Implementation of Time Scheduling System for Staff Using Genetic Algorithm

Moe Ka Lhyar Naing, Myo Myo Khin
Computer University (Dawei), Myanmar
moekalayar2009@gamil.com

Abstract

Time Scheduling System represents a subclass of scheduling problems that are hard to solve. Genetic Algorithm (GA) has emerged as a useful tool for the scheduling and timetabling problem. This paper is concerned with the development of Genetic Algorithm (GA) for time scheduling system. The implementation of time scheduling system applied staff from a restaurant. The scheduling system consists of assigning staff to daily duties and off day during a planning period. Genetic Algorithm (GA) takes the combination of parameters or chromosomes from a set concerned with the staff to implement and evaluate the system. This system used constraint to give equivalent workload among staff. This scheduling system for restaurant can be view as constraint satisfaction problem. This system uses genetic algorithm and gives the feasible schedule for staff at restaurant.

1. Introduction

Genetic Algorithm was applied to solve problem arising in the area of scheduling and timetabling. Time scheduling system consists of shift assignment and rest day of staff working at a restaurant. Scheduling staffs involves considerable time and resources, and it is often difficult to create Schedule that satisfies all the requirements.

Successful implementations require mechanisms for overcoming these difficulties. Genetic Algorithm (GA) can be used in various optimisation problems [6]. Genetic Algorithm have been applied to many other practical application, such as scheduling problem, computer aided design (CAD), economics, game theory and many other.

Genetic Algorithm maintains a population of candidate solutions for the problem at hand, and makes it evolve by iteratively applying a set of genetic operator. The solution from the population is encoded as chromosome.

Fitness function is needed to successfully operate Genetic Algorithm. The fitness function is always problem dependent. The fitness of chromosome is linked to the constraint from problem or objective function. This system used search method of Genetic Algorithm (GA) to implement and evaluate time scheduling problem.

2. Genetic Algorithm (GA)

Genetic algorithms are adaptive methods, which may be used to solve search and optimization problems. There are several approaches [8] to staff scheduling based on constraint [9]. The genetic algorithm works with population of candidate solutions that are encoded as chromosomes. The search takes place in the chromosomes where changes, and their effects, are graded by the survival and reproduction of the species. This is the basis for survival of the fittest and the passing on of these characteristics to future generations [4].

In genetic algorithm, the population consists of a set of solutions or individuals instead of chromosomes. A crossover operator plays the role of reproduction and a mutation operator is assigned to make random changes in the solution. A selection procedure selects a certain number of parent solutions, also called offspring. At the end of each iteration the offspring together with the solutions from the previous generation, after undergoing a selection process to keep a constant population size. The solutions are evaluated in terms of their fitness values identical to the fitness of individuals [10].

Each chromosome encoded from the parameters holds the potential solution. These solutions are represented in binary as string of 0s and 1s, but other encoding is possible .Initially many individual solutions are randomly generated to form an initial population, covering the entiring range of possible solutions.

This algorithm consists of three fundamental steps, excluding the initial creation of the population. During the evolution process the fitness of the evolution is calculated. Next, a subset of the population is selected based on predefined selection criterion. Finally the selected subpopulations recombined, and the result is a new population .The algorithm begins again with the new population and the process continues until the solution is reached [7].

[Start] Generate random population of *n* chromosomes

Step1 **[Fitness]** Evaluate the fitness of each chromosome in the population

Step2 **[Selection]** Select two parent chromosomes from a population according to their fitness

Step3 **[Crossover]** Use a crossover probability cross over the parents to form new offspring.

Step4 **[Mutation]** Use a mutation probability mutates new offspring at each position in chromosome.

Step5 **[loop]** Go to step1**.** If the end condition is satisfied, stop, and return the best solution in current population.

The workflow of Genetic Algorithm with selection and genetic operator (crossover and mutation) is shown in Figure 1.

