**Problem Statement:**

Purpose: The project helps medical staffs schedule the operations of hospitals automatically with information and requirements related to the operations:

Basic Requirements:

1. The same doctor can only operate on one patient at the same time.
2. The operating room can’t be scheduled for 2 different operations simultaneously.
3. The operating room need offer sufficient space for operating staffs, doctors and the patient.

Advanced Requirements:

1. Doctors may have preference of time to do the operation.
2. The patient can choose their time to do the operation.
3. The doctor must have 8 hours break between two operations.
4. The surgery should be scheduled by order of importance or urgency.
5. The familiarity between doctors will also be taken into considerations.
6. Patients may have preference of attending physician.

Information related to the operations:

1. Operation: the operations will be classified by different departments via different codes.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Plastic Surgery | Internal Medical | Cerebral Surgery | Neurosurgery | Thoracic Surgery | General Surgery | Traumatology | Orthopedics | Anus & Intestine Surgery | Urology |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |

If doctors from two or more departments are involved in the operation, the operation code will combine two or more department codes (No more than three different departments or choose the most important departments involved the operation to represent), e.g. 1518 means that the operation is done by doctors from thoracic surgery department and orthopedics department. And the last three digits of the operation code are randomly produced by the hospital system.

1. Operating room: a room with room code to be scheduled in the whole operation schedule. Different rooms have different capacities, which is also an important parameter to be considered.
2. Time Table: all the scheduled operations on the time table
3. Doctor: The Doctor can only do one operation, but an operation may require several doctors. To guarantee the success of the operation, the doctor will be forbidden to do the next operation if the doctor doesn’t have enough rest.
4. Patient: The patient will be received treatment by the operation.

**Project Design:**

This project aims to help hospital staffs arrange all operations within two weeks, arrange all doctors to do operations, arrange operating rooms to guarantee enough space for the operation, and arrange appropriate time to do the operation.

**Main Idea and Methods:**

Genetic algorithm, Greedy algorithm

Genetic Algorithm:

1. Genetic Representation: Each candidate solution has a set of properties, chromosomes [1]. In this project, the chromosomes, which represents the operation schedule, will include all the needed information, operation number, operating room number, time, doctor number, and patient number, which can be mutated and altered. Each chromosome will be scheduled in the time table possibly and each operation schedule is assumed as a chromosome.

Operation = 001518001

Patient = 0021

Doctors = 0023,0437

Operating Room = 021

Time = 10

Operation = 131415902

Patient = 0164

Doctors = 0102, 0231, 4765

Operating Room = 086

Time = 12

Operation = 001618081

Patient = 1652

Doctors = 023, 7821

Operating Room = 193

Time = 06

…

Fig. 1 a candidate solution with some chromosomes

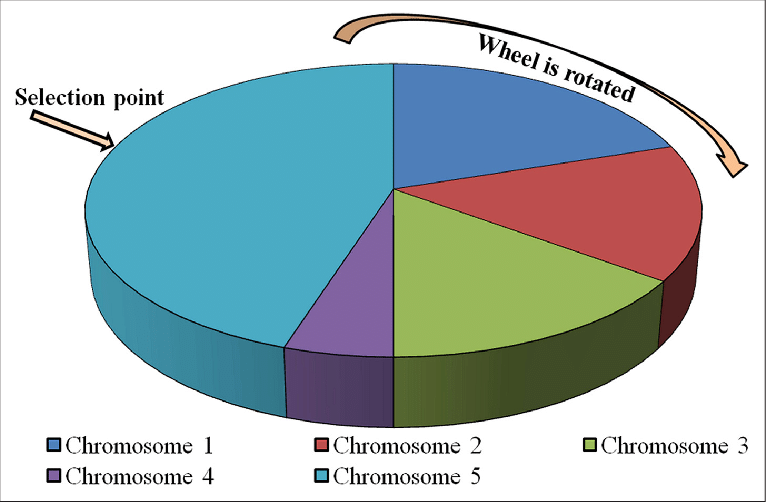
1. Fitness Function: Each candidate solution will go through the fitness function and calculate the fitness score. The fitness score is a numerical method to accurately reflect whether the candidate solution is a reasonable schedule or a better schedule. For each solution, it has a fitness score, which is from 0 – 1, the higher the better. And 1 is the best condition for the operation schedule.
2. Genetic Process: Initialization, Selection, Genetic operators (Crossover, Mutation), Termination [1]. At the beginning, the program will get information from user’s input, and generate a group of chromosomes randomly.

