

Estimation of Distribution Algorithms For Job Schedule Problem

Jun Li, Yong JIANG

Department of Information and Computing Science
Nanjing University of Science and Technology
Nanjing, 210094, P. R. China
e-mail: math@mail.njust.edu.cn

Abstract—Well known job schedule problem is NP-hard in management and operating system. A few of evolution algorithms such as GA have been proposed to approach this kind of problems. At present estimation of distribution algorithms are the leading heuristic algorithms which overcome the defect of tradition GA and reflex more excellent property than others. In this paper effective of EDA are tested by the job scheduling problem and the simulation results are analyzed in details.

Keywords—Estimation of distribution algorithms; job schedule problem

I. INTRODUCTION

Job schedule problem with deadline can be found in many applications such as management and operating system [1]. It seems that this is a just simple combinatorial optimization problem which can be described as follow.

Suppose that there is only a uniprocessor, non resource constraint and each job has the same processing time. M jobs are offered however merely a sequence of $n(n < M)$ jobs can be solved on the uniprocessor. The purpose is to find a permutation of n jobs which must be finished before respective deadline.

For a small value n and M , by enumerating all the permutations you can easily find the optimum. But for larger one, it would be impractical as there exists A_M^n ways.

In this work we approach the problem by estimation of distribution algorithms (EDA), which is motivated to overcome the defect of traditional GA. The unique strength of EDA lies in both directly using the correlation of the variables and explicitly expressing the structure of interdependence among variables for conducting the promising search direction, which may often capture the fastest convergence speed on the challenging optimization problems.

II. PROBLEM DEFINITION

It is assumed that n jobs are to be processed in a uniprocessor. The amount of processing time for each job is unit. d_i represents the deadline of job $i(1 < i < n)$. As long as the job i is done before its own deadline d_i , we can obtain a part of profit produced by job i which is denoted

by p_i . Let $p = \sum_{i=1}^n p_i$ be the total profits by solving the sequence of n jobs. The problem investigated here is to schedule n jobs from jobs set M which maximize the profit function p and avoid any conflicts with respect to own deadline, therefore the objective function can be formulated as below:

$$f(i_1, i_2 \dots i_n) = \max \sum_{k=1}^n p_{i_k} - f_{penalty} \quad (1)$$

where $\pi(i_1, i_2 \dots i_n)$ is a permutation. The constraint of feasible solution is transfer to a penalty term labeled by $f_{penalty}$. The expression of $f_{penalty}$ has the following ways:

$$f_{penalty} = \begin{cases} \sum_k \alpha p_{i_k} & i_k < k \\ 0 & i_k \geq k \end{cases} \quad (2)$$

α means the penalty factor.

The task is to choose appropriate permutation π^* which maximizes the objective function $f(i_1, i_2 \dots i_n)$.

III. ESTIMATION OF DISTRIBUTION ALGORITHMS

The general mechanism of EDA consists in building probabilistic graphical model estimated by the last selected promising population. The graphical model plays a vital part in the whole procedure since in the next step high quality individual will be generated according to the probabilistic distribution coded by the graphical model [2, 3].

The main scheme of general EDA is given follow:

- 1 Create an initial population Pop_0 consisting of D individuals;
- 2 Repeat steps 3-5 until termination criteria are satisfied;
- 3 Select $Se < D$ promising individuals from last population;
- 4 Calculate the joint distribution of select individuals;
- 5 Generate offspring according the calculated probability distribution.

The structure of model we use in the EDA is a directed acyclic graph (DAG) which tries to describe conditional interdependence about nodes.

Different EDA are classified by the maximum number of parents that a variable can have in the graph model.

- UMDA and PBIL do not consider any dependencies among nodes, as shown in fig. (a), i.e. all the nodes have none parents. Due to its simplicity, both of them perform excellent in linear problem where the variables (nodes) are assumed not significantly interdependent.
- MIMIC as an example of EDA which use bivariate model of distribution, as shown in fig.1 (b). The graphical model of MIMIC has the form of a chain that attempt to express the simplest possible interdependencies between variables.
- Algorithms in this category appear exact from the nature of complex optimization problem. EBNA_{BIC} and EBNA_{K2} as two examples take more than two interdependencies (parent nodes) into consideration. fig.1(c) illustrates complicate model which this kind of algorithms would like to construct.

