**Diabetic Diet Scheduling using Genetic Algorithm**

Prachi , Dr. Shampa Chakraverty

Department of Computer Engineering

      Netaji Subhas University of Technology (previously NSIT, DU)

**Abstract:**This article presents a system which schedules diet based on the nutritional needs  and preferences of a diabetic patient . Several researches have used different parameters to determine the diet schedule previously. We have taken into consideration the amount of some major nutritional components like carbohydrate, saturated fat and glycemic index needed to regulate the amount of sugar into the bloodstream as well as maintaining a balanced diet .Based on these parameters ,the fitness of the diet is calculated and genetic operators like selection, crossover and mutation are applied .

The experiment is performed over several generations and sensitivity of this experiment is analyzed with respect to mutation rate and crossover rate.

**Keywords:** Diabetic Diet Schedule , Genetic Algorithm, Chromosomes, Genes, Selection, Crossover, Mutation

**Introduction:**

Diabetes is a group of metabolic disorders in which there are high blood sugar levels over a prolonged period. If left untreated, diabetes can cause [many complication](https://en.wikipedia.org/wiki/Complications_of_diabetes_mellitus)s such as  cardiovascular disease, stroke, chronic kidney disease, foot ulcers, and damage to the eyes. Prevention and treatment involve maintaining a [healthy diet](https://en.wikipedia.org/wiki/Healthy_diet) and regular [physical exercise](https://en.wikipedia.org/wiki/Physical_exercise). In the Diabetic Diet Scheduling Problem , the nutritional composition of all the food items and preferences of user are given . The diet is scheduled in such a way that the total calories provided by carbohydrate  is 40% of the total calories needed and glycemic index(GI) and the saturated fat should be low so as to limit the rate of release of sugar into the bloodstream .The proteins ,vitamins and user preference is maximized to maintain a healthy and hearty diet. The next section depicts some of work done in this area before. In this experiment, we used Genetic algorithm to schedule the diet therefore, firstly an initial population is generated and then genetic operators which are selection, crossover and mutation are applied in sequence for a multiple number of generations and then a graph is plotted between fitness and generation to analyze the results. Afterwards, fitness is plotted against mutation rate and crossover rate.

**Prior Work:**

The Classical diet problem is a 0/1 multi-dimensional knapsack problem with the objective to generate a menu with the lowest cost subject to some daily nutritional requirement constraints defined as lower and upper bounds on nutritional element (e.g. energy, protein, calcium, iron, vitamin A, etc.) intake amounts. A comprehensive review of the multi-constrained 0-1 knapsack problem and the associated heuristic algorithms is given by Chu and Beasley [1].The bi-objective diet problem introduced by A. Kahraman  and A.Seven [2] is similar to the classical diet problem but has an additional objective. The additional objective is to maximize the preference of the user. The user is expected to rate meals according to personal taste. By using the age and gender information input by the user, constraints are determined.Both the above methods uses Knapsack 0/1 problem therefore, it does not include portion or servings of each food item included in diet.

The Scheduling Diet for Diabetes Mellitus Patients by M F Syahputra [3] uses Harris-Benedict equation for the calculation of calories need and then calculates the diet type. The diet is divided in 5 types of food (staple food ,fruits ,vegetables ,side dish, complementary)and all must be included in a meal in specified proportion .So,in this scheduling ,each type of food must be included in the diet which may not be feasible for the patient at every time in real life.The method does not take the preference of user into consideration .A scheduling diet for cancer patient had been conducted by Husain, et al. using Genetic Algorithm (GA) [4]. They succeed to schedule diet with high accuracy, but they limited the data of food and generated it only per day.

This project creates a diet schedule for a weak taking the user preference into consideration as well as the proper amount of each food item included in diet is also specified.