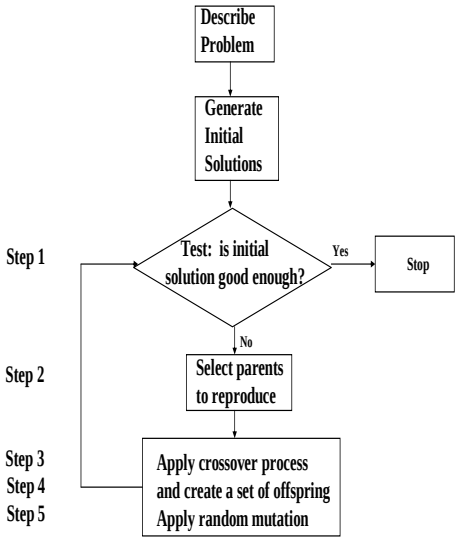

Figure1. The flow chart of Genetic Algorithm

3. Staff Scheduling System

The problem is that of creating weekly schedule for staff at restaurant. In this problem used three shift system [1] .Each day consists of three shifts: a morning shift or M (6a.m. to 3p.m.), an evening shift or E (3 p.m. to 12 a.m.) and a night shift or N (12a.m. to 6 a.m.). Staff has to be assigned only one off day per week. On any day only one shift can be assigned to any staff. The solution is obtained by randomly assigning each staff to one of three shifts or off day.

This system includes:

_ The number of staff assigned to each working shift must be within the range.

_ The number of shifts assigned to each staff must be within the limitation.

_ Prohibited working patterns must be prevented.

"Working pattern" is a sequence of working shifts over several days.

The scheduling problem evaluate under different conditions. For example, the scheduling term of one week and 14 staff to be scheduled. The problem is to assign shift numbers (morning, evening, night and off day) .i.e.; value to the number of staff *number of days (14*7) 98 variables. This system has three types of constraint.

All the constraint must be satisfied to obtain a feasible solution.

(1)Type one constraint

Every staff works exactly one shift per day. But, staff receives only one off day per week.

(2)Type two constraints

The following prohibited working patterns must be avoided by the system to assign feasible shifts.

-morning shift after night shift

-three consecutive night shift

(3)Type three Constraint

Restaurant system has three types of staff: cashier, waiter and chief. Number of worker at each type can be limited by administrator or manager from the restaurant.

	CASHIER	WAITER	CHIEF
Morning	2	7	5
Evening	2	7	5
Night	2	4	2
Off	1	3	2

This paper is concerned with the constraint. The constraint can be formulated as integer linear programs follow:

Decision variables

$$x_{ij} := \begin{cases} 1 & \text{staff i works shift j} \\ 0 & \text{else} \end{cases}$$

$$y_i := \begin{cases} 1 & \text{staff i work off day} \\ 0 & \text{else} \end{cases}$$

Parameters:

p =number of employees

i =index for employee

j = index for off day

c1=number of three consecutive night shifts

c2=count of morning shift after night shift

$$\sum_{i=1}^{p} \sum_{j=1}^{7} x_{ij} = 1 \qquad (1)$$

Every staff works exactly one shift pattern.

$$\sum_{i=1}^{p}\sum_{j=1}^{7}(c_1 + c_2) \qquad (2)$$

Staff does not have three consecutive night shift and morning shift after night shift.

$$\sum_{i=1}^{p} y_i = 1 \qquad (3)$$

Every staff receives one off day per week.

Using Integer linear Program [3], the fitness function becomes:

$$\sum_{i=1}^{p}(\sum_{j=1}^{7} x_{ij} - (\sum_{j=1}^{7} (c_1 + c_2))) + \sum_{i=1}^{p} y_i \qquad (4)$$

This value is equal to the raw fitness of a solution. The system use tournament selection throughout the experiment. This selection method randomly chooses chromosomes or solutions from the population and makes these solutions as sub-population. Thus, the raw fitness was used to calculate the sub-population.

The success of a genetic algorithm relies on the crossover operator being able to combine good partial solutions, so called building blocks, into complete good solutions [5]. The system was motivated by an observation made during parameter testing. One point crossover sometimes gave better results than uniform. The crossover operator is to force the genetic algorithm (GA) to keep good sub-solutions or building blocks together [2].

4. Implementation of the System

In this paper, the time scheduling system is implemented by using Genetic Algorithm (GA). To make schedule for staff, GA needs number of staff, shift and day as chromosome. The system uses 10 bit chromosomes for representing three different portions. There are 3 bit for day(Sunday, Monday, Tuesday, Wednesday, Thursday, Friday and Saturday), two bit for shift (morning M, evening E, night N and off day) and five bit for staff. For example -0011011001, duty of staff ID 24 is night shift on Sunday.