What this paper concentrated on is confined to the applications of discrete EDA due to the feature of job schedule problem. Gaussian net algorithm is an instance of EDA in continuous domain where it is assumed the joint density function follow a multivariate Gaussian density [3].

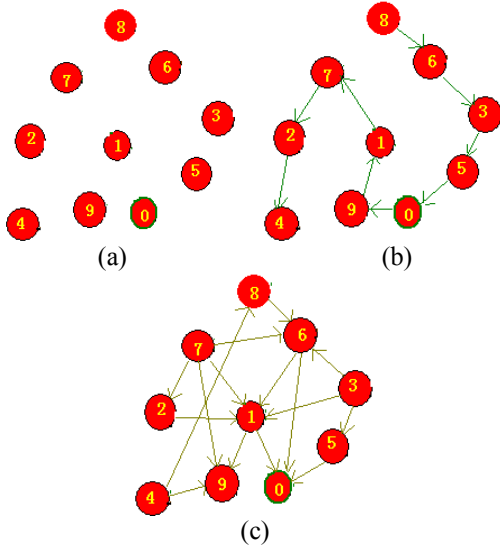


Figure 1 reveals the graph models of three types EDA respectively.

IV. EDA APPROACHES FOR JOB SCHEDULING

A. Order encoding

For job scheduling problem order encoding is used. The individuals have n nodes, each of them taking M possible values. For example:

$\{12, 3, 7, 21, 11, 34, 9, 4, 23, 16\}$ is an instance of $n=10$ and $M=50$. $\{49, 6, 11, 32, 2, 9, 22, 5, 13, 18\}$ is a possible permutation either.

B. Fitness function

The fitness function for EDA to evaluate in each step has the same form as the previous mentioned objective function:

$$f_{fitness} = f(i_1, i_2 \dots i_n) = \max \sum_{k=1}^n p_{i_k} - f_{penalty} \quad (3)$$

C. Estimating the probabilistic graphical model

- Firstly, the first population Pop_0 consisting of D individuals is sampled. The sampling of each individual is usually done by supposing a uniform distribution on each node. As for the first topological order node, the probability of taking arbitrary M jobs (value) is the same (that is $\frac{1}{M}$). However in the next turn, the probability of the job (value) has appeared in the first node must be set to 0 and the probability of the jobs (value) not still appeared are normalized accordingly. The second node is forced to sample in the left $M-1$ job sets.
- Secondly, a number of $Se < D$ individuals are selected by their fitness. The selection method is well known Roulette wheel selection.
- Thirdly, the graphical model reflecting the selected individuals is constructed. As for UMDA and PBIL no graphical model learning is required, but for MIMIC, EBNA_{BIC} and EBNA_{K2} structure learning and refining probability distribution are both required.
- Finally, the new promising population is created by the graphical model learned in the previous step. The graphical illustrates the interrelationships among nodes. In the view of this structure, we can learn the parents of nodes and furthermore the conditional probability.

V. EXPERIMENT

Experiment is carried out to test the performance of EDA. There are 50 jobs available and 10 jobs are to be chosen to schedule on the uniprocessor. The processing time of each job is unit (1).

The profits of 50 jobs are given below:

$p[0]=2$ $p[1]=34$ $p[2]=36$ $p[3]=54$ $p[4]=71$
 $p[5]=28$ $p[6]=11$ $p[7]=57$ $p[8]=17$ $p[9]=69$
 $p[10]=83$ $p[11]=22$ $p[12]=96$ $p[13]=93$ $p[14]=46$
 $p[15]=67$ $p[16]=55$ $p[17]=12$ $p[18]=13$ $p[19]=48$
 $p[20]=57$ $p[21]=87$ $p[22]=18$ $p[23]=70$ $p[24]=54$
 $p[25]=63$ $p[26]=79$ $p[27]=47$ $p[28]=38$ $p[29]=61$
 $p[30]=18$ $p[31]=30$ $p[32]=41$ $p[33]=42$ $p[34]=50$
 $p[35]=1$ $p[36]=7$ $p[37]=26$ $p[38]=36$ $p[39]=20$
 $p[40]=22$ $p[41]=76$ $p[42]=91$ $p[43]=47$ $p[44]=4$
 $p[45]=46$ $p[46]=60$ $p[47]=12$ $p[48]=65$ $p[49]=88$

