# A NEW RESOURCE LEVELING MODEL FOR THE CONSTRUCTION PROJECT MANAGEMENT OF BUILDING STRUCTURES: THE GENETIC ALGORITHM

***Associate Prof. Dr. S.Cengiz Yesugey, MSc., PhD***
Dokuz Eylul University, Faculty of Architecture, Izmir-TURKEY
***Dr. Abdulkadir Karakaya, MSc., PhD***
Izmir-TURKEY

**Abstract**

In this article, the use of the genetic algorithm method was examined with regards to the optimization of resource leveling during planning studies performed within the scope of the construction project management for buildings. The aim was to develop a model that would assist in determining how genetic algorithms can be used in this field In addition, the developed model was also capable of solving problems in case the building projects consisted of one or several sub-projects. In sum, the aim of the study was to develop a model for leveling resources for cases consisting of single or multiple projects within the context of planning activities for building construction project management based on following assumptions:

- The project is time constrained, but no limitations are involved with respect to the availability of resources.
- One type of resources is used during the activities (money and workforce)
- Leveling methods are used (the heuristic model and the genetic model).

The Critical Path Method (CPM) was used for illustrating the operations performed in the provided examples, and to calculate the planning times. Two different computer programs were developed for the heuristic algorithm model (the traditional method) and the proposed genetic algorithm model, and the two models were compared in the given/evaluated cases. The results of these comparisons were illustrated using tables.

**Keywords:** Genetic Algorithm, Project Management, Resource Leveling, Optimization.

## Introduction

The literature defines project as the combination of interrelated operations/activities that need to be completed in a particular order to ensure the completion of an entire task/work. Performing the activities that constitute a project requires both time and resources. In other words, a project is an activity whose beginning and completion is expressed in terms of the resources involved, and which seeks to achieve certain clearly defined goals according to specific budget and time constraints. In this context, it is necessary to examine how it will be possible to ensure that the processes and resources of a project are compatible and consistent with one another, and, more importantly, to determine how processes and resources will allow the project goals to be achieved. For this reason; the definition, preparation, management and control of project data to this end is required. These features constitute the concept of management. Management consists of planning, organization, order-command (execution), coordination and control-correction phases. When planning of deterministic projects (i.e. projects in which the relevant activities will be absolutely performed within the determine timeframe); the Critical Path Method (CPM) (Fig. 1) and Precedence Diagram techniques will be used for projects without similar activities where the process times are definite; while the Program Evaluation and Review Technique (PERT) will be used for projects without similar activities where the process times are not definite. For projects with similar processes; the Linear Scheduling Method (LSM) will be used in those that are linear/non-linear shaped, while the Line of Balance (LOB) or the Vertical Production Method (VPM) will be used in those that are horizontal-vertical shaped. In studies on resource balancing, the CPM and Precedence Diagram are the generally preferred techniques.

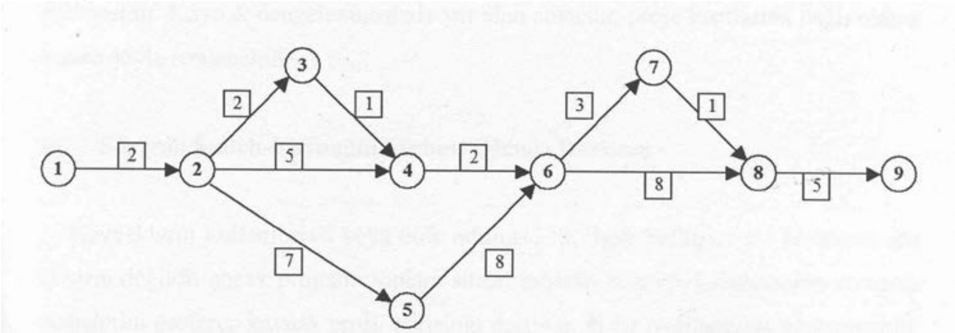



Figure 1. A Network Diagram in CPM

**Resources and the purposes of resource leveling**

During a project; basic resources such as workforce, machine power, materials, and time are used for production. Within the context of construction projects, these basic resources interact with one another in various different ways. During the definition and planning of tasks, it necessary to level resources with specified constraints (if any). The graph in

Figure 2 illustrates the time-dependent change in resource utilization. Based on the applicable project constraints, the goals of resource leveling can be listed as follows:

***A) Cases where Time is Limited, but Resources are Freely Available:***

Although there is no limitation regarding the use or collection of resources, the total duration of the project is constant. Various equations can be found in the literature regarding resource leveling (or resource smoothing) that is performed in such a context (Eyeci, 1985).

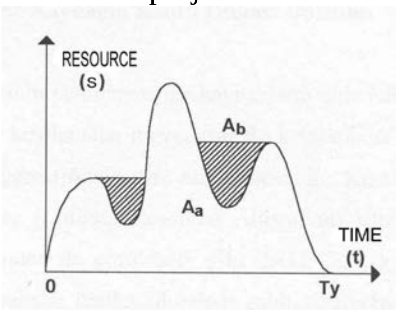


Figure 2. Resource Utilization

***B) Cases where Time is Freely Available, but Resources are Limited.***

During the planning of the project’s production processes, there are constraints with regards to the collection and/or use of resources for production processes. These constraints can affect the total duration of investments, and may even increase this duration when necessary. Based on resources profiles obtained through calculations relating to resources leveling (i.e. Limited Resources Allocation) (Fig.3), resources are not able to exceed their constraints. During production processes, the constraints might be constant, variable or periodic (Özdemir, 1988).

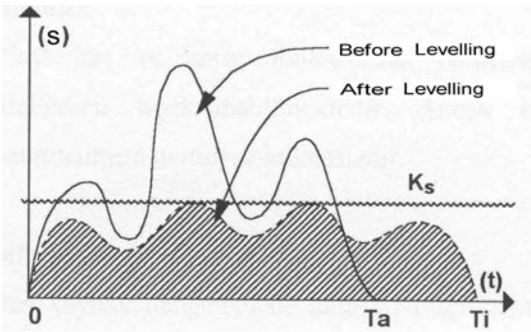



Figure 3. Cases where Resources are Constrained, but Time is Freely Available

***C) Cases where both Time and Resources are Freely Available (Determining the Optimum Investment Period)***

During resource leveling for such cases, the main goal is to identify the investment period with the minimum expenses (Time Cost-Trade Off). Thus, resource leveling attempts, for such cases, to minimize the indirect time-dependent expenses that occur in parallel to the production process during investments. In other words, the goal is to accelerate the production process. No limitations are taken into consideration regarding time and resources. Li and Peter (1997) have previously performed a study on this subject with genetic algorithms.

