ISSN 2250-3153

Application of Genetic Algorithm in Graph Theory

Dr. Mrs. G.Nirmala[1], Mr.D.Ramprasad[2]

[1]Associate Professor in Mathematics, PG and Research Department of Mathematics, K.N.G. Arts College for Women (Autonomous), Thanjavur
[2]Professor in Mathematics, PG and Research Department of Mathematics, A.V.V.M. Sri Pushpam College (Autonomous), Poondi

Abstract- In this paper, we address a single machine family scheduling problem where jobs, each characterised by a processing time and an associated positive weight, are partitioned into families and setup time is required between these families. For this problem, we propose a genetic algorithm using an optimised crossover operator designed by an undirected bipartite graph to find an optimal schedule which minimises the total weighted completion time of the jobs in the presence of the sequence independent family setup times. The proposed algorithm finds the best offspring solution among an exponentially large number of potential offspring.

Index Terms- Genetic algorithm, single machine scheduling

I. Introduction

Genetic algorithms are search algorithms based on the mechanics of natural selection and genetics. They combine survival of the fittest among string structures with a structured yet randomized information exchange to form a search algorithm with some of the innovative flair of human search.

The central theme of research on genetic algorithms has been Robustness, the balance between efficiency and efficacy necessary for survival in many different environments. The normal action of GA is to combine good individuals with characteristics to produce better.

Genetic algorithm optimizers are robust, stochastic search methods modeled on the principles and concepts of natural selection and evaluation.

PRELIMINARIES:

Definition 1

A simple genetic algorithm that yields good results in many practical problems is composed of three operators.

- **Reproduction**

Reproduction is a process in which individual strings are copied.

100,001,011,000,000 → 100,100,000,011,000

- **Crossover**

Crossover is a process of changing one type to the next type.

100,100	011,001,001
100,100	011,000,000

100,100	011,000,000
100,100	011,001,001

- **Mutation**

Mutations creates a sudden bit change.

100,100,011,000,000 → 100,100,011,000,101

Figure 1

Definition 2

The objective function defining the optimization goal called a fitness function in the GA [$f : c \rightarrow R^+$].

The mutation operator provides a means for exploring portions of the solution surface that are not represented in the genetic makeup of the current population.

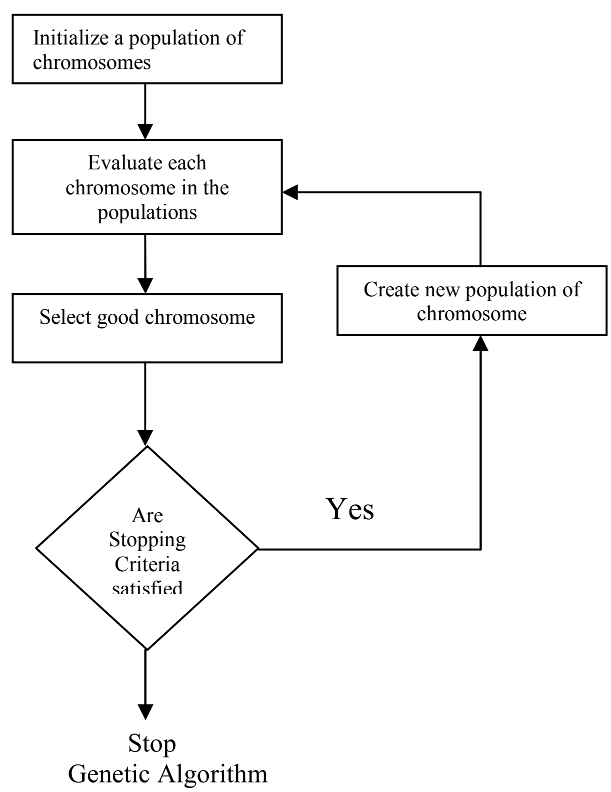

Figure 2

Definition 3

A natural way of encoding the problem would be to present each solution by a bit string. We use a binary {0,1} representation that is proposed by Mason. Using this binary representation we can define the partition of families into

ISSN 2250-3153

batches, where '1' means the first job in a batch and '0' means a contiguously sequenced job in a batch. The length of the individual corresponds to the number of jobs N.