And related deadlines are as follow:

$d[0]=1$ $d[1]=1$ $d[2]=3$ $d[3]=2$ $d[4]=10$
 $d[5]=5$ $d[6]=7$ $d[7]=1$ $d[8]=9$ $d[9]=8$
 $d[10]=10$ $d[11]=5$ $d[12]=9$ $d[13]=9$ $d[14]=7$

$d[15]=6$ $d[16]=8$ $d[17]=5$ $d[18]=7$ $d[19]=5$
 $d[20]=4$ $d[21]=3$ $d[22]=5$ $d[23]=2$ $d[24]=7$
 $d[25]=1$ $d[26]=3$ $d[27]=4$ $d[28]=6$ $d[29]=3$
 $d[30]=9$ $d[31]=4$ $d[32]=3$ $d[33]=1$ $d[34]=10$
 $d[35]=5$ $d[36]=1$ $d[37]=10$ $d[38]=5$ $d[39]=2$
 $d[40]=3$ $d[41]=9$ $d[42]=8$ $d[43]=6$ $d[44]=2$
 $d[45]=5$ $d[46]=3$ $d[47]=8$ $d[48]=9$ $d[49]=1$

The total profit of optimum solution is 833. The optimum profit and related set of jobs are fixed but some of them can be placed in different positions on the condition of not leading to any conflicts with deadline.

{49, 21, 26, 42, 10, 41, 9, 12, 13, 4} and {49, 26, 21, 13, 4, 12, 42, 9, 41, 10} are two feasible solutions.

For EDA the population size is 800 from which 150 high quality individuals are selected for constructing graphical model. In order to enhance the convergence speed, elitist approach was adopted (that is always the best individual is included for the next population). The algorithm were designed to finish the search when a maximum of 800 generations were reached.

The experiment is executed in 1.5 GHz Intel(R) Centrino(R) Duo processor under Windows XP operating system with 2G RAM.

VI. RESULT

- a) At first we apply EBNA_{BIC} to find the solution of job scheduling problem. During the procedure of modeling, the parents of each node are printout. After the first step, EBNA_{BIC} build the structure with only two node having parents, the corresponded permutation and fitness are {22, 48, 20, 4, 10, 8, 29, 30, 37, 7} and 73, as shown in fig.2(a). In the second step by using the created graphical model, new population is generated for the successive constructing stage. When the second step is done, graphic model remains the same however the solution is more suitable than previous one in terms of fitness value 469, as shown in fig.2 (b). The important point is the third step, the result of third step show both fitness value and permutation sequence equals that of second step (that is {22, 48, 20, 4, 10, 24, 34, 30, 12, 7} and 469), but the graphic model has been reduced to the simplest no-parents graphic model as UMDA and PBIL, as shown in fig.2 (c). There no further structuring learning exists in the consequence step, refining the probability distribution becomes the key part for directing the search space. Fig.2 (d) plots the convergence curve of fitness value.
- b) In view of the performance of EBNA_{BIC}, we may conclude that job scheduling is just a linear problem which doesn't involve interactions among variables (nodes). As displayed previous the idea of MIMIC lies in joining all the nodes together like a chain, it implies MIMIC will fail in the special case when

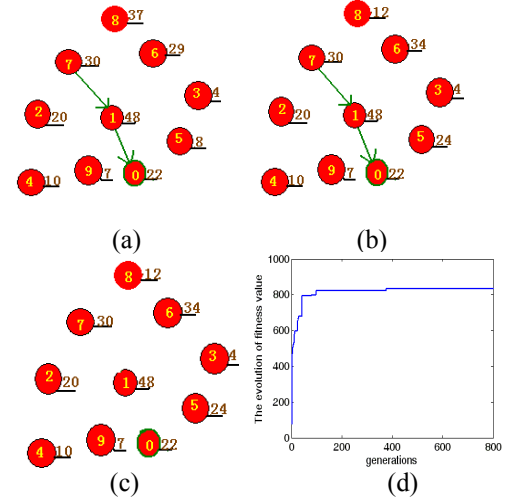


Figure 2 the models of first three steps and curve of fitness value

the nodes itself doesn't interact with any one of them. Therefore MIMIC is expected to return the worst results. To verify this hypothesis, we use it to resolve our problem. Fig.3 reveals the performance of MIMIC. Chain model keeps the same through the whole iterations. It is clearly that for the topological order created by MIMIC and correspondence sampling sequence doesn't coincide the inherent essential of job scheduling problem. The results also show the drawback of EDA, since everything in the world