## Solution Techniques

***A) Analog Methods***

The approach towards a possible solution is intuitive. In this context, methods such as physical methods (e.g. a model for the problem), electrical analogs and dual series are utilized. Employing these methods is not always possible.

***B) Analytical Methods***

Mathematical models can be utilized in cases where the constraints and the objective function of the problem can be determined. However, defining and solving large-scale problems by using this method is nearly impossible.

***C) Heuristic Methods***

In the abovementioned resource leveling, the solution of the problems is determined heuristically (intuitive, postulate, heuristic) by using equations mentioned in the objectives. The studies include an algorithm structure, the activities are listed and/or assigned according to previously accepted assumptions, and the planning studies are conducted with the support of computers.

In practice, the study that has the most validity for this context is that of Burgess and Killbrew (1962). The amount of resources used during the project in terms of time units is considered as a measure of the total of separate sum of squares. All these processes are first planned during the early initiation period. In the process list; the processes will be screened towards the right in the distribution of the amount of resources, within the total abundance which we can consider as a postponing. Based on the calculations, the day that indicates the total of the smaller squares for each process is considered as the new starting time of the said process. Tulip (1972), on the other hand, demonstrated that when screening with the sum of squares, the full abundance should be screened for each process, and that the smallest sum should be identified. Eyeci (1985) showed that, on condition that the postponing of processes remains limited, the retraction of processes will have positive results on the sum of squares and the concentration ratio. Yüksel (2000) noted that in case this retraction process was performed with more than one optimum value for the sum of squares, the process would be planned according to the maximum value in the resource profile. Thus, the heuristic method aims to perform resource leveling by minimizing the sum of squares, and the algorithm structure that encompasses all of the abovementioned study is, in its final form, known as the “mixed heuristic method.” This method has been used in various studies and in resource leveling activities relating to single and multiple project groups. As can be seen in the literature, different heuristic studies have been conducted based on scenarios in which the resources are constant, limited or variable.

**Genetic algorithms**

Generally speaking; genetic algorithm refers to the general-purpose solution and optimization method that is based on Darwin’s evolution theories and genetic science. There are constant efforts to improve the computer simulations of this method such that they provide better solutions to a problem by removing potential negative solutions. A method known as “Evolution Strategies” was first designed by Rechenberg towards the end of the 1960s to optimize real values (Rechenberg 1973). During this period; Fogel, Owens and Walsh (1966) also developed the concept of “Evolutionary Programming.” Holland and students in Michigan university further developed genetic algorithms during the 1960s and 1970s. Following the publication in 1975 of Holland’s book entitled the “Adaptation in Natural and Artificial Systems,” which provided the results and finding of his previous studies; the method Holland came to develop was referred to as genetic algorithms (or GA) (Holland, 1975).

Nowadays, the use of genetic algorithms in various different fields of science such as business, engineering, economy, electronics and mathematics is being further investigated and developed (Işçi, 2002); Emel and Taşkın, 2002 ); Şen, 2004).

**Presentation and Concepts (Gene, Chromosome, Generation, Fitness Function, Objective Function)**

Genes are the smallest units of genetic algorithms are represents values that belong to a mathematical variable. A chromosome is a series of genes (individual). It represents the full set of possible solutions for a problem. A set of chromosomes is called generation. For the initial generation, chromosomes are formed randomly. For chromosomes, the binary or decimal coding is used. In the work program with 8 activities (genes) shown in Table 1, the beginning of each work is coded onto a chromosome using a decimal system. When the beginning generation is randomly formed, numerous different work programs (chromosomes) will be obtained.

The objective function is used in order to calculate the total score of the chromosome. In the objective function, values that are taken into consideration are evaluated according to the applicable constraints and objectives. Constraints are equations, inequations and functions that clearly demonstrate the negativities within the objective of the problem. Priorities can be listed in terms of constraints such as time, resources, costs, capacity, etc. Objectives are equations, inequations and functions such as total duration, total cost, etc. that are used for obtaining a global solution to a problem. In addition, the constraints and objectives can be calculated separately according to their suitability to the problem, and allow the total score for the solution (chromosome) to be determined. For example; the sum of squares of the resources used in each time unit of the resource profile is used as a purpose function in the literature, and to determine the minimum solution.

In the literature, fitness is described as a grading function that assesses the level of success of each individual in a generation in resolving a problem. Regarding genetic algorithms, the following sources might be useful: Beasley (1993-a, 1993-b), Busetti (unknown date), Dianati et al. (2002), Rennard (2000), Zbigniew (1996) and Goldberg (1989). The problem parameters are measured through coding, and are used as introductions into the fitness function. They are then listed with the selection probability from the fitness levels (roulette wheel, etc.) and randomly selected. The individuals show development in the solution set (generation) through reproduction, crossing and mutation operators. Another approach described that the fitness of an individual for certain problems can be determined based on the error between the results obtained for the individual at the predicted results. In individuals with greater fitness, this error value is closer to zero. This error is generally calculated with the mean or sum of the combinations that of the entry will be presented again. The correlation factor between the expected and produced valued is used to calculate the fitness value.

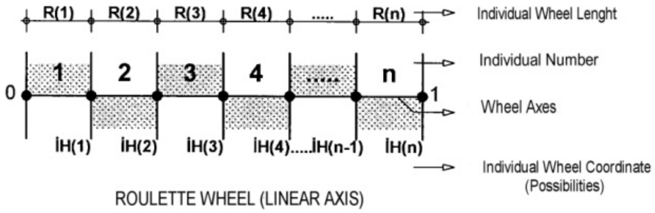
**Operators (Crossing, Mutation, Selection)**

Although they tend to differ according to the objective of the problem, genetic algorithms are used to form a solution group (the starting generation), and the successful solutions within this solution group will be selected. The selection operator is, brief, the selection of individuals within the generation (Mother, Father). Selection methods such as the Roulette Wheel, Tournament Selection and Elitism may be used. Table 2 provides, as an example, the objective (H(i)) and fitness (F(i)) values and the probability

(R(i)) and the consecutive total probability (IH(i)) values for the mother and father candidates who were selected randomly according to the rate of random crossing. The shapes of the axes and the distribution of the mothers and father on these axes are shown in Figure 4. Crossing is the formation of a new solution (children) by using the structures of two currently available solutions. Mutation is the generally the random changes in the genes of individuals. The performance of the genetic algorithm significantly affects the selection, crossing and mutation operators and the ratios of crossing and mutation. In the genetic coding of the work program provided in Table 1; a mutation was performed by cutting between Gene 5 and Gene 6, crossing the data of these genes with one another's, and by randomly changing the value for Gene 5. Based on this operation, new work program solution can be obtained for the current generation. Figure 5 and Figure 6 belong to a genetic algorithm that uses a total of 100 Generations, with 25 individuals (chromosomes, work programs) per generation, to perform resource leveling for a work program. It is required that the objective function is at its minimum value. It can be seen that, when the genetic algorithm is used for 25 different work programs for 100 generations, all solutions develop together towards suitable minimum values. The solution that is sought is generally the minimum solution obtained based on all genetic calculations. (For example: identifying the plan, work program or individual that requires the least work force).