Sun	Mon	Tues	Wed	Thus	Fri	Sat
001	010	011	100	101	110	111

Table 1.Example chromosome for day

M	**E**	**N**	**Off (X)**
00	01	10	11

Table 2. Example chromosome for shift

Evaluation of the fitness takes these chromosomes to get feasible schedule. By using selection method, two parents are needed to choose randomly as follows:

Day	Shift	Employee
001	01	11001

Table 3. Random number to parent

Parent 1: 0010111001
Parent 2: 0101011011

Two chromosomes are separated at a random site and then swapped the tail of the two to get new offspring as follows:

Offspring 1: 0010011011
Offspring 2: 0101111001

After getting offspring, single bit changes randomly as follows:

Offspring 1: 0**1**10011011
Offspring 2: 010111**0**001

The final offspring is needed to calculated fitness function. The result of fitness function is the total number of shift count. For example, there are 14 employees for chief in this system. By substitution number of employee in equation (1), total number of shift count for chief is 98. Then, equation (2) is applied to satisfy the type 2 constraints and the number of shift count is 98. In this example, the result of shift count from equation (1) and (2) is the same. Thus, the end condition is satisfied in this case. After that, equation (3) is applied to calculate the off day in a week for every staff. In this example, the number of off day is 14. Thus, by equation (4) total number of shift count for 14 employees is 210.

There are three constraints in this system. They are staff does not have three consecutive night shift and morning shift after night shift. And then, every staff works exactly one shift per day and one off day per week. All the constraints must be satisfied to obtain feasible solution. After completing all the steps, the final result of the system is as shown in table 4 and 5.

Chief	Sun	Mon	Tue	Wed	Thu	Fri	Sat
MiMi	10	01	10	10	01	00	11
Ni Ni	01	10	11	01	00	00	01
Pa Pa	11	00	01	00	00	10	01

Table 4. Chromosome Solution for chief

Chief	Sun	Mon	Tue	Wed	Thu	Fri	Sat
MiMi	N	E	N	N	E	M	X
Ni Ni	E	N	X	E	M	M	E
Pa Pa	X	M	E	M	M	N	E

Table 5. Example solution for chief

5. Process Flow of the System

The system flow for staff level has two steps. There are view staff list and schedule. Staff list has three types: cashier, waiter and chief. In view schedule state, staff can only view schedule about their duties during weekly period.

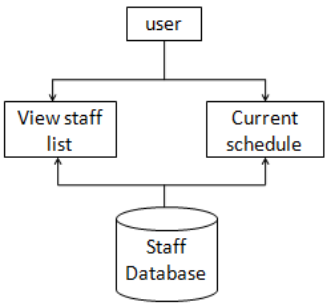

Figure 2. System flow for user

The administrator needs to enter name and password to the system. If name and password is valid, administrator can edit and create schedule with GA. When administrator edits staff data, this data is connect with staff database. In the create schedule process, administrator can create new schedule for next week. And then save this schedule to the staff database, so user can view their duties schedule.

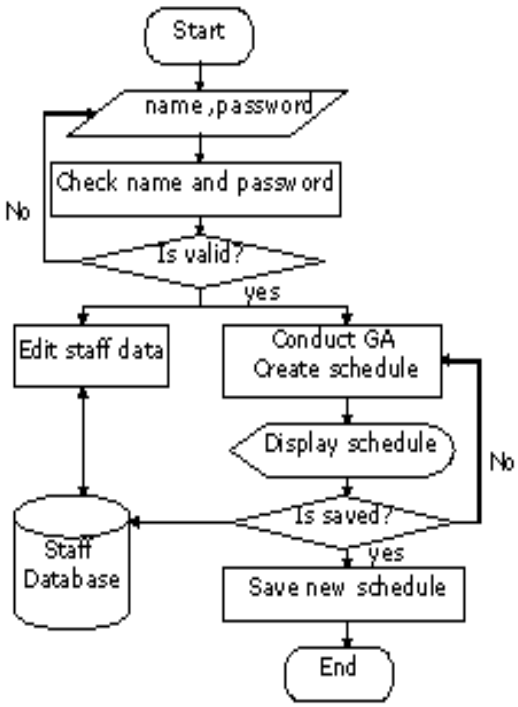

Figure 3.System flow for administrator

The user can knows the constraints about their schedule as shown in figure 4.