Definition 4

The fitness function is used to evaluate the individuals which are introduced into the population. We define the fitness function using the property of SWPT (Shortest Weighted Processing Time) rule for batches developed by Mason and Anderson, in which there exists an optimal schedule that the batches appear in SWPT order. As mentioned earlier, a batch is a maximum group of contiguously scheduled jobs with in a family. A sequence independent family setup time S_f , is required at the start of the schedule and also whenever there is a switch in processing jobs from one family to jobs of another family.

Definition 5

A linear graph G = (V,E) consists of a set of objects V = { v_1 ,v_2 , …} called vertices and another set
E = { e_1 ,e_2 ,….} , whose elements are called edges. A graph that has neither self-loops nor parallel edges is called a simple graph.

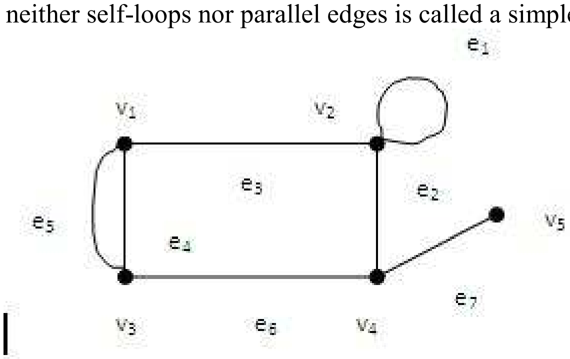

Figure 3

Definition 6

A graph G is called bipartite if its vertex set V can be decomposed into two disjoint subsets V_1 and V_2 such that every edge in G joins a vertex in V_1 with a vertex in V_2.

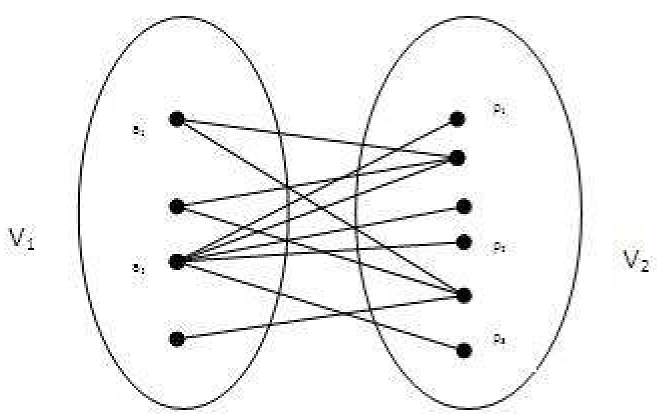

Figure 4

Definition 7

A directed graph G consists of a set of vertices V={v_1,v_2,…} , a set of edges E = {e_1,e_2,…} and a mapping ψ that maps every edge onto some ordered pair of vertices (v_i,v_j). As in the case of undirected graphs, a vertex is represented by a point and an edge by a line segment between v_i and v_j.

Definition 8

Any set M of independent lines of a graph G is called a matching of G, we say that u and v are matched under M. we also say that the points u and v are M-saturated. A matching M is called a perfect matching if every point of G is M-saturated. M is called a maximum matching if there is no matching M' in G such that | M' | > | M | .