Table 1. Example of a Simple Genetic Coding in a Work Program

| Work Program Coding (Gene Definition) | G1 (Gene Number) | 1 (Work Starting Time) | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Chromosomes (n number) | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 |
| | 1 | 3 | 4 | 8 | 6 | 9 | 11 | 14 | 1 | 2 | 5 | 5 | 6 | 11 | 9 | 13 |
| | BEFORE | | | | | | | | AFTER | | | | | | | |
| Crossing (Single-Point, Piece Change) | Chromosome (mother) | | | | | | | | Chromosome (child 1) | | | | | | | |
| | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 |
| | 1 | 6 | 8 | 7 | 10 | 9 | 9 | 12 | 1 | 6 | 8 | 7 | 10 | 10 | 11 | 14 |
| Crossing (Single-Point, Piece Change) | Chromosome (father) | | | | | | | | Chromosome (child 2) | | | | | | | |
| | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 |
| | 1 | 2 | 5 | 8 | 6 | 10 | 11 | 14 | 1 | 2 | 5 | 8 | 6 | 9 | 9 | 12 |
| Mutation (Random Change) | Chromosome | | | | | | | | Chromosome | | | | | | | |
| | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 |
| | 1 | 3 | 4 | 8 | 7 | 7 | 10 | 13 | 1 | 3 | 4 | 8 | 9 | 7 | 10 | 13 |



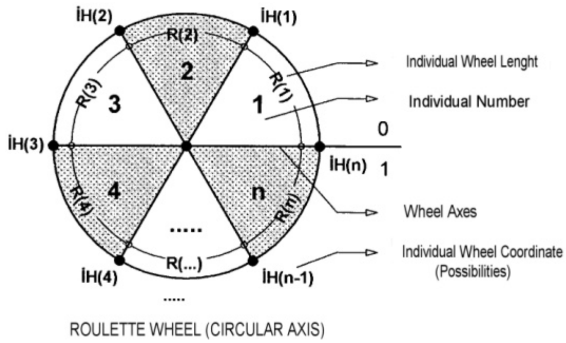




Figure 4. The Possibility Distribution Axes for Crossing Selection

Table 2.Mother and Father Crossing Values fort he Current Generation

| Moth.Fath.R(n) | (n) | Z(i) | F(i) | R(i) | İH(i) |
|---|---|---|---|---|---|
| 1 | 1 | 638 | 13,652 | 0,081 | 0,081 |
| 2 | 2 | 656 | 13,277 | 0,079 | 0,159 |
| 3 | 3 | 658 | 13,237 | 0,078 | 0,238 |
| 4 | 12 | 684 | 12,734 | 0,075 | 0,313 |
| 5 | 10 | 672 | 12,961 | 0,077 | 0,390 |
| 6 | 14 | 690 | 12,623 | 0,075 | 0,464 |
| 7 | 6 | 666 | 13,078 | 0,077 | 0,542 |
| 8 | 11 | 678 | 12,847 | 0,076 | 0,617 |
| 9 | 15 | 692 | 12,587 | 0,074 | 0,692 |
| 10 | 5 | 660 | 13,197 | 0,078 | 0,770 |
| 11 | 13 | 688 | 12,660 | 0,075 | 0,845 |
| 12 | 8 | 688 | 13,039 | 0,077 | 0,922 |
| 13 | 4 | 660 | 13,197 | 0,078 | 1,000 |
| Total | | 8710 | 169,089 | 1,000 | |

Figure 5. Objective Function Solution Space (Three Dimensional Appearance)

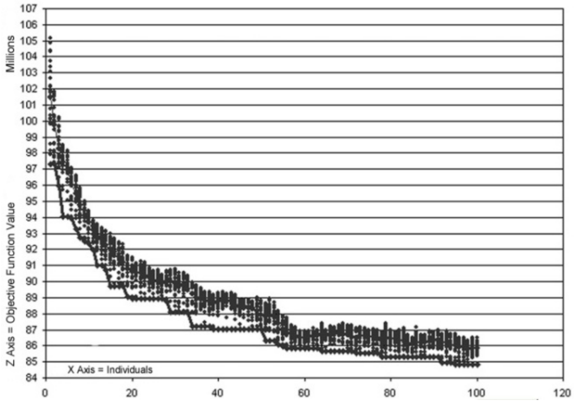
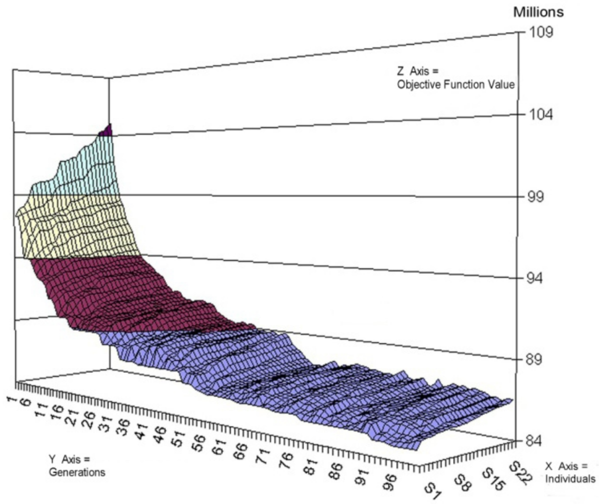



Figure 6. Objective Function Solution Space (Two Dimensional Frontal View)




## Generic algorithm model for the resource leveling

Based on data from the literature, the following algorithm was developed (Fig.7). The reproduction multiplier (which determines the number of new individuals who will be obtained based on the number of individuals in the initial generation and crossing), the mutation rate, the crossing rate and the generation number will first be obtained from the researcher, after which the random initial generation shall be formed (WhichGeneration=0).