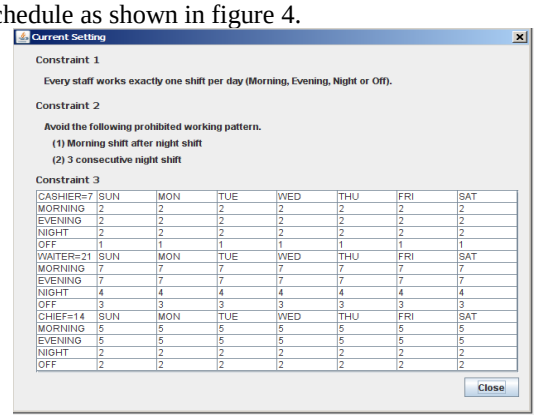

Figure 4. Constraints of the system

Administrators are authorized to control staff data. The figure 5 shows the staff information.

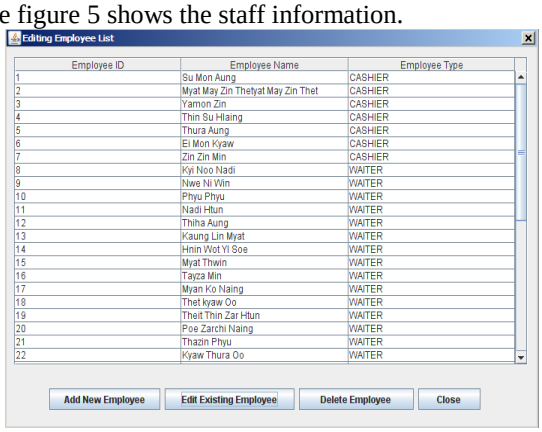

Figure 5. Staff Information

Administrator can create new schedule for next week. According to the staff's information and constraints, the schedule is achieved. The figure 6 shows solution of schedule to the system

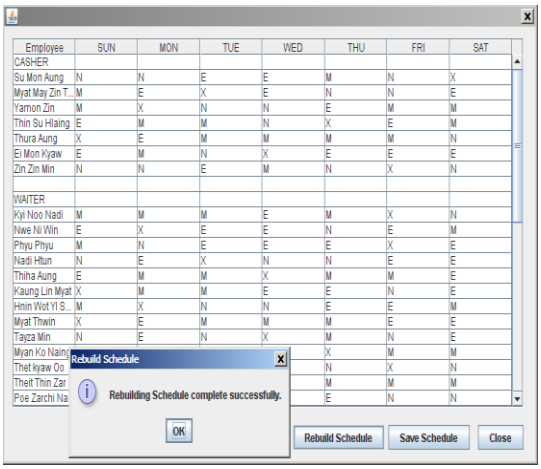

Figure 6. Schedule of the system

6. Conclusion

The genetic algorithm can be used in various optimization problems. Genetic Algorithm (GA) is more robust than other search methods. This paper is concerned with the development of GA for staff scheduling system at a restaurant. This system can apply in many scheduling system for staff. The system produces the schedule for staff according to constraint. Time Scheduling system can reduce costs and time .The scheduling system can give equivalent workload among staff.

[7] References

[1] A. Ikegami, A.sNiwa, M.Ohkura, "Nurse scheduling in Japan", Commun. Oper. Res.Society of Japan, vol.41, 1996.

[2] Beasly D., Mastin R and Bull D., Reducing Epitasis in Combinational Problems by Expansive Coding Proceedings ICGA 5, 1993.

[3] Hadj-Alouane A. and Bean J., A Genetic Algorithm for the Multiple –Choice Integer Program, 1997.

[4] Holland, J., Adaptation in Natural and Artificial Systems, Ann Arbor: The University of Michigan Press, 1995.

[5] Holland, J., Adaptation in natural artificial systems, University of Michigan Press, Ann Arbor, 1975.

[6] Holland, J., "Concerning Efficient Adaptive Systems, " Self Organizing systems ,Washington D.C. :Spartan books,1962.

[7] Koza, J., "Genetic Programming, Inc.HomePage," available online at www.genetic programming.com.

[8] S.Abdennadher, Schlenker, "Nurse scheduling using constraint logic programming.", In Proc. of the 11th Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-99),1999.

[9] V.Kumar, "Algorithms for Constraint Satisfaction Problems: A Survey", AI Maganizine, vol.13, no1, 1992.

[10] www.obitko.com/toturials/genetic-algorithm.com