Efficient EDA for Large Opimization Problems via Constraining the Search Space of Models

Elham Salehi, Robin Gras
School of Computer Science,
University of Windsor
Windsor, ON, N9B 3P4
{salehie, rgras} @uwindsor.ca

ABSTRACT
Introducing efficient Bayesian learning algorithms in Bayesian network based EDAs seems necessary in order to use them for large and complex problems. In this paper we propose an algorithm, called CMSS-BOA, which uses a recently introduced heuristic called max-min parent children (MMPC) [3] in order to constraint the models search space. This algorithm does not consider a fix and small upper bound on the order of interaction between variables and is able solve problems with large number of variables efficiently. We compare the efficiency of CMSS-BOA with standard Bayesian network based EDA for solving several benchmark problems.

Categories and Subject Descriptors
I.2.6 [**Artificial Intelligence**]: Learning; I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search

General Terms
algorithms, performance, Experimentation

Keywords
Bayesian optimization algorithm, estimation of distribution algorithms, probabilistic models, optimization.

1. INTRODUCTION
Using BOA for large optimization problem is not possible without using more efficient structure learning algorithms. In recent years several algorithms have been proposed which make learning Bayesian network from high dimensional data sets in a reasonable time feasible. In this paper we use one of the most efficient algorithm which has been introduced in [3]. This algorithm is a hybrid algorithm and use a heuristic called max-min parent children (MMPC) for finding the candidate parent set for each variable and then used a hill climbing approach on this constrained search space. We use this Heuristic for model learning in BOA and call it Constrained Model Search Space BOA(CMSS-BOA). Several experiments on different types of benchmark problems are carried out in order to study how the model building time and also population of promising solutions change through the optimization process using CMSS-BOA and standard BOA.

2. BOA and CMSS-BOA
The Bayesian optimization algorithm (BOA)[1,2] generates a population of candidate solutions by building and sampling Bayesian networks. Therefore the order of statistics in the factorization of joint probability distribution of the candidate solution is not restricted. After the random initialization of the population with a uniform distribution over all possible solutions, the population is then updated for a number of generations.

MMPC algorithm, uses a constrained based method to discover possible parents-children relationships in a Bayesian network. Then a search method such as greedy search can be used to find the network which maximize a selected score.

The MMPC algorithm uses a data structure called parent- children set, for each variable X_i that contains all variables that are a parent or a child of X_i in any Bayesian network faithfully representing the distribution of the set of examples.. MMPC uses G^2 statistical test on the set of examples to determine the conditional independency between pairs of variables given a set of other variables. The MMPC algorithm consists of two phases. In the first phase, an empty set of candidate parents-children (CPC) is associated with X_i. Then it tries to add more nodes one by one to this set using MMPC heuristic. This heuristic selects the variable X_j that maximizes the minimum association with X_i relative to current CPC and add this variable to it. The minimum Association of X_j and X_i relative a set of variables CPC is defined as below:

$$MinAssoc(X_i; X_j \mid CPC) = \arg\min Assoc(X_i; X_j \mid CPC)$$

for all subset S of CPC.

$Assoc(X_i; X_j \mid S)$ is an estimate of the strength of the association between X_i and X_j knowing the CPC and is equal to zero if X_i and X_j are conditionally independent given the CPC. The function $Assoc$ uses the *p*-value returned by the G^2 test of independence as a measure of association: The smaller the p-value the higher the association. The first phase of MMPC stops when all remaining variables are considered independent of X_i given the subset of CPC .

Algorithm 1: CMSS-BOA ALgorithm

1. Generate a random initial set of solution S
2. Calculate the fitness of individuals in S
3. Select a subset of promising solution in S
4. Find the CPC of each variable
5. Use a greedy search to find Bayesian network B in constrained space by CPCs witch maximize a score
6. Generate new set of solutions by sampling the Bayesian network B and replace S with this set.
7. If the termination criteria are not meet go to step 2

GECCO'11, July 12–16, 2011, Dublin, Ireland.
ACM 978-1-4503-0690-4/11/07.