Following this, the current generation number shall be increased (Which Generation=WhichGeneration + 1), and a listing shall be performed according to the objective function or the fitness function. To hide the best solution among all of the generations, the most suitable individual (solution, chromosome) in this listing shall be assigned to a variable. As the total number of individuals in the generation is constant, excess numbers of new individuals

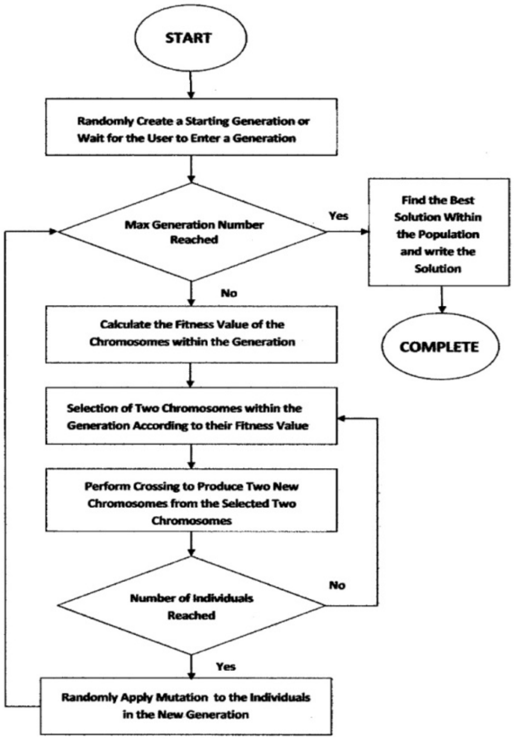



Figure 7. Algorithm Flow Chart

obtained from the processes of the previous generation are removed. In case the maximum number of generation is reached, the algorithm will be completed and the best solution will be obtained; otherwise, the crossing and mutation operators will come into play after the selection.

To perform a listing; the process will return to its second step, and the cycle will be continued/repeated. The algorithm steps are as follows:

***Step1:*** *TakeAlgorithmData*
*FormStartingGeneration*
*WhichGeneration=0*

***Step2:*** *WhichGeneration=WhichGeneration + 1*
*CalculateObjectiveFunction*
*PerformListing*
*HideandShowBestSolution*

***Step3:*** *RemoveasmuchasReproductionMultiplier*
*(If) WhichGeneration=MaximumGenerationNumber (Then)*
*CompleteCalculation ShowBestSolution: End*

*(Else) GoTo Step4*
*End If*
***Step4:*** *MakeSelection*
***Step5:*** *PerformCrossing*
***Step6:*** *PerformMutation*
***Step7:*** *GoTo Step2 'Continue Cycle.*

In case the amount of time available is constant and resources are freely available during the project planning process, the use of one type of resource is considered as a constraint for the processes. The most suitable solutions are then sought for the distribution of resources. The algorithm will be use in the planning of single and multiple projects. The number of generations, number of individuals, mutation rate, crossing rate, reproduction rate, crossing type, selection, elitism, exponential acceptance values are used defined and entirely parametric. This model and algorithm, which was prepared for educational purposes, was written using the Visual Basic program, and then employed in a set of different experiments.

**Definitions, Assumptions, Variables and Function of the Model**

The objective function indicates the individual with the smallest sum of resource squares among all individuals within a generation. Owing to the operations performed with the objective function for all of the generations; the genetic algorithm can identify the individual with the smallest resource squares. On the other hand, the resource squares method is also used for measuring the irregularity of resource distribution in the mixed heuristic model. In brief, the objective function is determined by common equation structures that allow the comparison of two different models.

Individuals with smaller total resource squares will acquire higher fitness values when the fitness functions provided below are used. Based on the fitness value of individuals, the roulette (possibility) wheel length was determined for each individual, and the sum of these values was used to identify the roulette wheel coordinate of each individual.Certain numerical results obtained using the equations below can be seen in Table 2. and Fig.4.

**Definitions**

| | | |
|---|---|---|
| N | = | Total Number of Individuals in a Generation |
| n | = | Total Number of Individuals in the Reproduction Pool |
| i | = | The Starting Nodal Point of the Process |
| KT(i) | = | The Resource Square Sum of an Individual |
| Z(i) | = | The Objective Function Value of an Individual |
| Z | = | Objective Function |
| Ztop | = | The Total Objective Function of the Generation (Mother and Father) |

F(i) = The Fitness Function Value of an Individual (Mother and Father)
Ftop = The Total Fitness of a Generation (Mother and Father)
R(0) = Roulette Wheel Origin
R(i) = The Roulette (Probability) Wheel Length for an Individual
Rtop = Total Wheel Length
IH(i) = Wheel Coordinate for an Individual
IH(n) = Total Wheel Length

J = The Ending Nodal Point of the Process
IB = The Starting Time of the Process
TE = The Early Completion Time of the Node
TG = The Late Completion Time of the Node
TiE = The Early Starting Time at Node i of the Process
TjE = The Early Completion Time at Node j of the Process
TiG = The Late Starting Time at Node i of the Process
TjG = The Late Starting Time at Node f of the Process
Tij = Duration of Process

**Constraints**

The antecedent and consecutive relationships between the processes and the total duration of the project shall not change during the calculation.
$n \leq N$ (whole number), $0 < n$ (whole number), $0 < i \leq n$ (whole number),
$0 < R(i) \leq 1$, $0 < IH(i) \leq 1$

**Objective Function:**

**Z(i)** = KT(i); **Z** = min (Z(i)), $0 < i \leq n$

**Other Functions and Assumptions:**

$$\mathbf{Ztop} = \sum_{i=1}^{n} Z(i), \quad \mathbf{F(i)} = Ztop/Z(i), \quad \mathbf{Ftop} = \sum_{i=1}^{n} F(i), \quad \mathbf{R(0)} = 0, \quad \mathbf{R(i)} = F(i)/Ftop,$$

$$\mathbf{Rtop} = \sum_{i=1}^{n} R(i) = 1$$

$$\mathbf{IH(i)} = \sum_{i=1}^{i} R(i), \quad \mathbf{IH(i)} = R(i)+IH(i-1), \quad \mathbf{IH(n)} = 1= \ Rtop$$