Examples

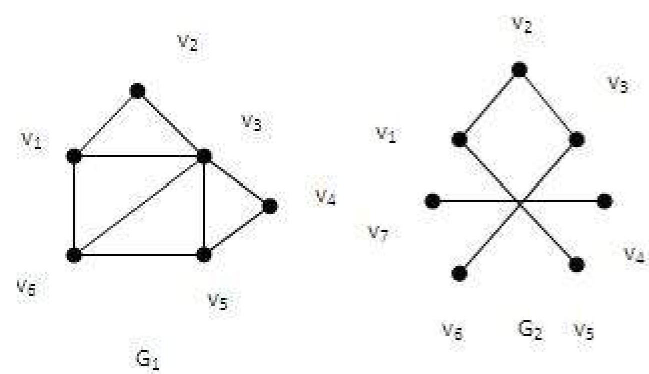

Figure 5

Consider the G_1, M_1 = {v_1v_2, v_6v_3, v_5v_4} is a perfect matching in G_1. Also M_2 = {v_1v_3 , v_6v_5} is a matching in G_1. However M_2 is not a perfect matching. The points v_2 and v_4 are not M_2–saturated. For the G_2 ,

M = {v_8v_4 , v_1v_2} is a maximum matching but is not a perfect matching.

II. RESULT

Solution for single machine family scheduling problem by applying Graph theory in GA.

ALGORITHMIC REPRESENTATION

According to Aggarwal et al during the optimised crossover scheme, two parents produce two new children. The first child is called the X-child and the second child is called the Y-child. The X-child is constructed in such a way that have the best objective function value from the feasible set of children, while the Y-child is constructed so as to maintain the diversity of the search space. We propose an optimised crossover operator within genetic algorithm for the problem of 1 $S_f \sum w_j c_j$. The proposed optimised crossover using an directed bipartite graph finds the two new children which are called X-child and Y-child. We will now explain the optimised crossover strategy on determining the X-child and Y-child for the problem of $1 \mid S_f \mid \sum w_j c_j$.

Step-1

Identify the parent with the least weight $\sum w_j c_j$ as P_1 and select family within the P_1, which contains the batch with the largest weighted processing time as family f , label another parent as P_2. The family f will be used in both P_1 and P_2.

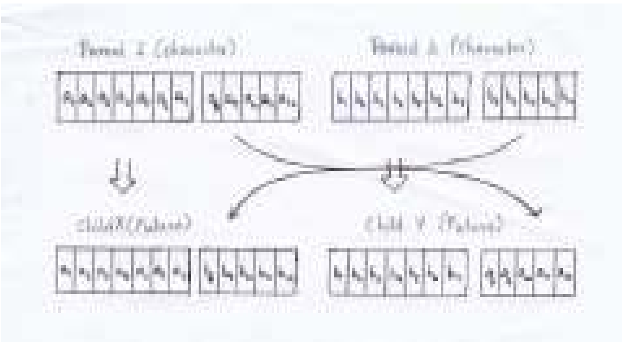

Figure 6

Step-2

Construct an undirected bipartite graph $G=(U \cup V, E)$ where $U=\{u_1, u_2, \ldots\ldots u_n\}$ representing the jobs of family f, $V=\{v_1, v_1', v_2, v_2', \ldots.\ v_n, v_n'\}$ representing bit situation of the jobs of family f in both P_1 and P_2 (i.e. v_i , v_i' Ɛ $\{1,0\}$) , and E representing the arc set in the graph in which , $\{u_j, v_i\}$, $\{u_j, v_i'\}$ Ɛ E iff job j is represented with the bit situation v_i and v_i' respectively.

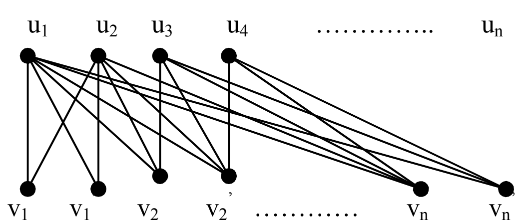

Figure 7

$\{u_1v_1 , u_2v_1' , u_3v_2 ,\ldots\ldots\ldots u_n\}$,

$\{u_1v_1' , u_2v_2 , u_3v_2' ,\ldots\ldots\ldots\}$.