**Technical Description:**

Database:

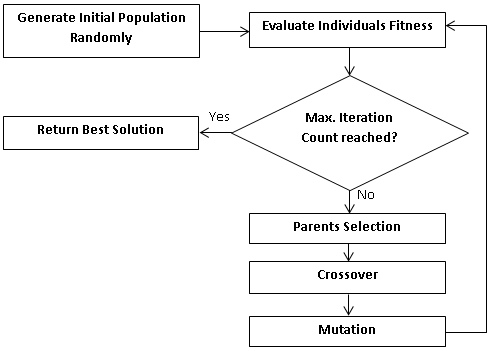
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| parameter | FI1(g) | FI 2 | FI3 |  |  | FI n |
| carbohydrate | 453 | 600 | 800 |  |  | 1000 |
| proteins | 56 | - |  |  |  |  |
| vitamins | 76 | - |  |  |  |  |
| Saturated fat | 560 | - |  |  |  |  |
| GI | 8 | - |  |  |  |  |
| calories | 230 | - |  |  |  |  |
| preference | 5 | 6 | 1 |  |  | 2 |

FI- food Item

This database is given as input to the system which consists of nutritional values of different food item , say per gram.

The database values are used to find the objective functions which in turn are used to find the fitness of each chromosome.

Flow chart for applied Genetic Algorithm :



Chromosome encoding:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **gene**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **x0** | **x1** | **x2** |  |  |  |  |  |  |  |  |  | **xn-1** | **xn** |     **chromosome** |

Constraints:

For a normal person the total calorie needed for a balanced diet in a day should be in range 1500-1800 or 35 cal/kg for the weight of the person.For a diabetes patient ,total calories from carbohydrate must be 40%-60% of the total calories for a healthy diet.

Therefore constraints are:

a)  1500 <=𝝨ci  <=1800

b)  total carbohydrate amount \*4 calories should be between 40% -60% of total calories.

c)    0 <=xi<=1800/ci

Initialisation : Number of chromosomes in the population is selected and random values are generated for each gene between 0 and xi.

*Fitness Evaluation:*

Our objectives are:

1: maximize protein in diet

2: maximize vitamins in diet

3: maximize user preference

4: minimize glycaemic index

5: minimize difference between total carbohydrate and 40 percent of total calories.

6: minimize difference between calories outcome of diet and expected calories.

Objective functions:

obj1= 𝝨protein\*xi

obj2=𝝨 vitamins\* xi

obj3=𝝨preference\* xi

obj4 =𝝨saturated fat\* xi

obj5=𝝨carbohydrate\* xi - 1080

obj6=𝝨calories\* xi -1800

obj7=𝝨GI\* xi x i =value of gene for each food

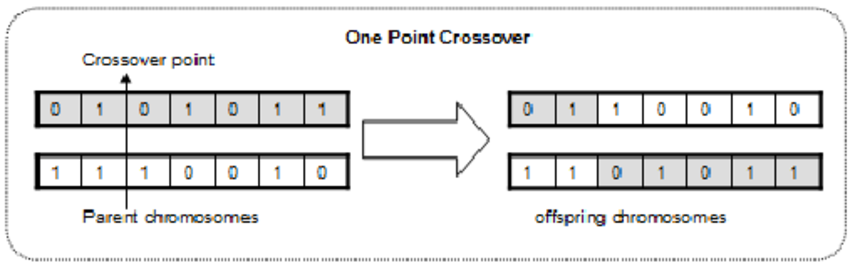
Fitness function=  (w1\*obj1 + w2\*obj2 +w3\*obj3 +w7\*obj7) / (   w5+ w4\*obj4+w6\*obj6 )

Where w1,w2,w3,w4,w5,w6 and w7 are weights which denote the weightage a parameter have in fitness function

The fitness function of each chromosome in population is calculated. Higher the fitness value, higher is the probability for its selection for next generation. For the selection Roulette wheel selection[5] method is used. Then, one point crossover is applied on the new selected population to exploit the search space.

*One- point Crossover*: For each chromosome, we generate a random number and if it is less then the crossover rate then we find the crossover point for two parents. The best features of parents are chosen and are included in the future generations.

The crossover rate(⍴) was set at 0.25 in this experiment.



*Mutation*: Mutation is done to prevent GA from falling into local extreme.

Mutation is the part of the GA which is related to the “exploration” of the search space.

Mutation process is done by generating a random integer between 1 and total\_gen . If

generated random number is smaller than mutation rate(ρm) variable then mark the position

of gen in chromosomes . ρm was set to 0.125 in this experiment.