## Chromosome Structure

Chrosome structures were evaluated in the literature, and it was observed that different studies used different assumptions for these structures. It is generally accepted that, in most cases, the chromosome structure is simple and linear. In this model, each chromosome is arranged in a linear way according to the starting time and non-puppet processes. For this reason; each chromosome represents a work program, and the total number of chromosomes (individuals) in the generation indicates the number of different planning types used for the project in question. In the different models identified while reviewing studies in the literature, it was observed that a single chromosome structure per individual was used in most cases. In the recommended model, on the other hand, a genetic algorithm structure

Table 3. Explanation of the genetic algorithm model

| | |
|---|---|
| **Coding of the Work Program (Gene Definition)** | GNo \| IB \| Tij / TiE \| TjG<br>Gene No. (FNo=Process<br>Process Start Time: i node TE<br>Process duration: j node TG |
| **Individuals (with Multiple Chromosomes) (1 Individual = 1 Work Program)** | BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>BSNo= Individual No, Amax= Maximum Amount of Resource Used for the Individual<br>KrNo= Chromosome No., G1....G5= Genes |
| **Crossing (On a Chromosome Basis)** | BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>Individuals (2 Indv., Mother and Father) (E.g.: With 2 Chromosomes)<br>SNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>Single-Point Crossing, Single Gene Change (Type 1) (KrNo=1-GNo=3, KrNo=2-GNo=4) (Child 1, Child 2)<br>BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>Single-Point Crossing, Piece Change – Multiple Gene Change (Type 2) (KrNo=1-GNo=2-3, KrNo=2-GNo=3-4) (Child 1, Child 2)<br>BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>Double-Point Crossing, Partial Piece Change – Multiple Gene Change (Type 3) (KrNo=1-GNo=2-3/4-5, KrNo=2-GNo=1-2/4-5) (Child 1, Child 2) |
| **Mutation (On a Chromosome Basis)** | BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>Individuals (Example: 2 Chromosomes)<br>BSNo \| Amax \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5 \| KrNo \| G1 \| G2 \| G3 \| G4 \| G5<br>Random IB Change, (Type 1=Random / Type 2=Amax important) (KrNo=1-GNo=2, KrNo=2-GNo=3) |

that differed from the multiple project solution approach was used. Each sub-project in each individual is defined by a different chromosome. Thus, each individual has a large number of chromosomes (equal to the number of chromosomes). Structurally, the chromosomes can be arranged consecutively in a linear fashion, or be arranged side-by-side in bundles.

Operators such as crossing, mutation were performed at the level of chromosomes (i.e. between the sub-projects). In the computer program based on the prepared genetic algorithm, the individuals were generally considered as having multiple chromosomes. On the other hand, an individual with a single chromosome represented an exceptional case in the model. Table 3 below provides descriptions for certain concepts that are part of the proposed genetic algorithm.

**Forming the Starting Generation**

In genetic algorithm studies, a general assumption regarding the starting generations is that they were formed randomly. However; in case there is a solution interval that is expected or considered suitable, the first individuals can be formed mainly in that interval. This, in turn, will have a positive effect on the work period of the algorithm. In the problem being considered, predicting a suitable solution or interval beforehand is almost impossible. It is hence suitable to form the starting generation randomly. For these reasons, the processes for an individual within the starting generation of the proposed model was started earlier, while the remaining individuals within the generation were all planned randomly.

**The Values of the Objective Function and Fitness Function**

The objective function and fitness function values of each individual (work program) were determined using the functions above. The listing of the individuals within the generation is important. This listing is performed for each generation during the evaluation phase. To reduce the algorithm calculation period for the listing processes; the listing algorithms that are accepted and used in the literature were reviewed, evaluated and applied to a suitable listing algorithm.

**Selection of Couples**

The selected number of couples was equal to the crossing rate. During the selections of couples (mother and father; parents), two different assumptions were made and experiments were performed. In the first approach; the 25% best in the listing of individuals would be transferred during the selection of couples for upper crossing, while the rest would be selected randomly. In the second approach, the couples would be selected entirely randomly within the generation. During the experiments, it was observed that the first approach had a positive effect in the determination/finding of appropriate solutions for the genetic algorithm.

**Elitism**

The most suitable individual in the current generation can undergo change in the following generation due to mutation, causing the individual's fitness function value to worsen. For these reasons, the elitism operator was added to the model, and various experiments were performed with it. In the trials performed with the elitism factor, a constant progress was observed in the objective function of the generations, although no breaking was observed. In conclusion, the transfer of the fittest individual in the current into the following generation (i.e. elitism) was accepted and used by the proposed model.

**Reproduction (Crossing Rate, Reproduction Rate)**

Various experiments were performed in order to increase the diversity of the solution space. The crossing rate is parametric. It is accepted as 0.85. The crossing rate was used for obtaining the total number for the mother and father. It was decided that the number of children resulting from the reproduction rate concept should be parametrically controlled by the user. In a generation with 100 individuals, and with a crossing rate of 0.85; there are a total of 100x0.85=85 mothers and fathers. If the reproduction rate is taken as 100; 100x1.0=100 children will result, and the algorithm will have 100/2=50 crossing cycles, which will be performed with the list of 85 mothers and father by using the roulette technique. In each cycle; a mother and father will be selected and crossed, resulting in 2 children. As a result of crossing, the total number of individuals will increase from 100 to 200. With crossing, the genes of each chromosome will be transferring information to one another. The assumption here is that corresponding processes (genes) is changed with the corresponding starting time (IBcurXIBgen=IBnew). What is important here is that it should be possible to assign the starting time of the work (IBgen) to the current nodal times (TiE, TjG). If IBgel is smaller than the TE at node i (Early Completion Time), the IBnew (Starting Tim of Work)=TiE; and if IBgen+Tij (Process Duration) is greater than TG at node j (Late Completion Time), IBnew=TjG-Tij. For this reason, the genetic calculation performed regarding the total project period does not change and always remain constant. Three types of crossing are recommended. Single-point crossing (gene change), single-point crossing (piece change), and double-point crossing (piece change) (Table 3).