Step – 3

Determine all the maximum matchings in graph G. Suppose that there are K jobs of family f that are represented with a different bit situation in the two parents. There will be exactly 2k maximum matching's in graph G.

Jobs

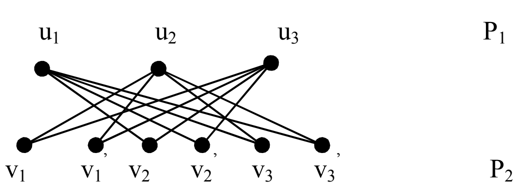

Figure 8

bit situation of jobs of family in both P_1 and P_2

Maximum matchings are

$\{u_1v_2 , u_2v_1 , u_3v_1'\}$, $\{u_1v_2' , u_2v_1' , u_3v_2\}$

$\{u_1v_3 , u_2v_3' , u_3v_2\}$, $\{u_1v_3' , u_2v_3 , u_3v_2'\}$

$\{u_1v_3 , u_2v_1' , u_3v_1\}$, $\{u_1v_3' , u_2v_3 , u_3v_1'\}$

$\{u_1v_3' , u_2v_3 , u_3v_2\}$, $\{u_1v_2 , u_2v_3 , u_3v_2'\}$

Step-4

Generate a temporary offspring from P1 by replacing the bit situations of the jobs in family f which corresponds to one of the maximum matchings in graph G. Repeat the procedure for 2k-1 times to generate 2k temporary offspring. Note that one of the temporary offspring is exactly as P1, thus removed.

REFERENCES

[1] Aggarwal, C.C., Orlin, J.B and Tai, R.P "Optimized crossover for the independent set problem", Operations research.

[2] Ahuja, R.K., Orlin, J.B. and Tiwari, A., 2000. "A greedy genetic algorithm for the quadratic assignment problem", Computers & Operations Research.

[3] Allahverdi, A., Gupta, J.N.D. and Aldowaisan, T., 1999. "A review of scheduling research involving setup considerations", OMEGA, International Journal of Management Science.

[4] Allahverdi, A., ng C.T., Cheng, T.C.E. and Kovalyov, M.y., 2008. "A survey of scheduling problems with setup times or costs", European Journal of Operational Research.

[5] Balas, E. and Niehaus, W., 1998. "Optimized crossover-based genetic algorithms for the maximum cardinality and maximum weight clique problems", Journal of Heuristics.

[6] Holland, J.H, 1975,"Adaptations in natural and artificial systems" Ann Arbor, The University of Michigan Press.

[7] Kacem, I. and chu, C., 2008. "Worst-case analysis of the WSPT and MWSPT rules for single machine scheduling with One planned setup period", European Journal of Operational Research.

[8] Leung, J.Y-T., 2004. "Handbook of scheduling. Algorithms, models, and performance analysis", Chapman & Hall/CRC, US.

[9] Mason, A.J 1992 " Genetic algorithms and scheduling problems", Ph.D thesis, university of Cambridge, UK.

[10] Mason, A.J and Anderson, E.J 1991. "Minimizing flow time on a single machine with job classes and setup times", Naval Research Logistics.

[11] Monma, C.L. and Potts, C.N., 1989. "On the complexity of scheduling with batch setup time", Operation Research.

[12] Smith, W.E., 1956. " Various optimizers for single-stage production" , Naval Research Logistics Quarterly.

[13] Wang, D., Gen, M. and Cheng, R., 1999. "Scheduling grouped jobs on single machine with genetic algorithm", Computer and Industrial Engineering.

AUTHORS

First Author – Dr. Mrs. G.Nirmala Associate Professor in Mathematics, PG and Research Department of Mathematics, K.N.G. Arts College for Women (Autonomous), Thanjavur. Email: nirmalamanokar11@yahoo.com

Second Author – Assistant Professor in Mathematics, PG and Research Department of Mathematics, A.V.V.M. Sri Pushpam College (Autonomous), Poondi. Email: dramprasaad81@yahoo.com