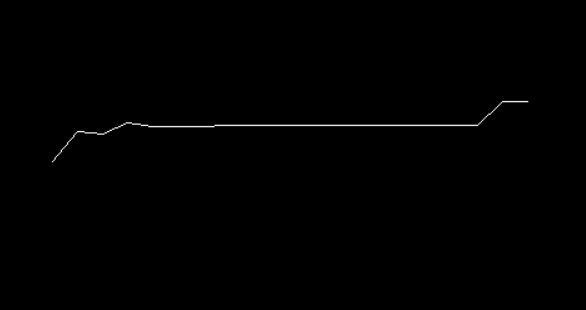
This marks the end of our one generation. After this we will have a set of new chromosomes. These new Chromosomes will undergo the same process as the previous generation of Chromosomes such as evaluation, selection , crossover and mutation and at the end it produces new generation of Chromosome for the next iteration. This process will be repeated until a predetermined number of generations or until the convergence point has been met.

**Experimental Results:**

The experimental results can be analysed using the fitness vs generation graph.

The sensitivity of the experiment is analysed by plotting two graphs that are :

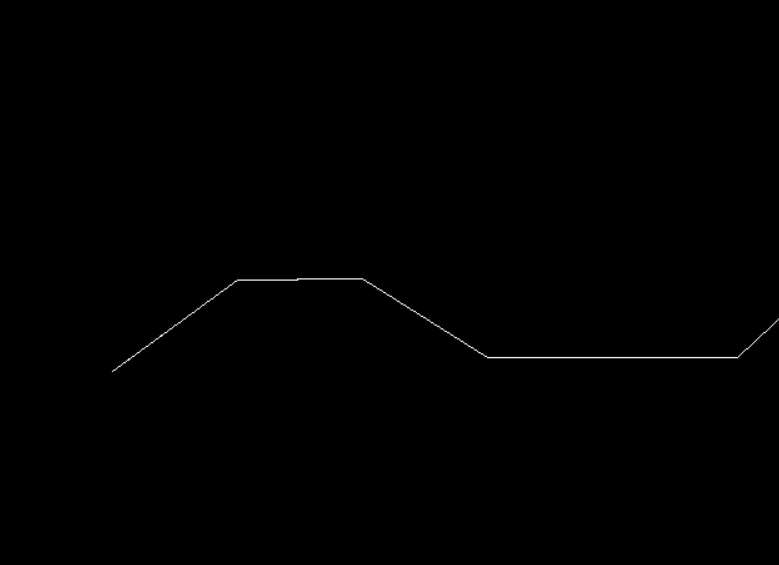
Fitness Vs Crossover rate and Fitness Vs Mutation rate.



Fitness----🡪

Generations-------->

Fitness VS Generations Graph



Fitness----🡪

**Crossover rate ------>**

Fitness Vs Crossover rate Graph.

Crossover rate is varied from 0 to 1 with a gap of 0.15.

****

Fitness----🡪

Mutation rate---->

Fitness Vs Mutation rate

Mutation rate is varied from 0 to 1 with a gap of 0.075.

**Conclusion:** The graph plotted between fitness and the generation in this experiment shows that the experiment did not stuck at local maxima instead it came out of it and achieved a global maxima . So from the experimental results, we can conclude that the suitable crossover rate is between 0.25-0.4(approx.) .From the Fitness vs mutation rate, we see that global maxima is achieved when the mutation rate is low.

**References:**

[1]P. C. Chu and J. E. Beasley, “A Genetic Algorithm for the Multidimensional Knapsack Problem”, Journal of heuristics, 4, Kluwer Academic Publishers: 1988, pp. 63-86.

[2] A. Kahraman, and A. Seven: Healthy Daily Meal Planner, Genetic and Evolutionary Computation Conference – Undergraduate Student Workshop (GECCO’ 05 UGWS), June 25-29, 2005, Washington, D.C. USA.

[3] M F Syahputra : Scheduling Diet for Diabetes Mellitus Patients using Genetic Algorithm   IOP Conf. Series: Journal of Physics: Conf. Series 801 (2017) 012033

[4] Husain W, Wei, L J, Cheng S L and Zakaria N 2011 Application of data mining techniques in a personalized diet recommendation system for cancer patients IEEE Colloquium on Humanities, Science and Engineering Research (CHUSER 2011) pp 239-244

[5] https://en.wikipedia.org/wiki/Fitness\_proportionate\_selection