**Mutation**

The mutation rate is parametric. It is accepted as 0.05. Two different approaches have been accepted and adopted. The first approach consists of randomly determining the new starting time of a gene (processes) based on the total abundance of the planning times at the moment, and by excluding the current starting time of the work. In the second approach, the current maximum resource day of the relevant individual on the chromosome to be mutated is important. In this mutation type, the starting time of the process will be shifted further left if it is already at the left of the starting time of the maximum resource day, and further right if it is at the right of the of the starting time of the maximum resource day. However, in case the current abundance and the starting and ending times of the project are not suitable, the mutation process will be performed in the reverse direction. The purpose of this approach is to prevent the piling of assigned tasks/processes to the time periods that are closer to the maximum resource day. However, as the maximum resource day can change in every generation, it will continue to prevent piling according to the new situations of the mutation source profile. In trial performed with both approaches, it was observed that the second mutation type was more successful.

**Approach Testing**

In the model, the completion of the algorithm is, as it is generally described in the literature, based on the total number of generations. This number is entered parametrically into the computer program by the researcher. For the interim evaluations to be performed, the program can interrupt the algorithm and evaluate current solutions. The algorithm can then continue where it left. In addition, the addition of an approach testing to the model was not considered necessary.

**Completion of the Calculations**

With the completion of the genetic algorithm calculations; the objective function, fitness function information (individuals, all generations), and all types of technical information regarding the solution (nodal times of the processes, the work starting times, the work program graph, the resource profile, the cost summary table, etc.) were listed.

**Testing of the genetic algorithm model**

To test the proposed genetic algorithm model, it was necessary to compare it with a valid method. To this end, the proposed model was compared with the "mixed heuristic model," which is currently the most commonly used model in resource leveling work programs. To this end, two separate computer programs were prepared for the genetic algorithm and the mixed heuristic models. The correctness of the calculations performed by these two computer programs was tested based on the five conditions listed below. The total project duration, the completion of the work (time, resources), and the constancy of the total resources used:

- The constancy of the relationships between the project tasks,
- The starting and ending times of the processes based on calculation; the necessity for the initial state (IB=0) to be equal to or between the nodal times

for that particular process (TE0, TG0);

- The necessity for the starting and ending times of the processes (determined based on calculations) to be equal to or between the nodal times (TE0, TG0) of the initial state (IB=0);
- In the balanced state, the constancy of the beginning and completion of the nodal times relating to the relevant process.

Objective function has been defined in different ways in the literature. As the resource squares method was, in this context, used within the mixed heuristic method, it was also considered as the objective function. It was thus ensured that the performance of the resource profiles in both methods was tested based on similar criteria. The fitness of an individual was determined based on the ratio between the sum of resource squares for all individuals in the generation and the total of the resource squares for the individual. Therefore, individuals with lower total resource square values had higher fitness levels. In brief, the reciprocal multiplier of the objective function was considered as the fitness function. In the literature, there are studies using more than one equation for the objective and fitness functions. Studies have also been conducted on multiple resource use. To clarify comparisons; information is briefly provided below regarding the "Mixed Heuristic Model" algorithm.

**Mixed Heuristic Model**

To test the correctness of the model, the algorithm information regarding the mixed heuristic model used in the evaluation section is provided below (Yüksel, 2000). In the algorithm, the suitability of the resource profile is determined by using the resource squares method, and the processes are conducted in sequence. Using this algorithm, resource leveling can be performed for single-resourced project plans. The algorithm steps are as follows:

**1.** In case the completion nodes of the processes (j) and they are equal, the beginning nodes will be tested from smaller to larger according to (i).

**2.** All processes will be initiated according to the earliest starting times.

**3.** The process at the lower end of the list will be taken, the full abundance will be scanned, and the process will be used such that it will provide the smallest value for the total resources square (if there is more than one optimum, the first processes will be placed to the rightmost possible position in order to increase their postponing limits). The processes abundance and the starting and completion times will be changed. To determine the upper limit for the postponing of processes in the upper end of the list, the earliest completion time of the process' beginning node will be changed.

**4.** For all processes from the end to the beginning of the list, the $3^{rd}$ step will be repeated.

**5.** As long as there is a decrease in the sum of squares, step 3 will be repeated to rescreen the list. The day (time) at which the maximum resource was used will be determined in the screenings.

**6.** The processes from the first process of the list to the last shall be withdrawn by remaining limited to the abundance amounts to which they have been shifted in the previous steps, and the screening that decreases the sum of squares shall be continued in the opposite direction. In cases there are more than one situation that provides optimum values during the screening process, a different approach will be followed to increase the resource utilization rate. In case the starting time of the process corresponds to a date preceding the day in which the maximum resources were used; the process will be placed to the right (or left if the right is not possible), and the screening will be continued until no decrease is observed in the sum of squares.

**The genetic algorithm and mixed heuristic models in project planning testing with examples**

The prepared computer programs can performed the following calculations in different ways depending on the number of projects and the parametric nodal times of all of the projects:

- The situation 1 and 2 for a single project,
- The situation 1 and 2, or more than one project (multiple projects) being conducted simultaneously.

Situation 1= All projects start simultaneously, and the completion times are not equal.

Situation 2= All projects start simultaneously, and all completion times are constant.

In all of the evaluated studies; a single chromosome structure was recommended for each solution element (individual). In addition, the models for the single and multiple projects were also different. In our case, a multiple chromosome structure was developed for the proposed model. In other words, there is a corresponding chromosome for each project planning. As a result, one individual may be composed of more than one chromosome. In our case, each chromosome is a sub-project planning within a multiple project. A single project and single chromosome definition represents a special case. The crossing and mutation operations function separately for each chromosome within an individual. It was observed that the assumption that the operators should function on the basis of chromosomes gave better results in model test studies. The prepared mixed heuristic model algorithm-based program can produce solutions for evaluated cases that consist both of single or multiple projects. The examples were transferred to computer programs after being organized as a "Project Network Diagram" (Fig.8).