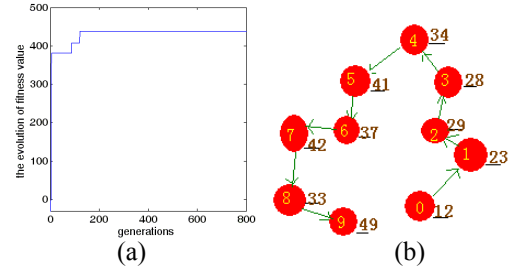


Figure 3 (b) denotes the finally graph model when the iteration is terminated.

(a) Shows the failure of evolution

has some extent of relationship. MIMIC incorporating pair wise dependences should perform better than that of no-parents graph. Naturally we will take consideration the orientation step which indicates the sampling order. In order to decrease the parameter of distribution, EDA introduce graph model to factorize the joint probability which is developed on the basis of condition independent relative to parents. Sometimes the transformation from pair wise dependence to condition independence is

misleading because it is just an approximation and lack of substantial proof.

- c) UMDA is suitable for the variable structure of optimization problem. EBNA_{BIC} destruct the graph model which it builds just before and return the original model. However UMDA did not spend extra time to search feasible structure so it outperforms than EBNA_{BIC}. Fig.4 verifies our hypothesis.

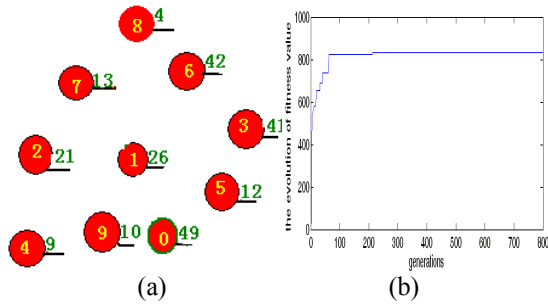


Figure 4 (a) represents the optimum solution.
(b) the fitness curve versus generations

VII. CONCLUSION

In this paper we present EDA as a novel approach for job scheduling problem. Three types of EDA with different complexity are adopted and the simulation results are satisfactory.

Although the results obtain the acceptable solution, it also uncovers the drawback of EDA.

It is clearly that each variable in the job scheduling problem is dependent with others, but surprisingly UMDA return the best results comparing with EBNA_{BIC} especially MIMIC. Intuitive, more than one parent should be involved instead of scattered and isolated nodes. It also means that the relationship rooted in the problem can not be simply exposed by the arrow from parents to node. The internal complication interactions beyond the competence of bayesian network.

In a great number of cases bayesian network as a directly acyclic graph adapted to EDA can not give us reliable information about the interactions among each part of problem. The actual correlation is more complex than what the explicit of bayesian network can describe.

Despite of disability of detecting latent connection underlying objects, EDA is reasonably well than traditional GA. The further task is to design applicable graph model which can give us more information in details for the characters of problems and at the same time decompose special optimization problem to simple parts in order to improve the quality of graph model.

REFERENCES

- [1] Hong Wei Huo, Jin Xu and Zheng Bao, "An algorithm of job-scheduling problem based on genetic algorithm," Systems Engineering and Electronics, Vol 22, No.4, pp. 69-73.
- [2] Larrañaga P, Lozano J A, "Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation," Boston: Kluwer Academic Publishers, 2002.
- [3] P. A. N. Bosman, D. Thierens, "Continuous iterated density estimation evolutionary algorithms within the IDEA framework," In: M. Pelikan et al.(eds.), Proceedings of the Optimization by Building and Using Probabilistic Models OBUPM Workshop at the Genetic and Evolutionary Computation Conference-GECCO-2000, San Francisco: Morgan Kaufmann Publishers, 2000, pp.197-200.