GA for all instances. Also included in Figure 4 is the relationship for descriptor $PEAK\_2$, which showed significant correlations for all GAs but no significant correlations for any of the EDAs. The strong correlation between success rates and dJ+dh_RES seems to suggest that the more the first excited states (low resilience), then there will be more local minima where the algorithm will likely get trapped. Thus these instances are harder. Also the strong anticorrelation between success rates and NOISE_SCR suggests that instances with multiple peaks (high NOISE_SCR) are typically harder than the ones with one peak.

5.4 Comparison with Simulated Annealing

As a final step we compared the results of the EAs to results achieved by SA. We ran two variants of SA proposed in [12] and named $an\_ss\_rn\_fi$ and $an\_ss\_ge\_nf\_bp$. Both variants are implemented for spin glasses with fixed number of neighbors but $an\_ss\_ge\_nf\_bp$ is conceived to take advantage of the structure bipartite graphs. The two variants were run with the same parameters: Number SA sweeps 2000, $\beta_0 = 0.1$, $\beta_1 = 3$, where $\beta = \frac{1}{T}$ is the inverse of the temperature, and β_0 and β_1 are the initial and final parameters of the linear schedule used for annealing. As in the case of the EAs these parameters are not expected to be optimal for all instances but we checked that increasing the number sweeps did not improve the results significantly. For each Ising instance, 10000 repetitions of the algorithm were executed and from these runs we computed the success rate.

Results for $an\_ss\_rn\_fi$ were very poor and therefore we present results here for $an\_ss\_ge\_nf\_bp$. Figure 5 shows the correlation between the success rate of SA and 1PCX-GA. SA achieved a success rate above 0.25 for only 51 instances (versus 991 for 1PCX-GA). However, it was able to find the optimum for all the instances at least once in 10000 repetitions. SA is also orders of magnitude faster than EAs. It is very difficult to compare both optimizers since the way they



a) b) c)

Figure 4: Pearson's correlation and linear approximation between the success rate of algorithm 1PCX-GA and three descriptors of the instances: a) $dJ + dh\_RES$**; b)** $NOISE\_SCR$**; c)** $PEAK\_2S$**.**

apply partial evaluation and combine it with full evaluation of the solutions differ between the algorithms. Our results show that GAs are at least competitive with SA. More importantly, the analysis of Figure 5 reveals that while there is a correlation between the hardness of the instances for both optimizers, 1PCX-GA exhibits a wider variability in its behavior and the sources of difficulty for both methods are not completely the same. Further experimental work is needed, using other problem benchmarks to assess the differences in the behavior between SA and 1PCX-GA.

6. CONCLUSIONS

In this paper we have investigated for the first time the behavior of EAs on problems defined of the Chimera instances used by D-Wave architectures. We have shown that a simple GA with one-point crossover is able to solve 991 of the 1000 instances considered although the success rate of the algorithm depends on the instances. Our results show that EAs that use probabilistic modeling of the solutions dot not produce an improvement over methods that do not incorporate any type of modeling. To some extent this was an unexpected result because different variants of EDAs had shown good results for Ising problems defined on other topologies. Using problem information does not provide improvements in terms of success rate, although by restricting the number pair-wise dependencies to the edges of the Chimera graph important gains in terms of computational time were achieved by Tree-EDAr over Tree-EDA.

We have also identified a number of instance descriptors which are correlated with the behavior of the algorithms. This could serve as a first step for a more complete characterization of the impact that certain features of the Ising instances have in the performance of EAs. An analysis of the impact of the same features for other optimizers could help to understand how different methods explore the space of solutions to identify the optimum.

Finally, although constructing instances whose pattern of interactions match that of the D-Wave architecture is a sensible way to investigate the performance of QA on the spin glass problem, it is not completely clear how the topological restrictions can affect the complexity of the instances and bias the behavior of other optimization algorithms applied to these instances. Our work could be used to advance the understanding of this and related issues.

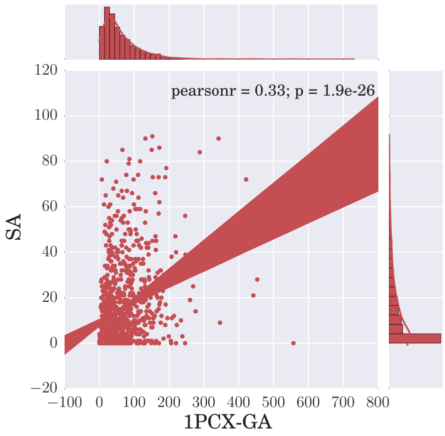


Figure 5: Success rate of 1PCX-GA versus SA variant ($an\_ss\_ge\_nf\_bp$**) designed for bipartite lattices [12]. The number of successful runs is shown for each algorithm.** 250 **runs were executed for 1PCX-GA and** 10000 **for SA.**

7. ACKNOWLEDGMENTS

R. Santana's work has received support by the IT-609-13 program (Basque Government) and TIN2013-41272P (Spanish Ministry of Science and Innovation). H.G.K acknowledges support from the National Science Foundation (Grant No. DMR-1151387). The work of H.G.K. is supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via MIT Lincoln Laboratory Air Force Contract No. FA8721-05-C-0002. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purpose notwithstanding any copyright annotation thereon.

8. REFERENCES

[1] C. A. Anderson, K. F. Jones, and J. Ryan. A two-dimensional genetic algorithm for the Ising problem. *Complex Systems*, 5(3):327–334, 1991.