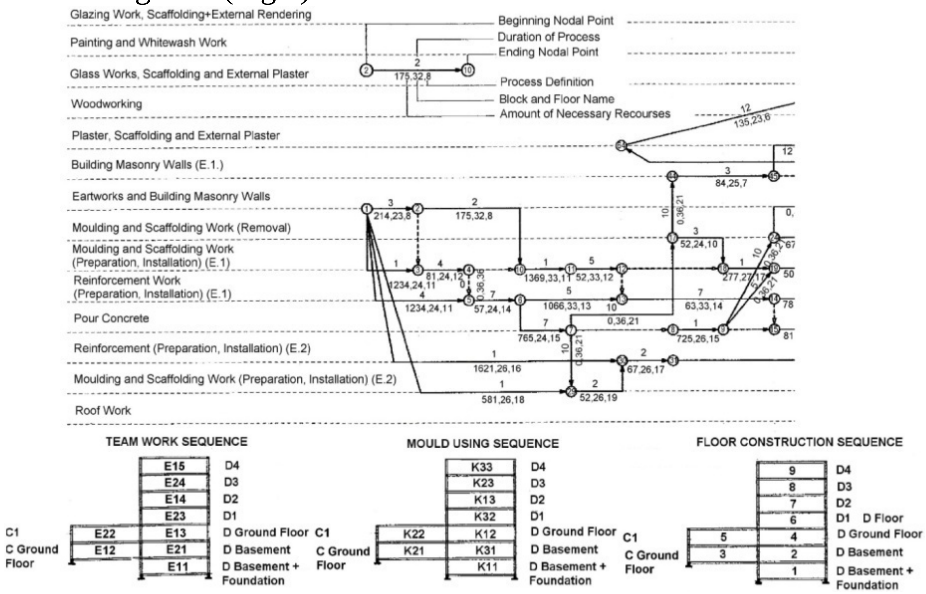



Figure 8. Sample simple project network diagram (Karakaya, 2007)

## The Testing of the Genetic Algorithm Model in the Planning of Single Projects

To test the genetic model for single project planning, the data set previously used in Kocatürk's (1987) study was employed. The project consists of two blocks, with Block C representing a maternal school, while Block D represents a hotel (Table 4). In the planning of each process, the aim was to ensure a steady flow of resources (budget) and the listing of resources based on daily expenditures. The project objective function for the sample single project is given below (Fig.9).

Table 4. Levels for the sample single project (Kocatürk,1987)

| (m2) | Basement | Ground Level | Suspension | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|---|---|
| BLOCK C | - | 628 | - | 472 | - | - | - |
| BLOCK D | 444 | 420 | 200 | 420 | 420 | 420 | 420 |

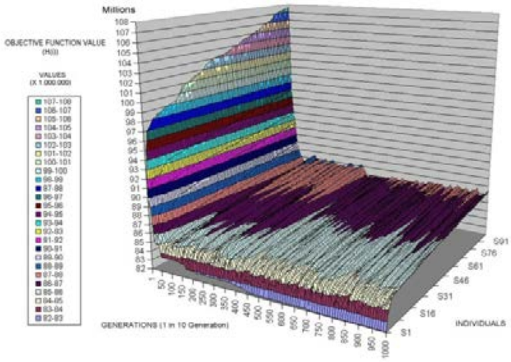



Figure 9. Project objective function for the sample single project (Karakaya, 2007)

## Comparison of the Results of The Model

The results obtained based on comparisons is provided in the table below (Table 5). The evaluation column below provides the percentage difference in the results obtained with the genetic algorithm model in comparison to the mixed heuristic model. The increase in positive values reflects the numerical decrease in the negative values solutions.

Table 5. Comparison of calculation models for the sample single project) (Karakaya, 2007)

| NO | Name Of Account | Heuristic Model | Genetic Model | Difference (%) |
|---|---|---|---|---|
| A) | Resource Squares Total: | 88560089 | 82062067 | -7.337 |
| B) | Available Amount of Resources: | 201112 | 206101 | 2.481 |
| C) | Amount of Used Resources: | 102119 | 102119 | 0 |
| D) | Maximum Amount of Resource: | 2614 | 1831 | -29.954 |
| F) | Maximum Resource Days: | 49 | 19 | --- |
| E) | Mean Amount of Resources: B/T | 1105.011 | 1132.423 | 2.481 |
| G) | Piling Ratio: D/B >= 1 | 2.366 | 1.617 | -31.657 |
| H) | Percentage Resource Use: C/B <= 1 | 0.508 | 0.495 | -2.559 |
| I) | Project Duration (T): | 182 | 182 | 0 |

Compared to the solution obtained with the heuristic model, the genetic model's solution exhibited:

- A 7.337% decrease in resource squares.
- A 2.481% increase in the available amount of resources.
- A 29.954% decrease in the maximum amount of resources, while the maximum resource days were 19.
- A 2.481% increase in the mean amount of resources.
- A 31.657% decrease in the piling ratio.
- A 2.559% decrease in resource utilization.

In sum; the total resource squares, the maximum amount of resources and the piling ratio decreased. The mean amount of resources increased. The available amount of resources increased. The percentage of resource utilization increased as well.

## The Testing of the Genetic Algorithm Model in the Planning of Multiple Projects

To test the genetic model for multiple project planning, the data set previously used in Eyeci's (1985) study was employed. The descriptive and summary information regarding this data set is provided below. In the example considered here; Project 1 is an administrative building with 3 Blocks, one being an administrative block (Block A), the second being a dining hall for civil servants (Block B), and the third being a research building (Block C). Project 2 is a branch-type social services building consisting of 3 Blocks, the first being the dining hall and dormitory for workers (Block A), the second being a dining hall and dormitory for civil servants (Block B), and the third being a control building (Block C). Project 3 involves a public housing building consisting of a single block. These three separate project envisage and plan the building of three groups of buildings (Table 6).

Table 6. Levels for the sample multible project, (Eyeci, 1985).

| **PROJECT** | **BLOCK A** | | **BLOCK B** | | **BLOCK C** | | Total Level Areas ($m^2$) | Total Building Areas ($m^2$) |
|---|---|---|---|---|---|---|---|---|
| | Level Area ($m^2$) | Building Area ($m^2$) | Level Area ($m^2$) | Building Area ($m^2$) | Level Area ($m^2$) | Building Area ($m^2$) | | |
| 1. PROJECT | 760 | 5320 | 590 | 1770 | 280 | 560 | 1630 | 7650 |
| 2. PROJECT | 400 | 1200 | 250 | 500 | 105 | 210 | 755 | 1910 |
| 3. PROJECT | 680 | 3400 | -- | -- | -- | -- | 680 | 3400 |
| Total | | | | | | | 3065 | 12960 |

A). All projects Start Simultaneously and have Equal Completion Times (Case 1).

The example being evaluated consists of three sub-projects. One of these projects has a longer completion period. The completion times of the other sub-projects are equalized with this time period, which leads to the abundances in these sub-projects to increase further. In sum, all of the projects will begin and end at the same time. The project objective function for the multiple sample project (case 1) is given below (Fig.10).