Fig.2 Fitness proportionate selection (roulette wheel selection)

After initialization, the selection part needs to use fitness function to score each solution firstly and use roulette wheel selection method to choose parent chromosomes in this generation to mate and give a birth to new chromosomes (new candidate solution born) with genetic operators. The fitness proportionate selection is a probability-based selection, the solution with higher fitness score has higher possibilities to be chose as the parent solutions to reproduce the next generation.

Next procedure of the program is the mating, which contains crossover and mutation and generate the new generations. Just as mating in the nature, all the candidate solutions crossover their chromosomes and even some chromosomes will be mutated. The program will select several bad chromosomes (these chromosomes represents those unreasonable operation schedules) first, else will select some chromosomes randomly. We assume these chromosomes as each operation schedule, so the information about the operation can be also assumed as some traits which expressed by the chromosome. For example, fig.1 illustrates us that a candidate solution, which contains many chromosomes (operation schedule). In each chromosome, there are many traits (operation schedule information) on the chromosome. The first chromosome of fig.1 tells us operation number, operating room number, doctors’ number, and time number, which are assumed as traits in the project. The mating process aims to exchange some traits on specific or random chromosomes to generate new generation chromosomes and new candidate solution. The mutation is also necessary for the mating process. Under the premise of no conflicts between all the operation schedules, mutation will occur randomly in the all chromosomes with a quite low possibility. For instance, there are two candidate solutions with all the chromosomes (operation schedules) mating with each other. In the example, the first chromosomes of these two-candidate solution are not reasonable enough, cause the room schedule for these two candidates is not the best choice. The program will exchange the room schedule, time table, or other traits on the chromosomes, which is the crossover processing between the chromosomes. In some specific conditions, the chromosomes have already crossover traits several times, however, there still are some schedule conflicts between some operation schedules (chromosomes). To fix these extreme conditions, the mutation also is referred from biology to the program. To satisfy the requirements, in the chromosomes which still have schedule conflicts mutation is needed to relief these conflicts at some degree. And mutation always is undirected, which could bring good results or bad results randomly in the nature. To improve the solution, we use the thought of greedy algorithm, and the directed mutation is used to deal with those specific conflicts in the chromosomes. The suitable room, time, doctors or any other traits will be served in these chromosomes in advance. For example, the fig.3 illustrates us the mating process between two candidate solutions, including crossover and mutation, and the first pair chromosomes exchange the operating room trait between the parent candidate solution, and the operating room is randomly selected to be exchanged to improve the fitness score. Mutation occurs to the second pair chromosomes in the candidate solutions, because crossover cannot fix the conflicts, then to make sure operation can be scheduled, we assign another free doctor, doctor No. 0103, to do the operation instead of No. 102. This mutation between the second pair chromosomes also can be considered as a directed mutation. In the last pair of chromosomes, we can swap the time to deal with the schedule conflicts, which is also considered as directed crossover involved with manual intervention.

Candidate Solution 1

Operation = 001518001

Patient = 0021

Doctors = 0023,0437

Operating Room = 021

Time = 10

Operation = 131415902

Patient = 0164

Doctors = 0102, 0231, 4765

Operating Room = 086

Time = 12

Operation = 001618081

Patient = 1652

Doctors = 023, 7821

Operating Room = 193

Time = 06

…

Operation = 001518001

Patient = 0021

Doctors = 0023,0437

Operating Room = 031

Time = 10

Operation = 131415902

Patient = 0164

Doctors = 0078, 0231, 4765

Operating Room = 086

Time = 12

Operation = 001618081

Patient = 1652

Doctors = 023, 7821

Operating Room = 193

Time = 02

…

Candidate Solution 2

Candidate Solution 3

Operation = 001518001

Patient = 0021

Doctors = 0023,0437

Operating Room = 031

Time = 10

Operation = 131415902

Patient = 0164

Doctors = 0103, 0231, 4765

Operating Room = 086

Time = 12

Operation = 001618081

Patient = 1652

Doctors = 023, 7821

Operating Room = 193

Time = 02

…

Operation = 001518001

Patient = 0021

Doctors = 0023,0437

Operating Room = 021

Time = 10

Operation = 131415902

Patient = 0164

Doctors = 0098, 0231, 4765

Operating Room = 086

Time = 12

Operation = 001618081

Patient = 1652

Doctors = 023, 7821

Operating Room = 193

Time = 06

…

Candidate Solution 4

Mating process

Crossover

Mutation

Fig.3 The Mating process between the two candidate solutions

After the reproduction process, the new generation of solutions with chromosomes will go into the selection process and repeat the whole process again and again until reaching the iteration times or the expected solution coming. Lastly, the output will print out the best solution ever founded in the termination part. The Fig.4 gives us a brief introduction about the whole procedure.