[2] S. Baluja. Incorporating a priori knowledge in probabilistic-model based optimization. In M. Pelikan, K. Sastry, and E. Cantú-Paz, editors, *Scalable Optimization via Probabilistic Modeling: From Algorithms to Applications*, Studies in Computational Intelligence, pages 205–222. Springer, 2006.

[3] S. Baluja and S. Davies. Using optimal dependency-trees for combinatorial optimization: Learning the structure of the search space. In *Proceedings of the 14th International Conference on Machine Learning*, pages 30–38, 1997.

[4] J. Brooke, D. Bitko, R. T. F., and G. Aeppli. Quantum annealing of a disordered magnet. *Science*, 284(5415):779–781, 1999.

[5] B. K. Chakrabarti and A. Das. Transverse Ising model, glass and quantum annealing. In *Quantum Annealing and Other Optimization Methods*, pages 1–36. Springer, 2005.

[6] E. D. Dahl. Programming with D-Wave: Map coloring problem. White paper, DWave. The Quantum Computing Company, 2013.

[7] S. Dash. A note on QUBO instances defined on Chimera graphs. *arXiv preprint arXiv:1306.1202*, 2013.

[8] V. S. Denchev, S. Boixo, S. V. Isakov, N. Ding, R. Babbush, V. Smelyanskiy, J. Martinis, and H. Neven. What is the computational value of finite range tunneling? *arXiv preprint arXiv:1512.02206*, 2015.

[9] C. Echegoyen, A. Mendiburu, R. Santana, and J. A. Lozano. Estimation of Bayesian networks algorithms in a class of complex networks. In *Proceedings of the 2010 Congress on Evolutionary Computation CEC-2010*, Barcelone, Spain, 2010. IEEE Press.

[10] F. Hamze and N. de Freitas. From fields to trees. In *Uncertainty in Artificial Intelligence (UAI)*, pages 243–250, Arlington, Virginia, 2004. AUAI Press.

[11] G. R. Harik, F. G. Lobo, and K. Sastry. Linkage learning via probabilistic modeling in the ECGA. In M. Pelikan, K. Sastry, and E. Cantú-Paz, editors, *Scalable Optimization via Probabilistic Modeling: From Algorithms to Applications*, Studies in Computational Intelligence, pages 39–62. Springer, 2006.

[12] S. V. Isakov, I. N. Zintchenko, T. F. Rønnow, and M. Troyer. Optimised simulated annealing for Ising spin glasses. *Computer Physics Communications*, 192:265–271, 2015.

[13] T. Kadowaki and H. Nishimori. Quantum annealing in the transverse Ising model. *Physical Review E*, 58(5):5355, 1998.

[14] H. G. Katzgraber, F. Hamze, Z. Zhu, A. J. Ochoa, and H. Munoz-Bauza. Seeking quantum speedup through spin glasses: The good, the bad, and the ugly*. *Phys. Rev. X*, 5:031026, Sep 2015.

[15] A. Maksymowicz, J. Galletly, M. Magdon, and I. Maksymowicz. Genetic algorithm approach for Ising model. *Journal of magnetism and magnetic materials*, 133(1):40–41, 1994.

[16] S. Mandrà, Z. Zhu, W. Wang, A. Perdomo-Ortiz, and H. G. Katzgraber. Strengths and weaknesses of weak-strong cluster problems: A detailed overview of state-of-the-art classical heuristics vs quantum approaches. *arXiv preprint arXiv:1604.01746*, 2016.

[17] C. C. McGeoch and C. Wang. Experimental evaluation of an adiabiatic quantum system for combinatorial optimization. In *Proceedings of the ACM International Conference on Computing Frontiers*, page 23. ACM, 2013.

[18] H. Mühlenbein, T. Mahnig, and A. Ochoa. Schemata, distributions and graphical models in evolutionary optimization. *Journal of Heuristics*, 5(2):213–247, 1999.

[19] 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*, Studies in Computational Intelligence, pages 333–349. Springer, 2006.

[20] M. Pelikan and H. G. Katzgraber. Analysis of evolutionary algorithms on the one-dimensional spin glass with power-law interactions. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pages 843–850. ACM, 2009.

[21] A. Prügel-Bennett and J. L. Shapiro. The dynamics of a genetic algorithm for simple random Ising systems. *Physica D: Nonlinear Phenomena*, 104(1):75–114, 1997.

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

[23] G. E. Santoro, R. Martoňák, E. Tosatti, and R. Car. Theory of quantum annealing of an Ising spin glass. *Science*, 295(5564):2427–2430, 2002.

[24] A. Selby. Efficient subgraph-based sampling of Ising-type models with frustration. *arXiv preprint arXiv:1409.3934*, 2014.

[25] S. Shakya, J. McCall, and D. Brown. Solving the Ising spin glass problem using a bivariate EDA based on Markov random fields. In *Proceedings of the IEEE Congress on Evolutionary Computation. CEC-2006*, pages 908–915. IEEE, 2006.

[26] S. Shakya and R. Santana, editors. *Markov Networks in Evolutionary Computation.* Springer, 2012.

[27] Z. Zhu, A. J. Ochoa, and H. G. Katzgraber. Efficient cluster algorithm for spin glasses in any space dimension. *Phys. Rev. Lett.*, 115:077201, 2015.

[28] Z. Zhu, A. J. Ochoa, S. Schnabel, F. Hamze, and H. G. Katzgraber. Best-case performance of quantum annealers on native spin-glass benchmarks: How chaos can affect success probabilities. *arXiv preprint arXiv:1505.02278*, 2015.

[29] I. Zintchenko, M. B. Hastings, and M. Troyer. From local to global ground states in Ising spin glasses. *Physical Review B*, 91(2):024201, 2015.