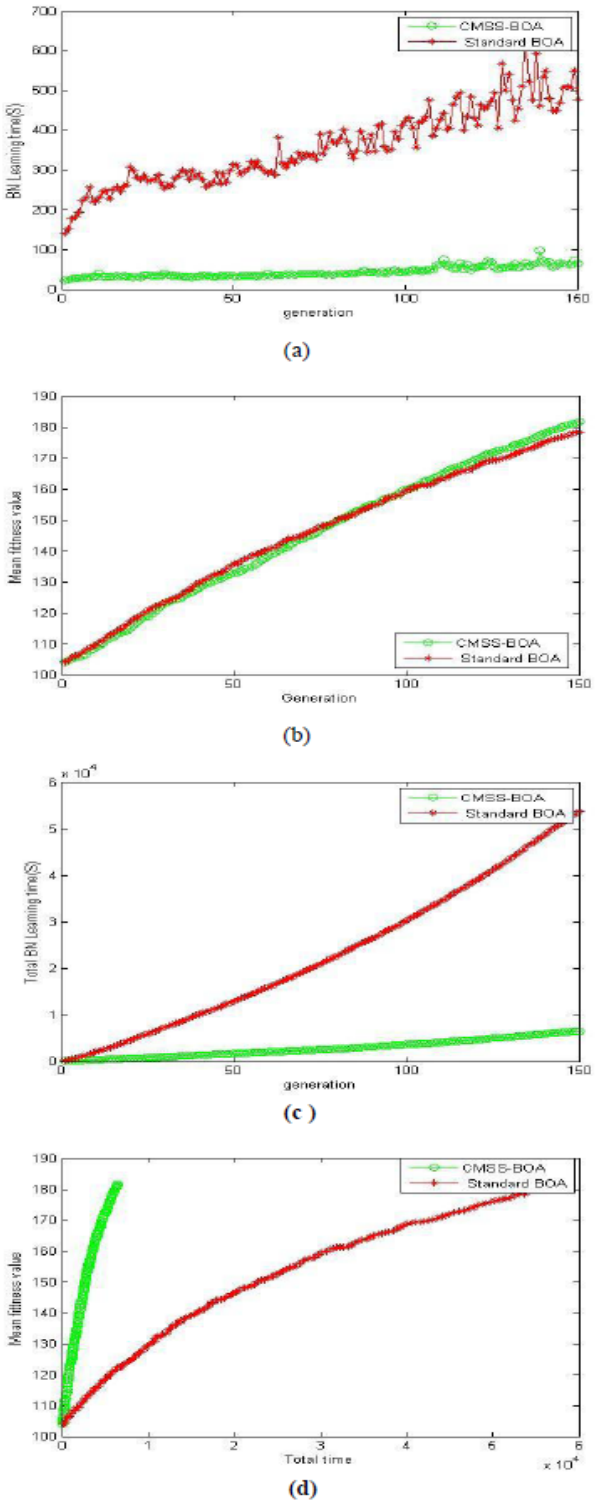

Figure 1 Performance comparison on CMSS –BOA and standard EDA for OneMax combined with 6 3-traps. Total program size is 200 and population size is 1000

After determining the candidate parent set of each variable then a greedy hill-climbing search is performed in the space of Bayesian networks. The important difference from standard hill climbing Bayesian network structure learning is that the search is constrained to only consider adding an edge if it was discovered by MMPC in the first phase. Algorithm 1 summarizes the steps of CMSS-BOA.

We use the MMPC implementation in [3] for finding the candidate parent-children set used in CMSS-BOA.

3. Experimental results

To compare the performance and behaviour of CMSS-BOA and BOA, the experiments on different benchmark functions are performed. In order to make problems with non-uniform sparseness we combine k-trap and *one-max*. In Figure 1 we present the results of the algorithm for a 200 bit problem which is a combination of 3-trap and *one-max*. The population size is 1000 and is chosen from several experiments with different population sizes. We try several population sizes and increased it gradually until obtaining the optimum or close to optimum result in most runs. We present the results of the two algorithms in Figure 1. On average, the best solution of CMSS-BOA has a slightly higher fitness value. The best solution in generation 150 has a fitness value 199 for CSMM-BOA comparing to 192 of standard BOA. Figure 1 (a) presents the average Model building times for each generation. As we can see BOA needs significantly more time for model building and through generations the model building time increases faster than the one of CMSS-BOA which makes the difference between total learning time even more. Figure 2 (c) shows the cumulative learning time in each generation. This result shows a strong improvement in efficiency as the total learning time for CMSS-BOA almost 10 time less than for standard BOA. Fiure1 (b) shows how the average fitness of the population changes through different generations. In this experiment CMSS-BOA has slightly higher results than BOA. Finally, Figure 2 (d) presents how the average fitness of the population changes through time.

4. Conclusion

In this paper we have proposed an Bayesian network based EDA using a recently introduced Bayesian structure learning algorithm.

In this Algorithm The models search space constrains by candidate parent children set without considering uniform sparseness. Therefore It is very useful for solving large and complex problems when searching non homogenous search space is required. Our results show that this algorithm. is able to obtained comparable results (and some time better results) with the algorithms which use the non constrained search space

5. References

[1] Larranaga, P. and Lozano, J. A. Estimation of Distribution Algorithms. Kluwer *Academic publisher, 2002.*

[2] Pelikan, M. Bayesian optimization algorithm: from single level to hierarchy, *Ph.D. Thesis, University of Illinois, 2006.*

[3] Tsamardinos , I., Brown, L. E. , Aliferis, C.F. The MMPC hill-climbing Bayesian network structure learning algorithm, Machine *Learning Journal*, 65(1):31–78, 2006.