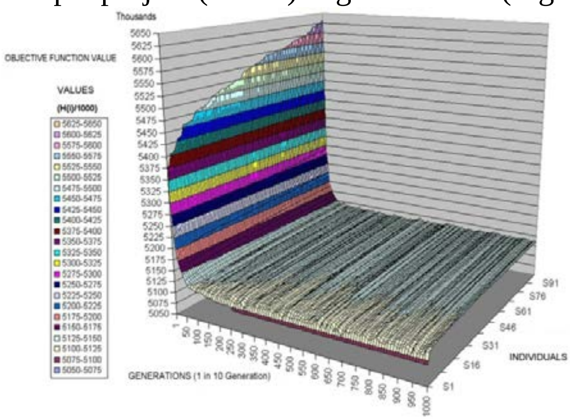



Figure 10. Project objective function for the multiple sample project (case 1) (Karakaya, 2007)

***Comparison of the Results of The Model (Case 1)***

The results obtained based on comparisons is provided in the table below (Table 7).

Table 7. Comparison of calculation models for multiple sample project (case 1). (Karakaya, 2007)

| NO | Name Of Account | Heuristic Model | Genetic Model | Difference (%) |
|---|---|---|---|---|
| A) | Resource Squares | 5110868 | 5093508 | -0.340 |
| B) | Available Amount | 42178 | 44286 | 4.998 |
| C) | Amount of Used | 37238 | 37238 | 0 |
| D) | Maximum | 183 | 173 | -5.464 |
| F) | Maximum | 54 | 84 | --- |
| E) | Mean Amount of | 150.636 | 158.164 | 4.997 |
| G) | Piling Ratio: D/B | 1.215 | 1.094 | -9.959 |
| H) | Percentage | 0.883 | 0.841 | -4.757 |
| I) | Project Duration | 280 | 280 | 0 |

Compared to the solution obtained with the heuristic model, the genetic model’s solution exhibited a:

- A 0.340% decrease in resource squares.
- A 4.998% increase in the available amount of resources.
- A 5.464% decrease in the maximum amount of resources, while the maximum resource days were 84.
- A 4.997% increase in the mean amount of resources.
- A 9.959% decrease in the piling ratio.
- A 4.757% decrease in resource utilization.

In sum; the total resource squares, the maximum amount of resources and the piling ratio decreased. The mean amount of resources increased. The available amount of resources increased. The percentage of resource utilization increased as well.

***B). All projects start simultaneously and all completion times are constant (Case 2).***

The example being evaluated consists of three sub-projects. The sub-projects commence at the same time, and their first critical targets include a project completion time (which is considered as a constant). In sum, all of the projects will begin at the same time, and the completion times of the projects will not change. In mixed heuristic models, there is no need to distinguish between single and multiple projects. For solving multiple projects with the genetic algorithm, a multiple-chromosome individual approach was employed. The project objective function for the multiple sample project (case 2) is given below (Fig.11).

**Comparison of the Results of The Model (Case 2)**

The results obtained based on comparisons is provided in the table below (Table 8).

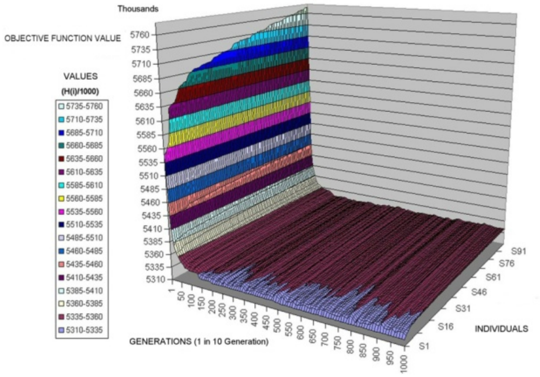



Figure 11. Project objective function for the multiple sample project (case 2) (Karakaya, 2007)

***Comparison of the Results of The Model (Case 2)***

The results obtained based on comparisons is provided in the table below (Table 8).

Table 8. Comparison of calculation models for multiple sample project (case 2). (Karakaya, 2007)

| NO | Name Of Account | Heuristic Model | Genetic Model | Difference (%) |
|---|---|---|---|---|
| A) | Resource Squares Total: | 5323826 | 5321272 | -0.048 |
| B) | Available Amount of | 40897 | 41940 | 2.550 |
| C) | Amount of Used Resources: | 37238 | 37238 | 0 |
| D) | Maximum Amount of | 208 | 201 | -3.365 |
| F) | Maximum Resource Days: | 84 | 84 | --- |
| E) | Mean Amount of Resources: | 146.061 | 149.786 | 2.550 |
| G) | Piling Ratio: D/B >= 1 | 1.424 | 1.342 | -5.758 |
| H) | Percentage Resource Use: C/B | 0.911 | 0.888 | -2.525 |
| I) | Project Duration (T): | 280 | 280 | 0 |

Compared to the solution obtained with the heuristic model, the genetic model’s solution exhibited a:

- A 0.048% decrease in resource squares.
- A 2.550% increase in the available amount of resources.
- A 3.365% decrease in the maximum amount of resources.
- A 2.550% increase in the mean amount of resources.
- A 5.525% decrease in the piling ratio.
- A 2.525% decrease in resource utilization.

In sum; the total resource squares, the maximum amount of resources and the piling ratio decreased. The mean amount of resources increased. The available amount of resources increased. The percentage of resource utilization decreased.

## Conclusion

In examples involving planning for single and multiple projects that utilize one type of resources, the proposed genetic algorithm mode and the mixed heuristic model were separately used for resource leveling purposes.

The tables summarizing the comparisons between these models are provided above. The percentages values provided in these tables indicate the percentage difference in the results obtained with the genetic algorithm model in comparison to the mixed heuristic model. Negative values indicate a decrease, while positive values indicate an increase. The general findings and results of these tables can be summarized as follows:

- Lower results were obtained with the genetic algorithm model in all calculations using the total resource squares.
- The genetic algorithm solution resulted in an increase in the amount of available resources.
- In both models, there was no difference with regards to the amount of resources being used.
- With the genetic algorithm model, a positive result was obtained in all examples/cases with regards to the maximum amount of resources. The maximum need for resources decreased.
- The genetic algorithm increased the mean amount of resources available and the efficiency of resource utilization.
- The genetic algorithms reduced the maximum need for resources, and increased the mean amount of resources. This, in turn, has considerably reduced the piling ratio in all calculations, bringing its value closer to 1 (its ideal limit).
- With the genetic algorithm, the percentage of resource utilization decreased. However, in all solutions, the maximum need for resources decreased considerably when using the genetic algorithm.
- In both models, the total project duration has not changed. In resource leveling, the total project duration is an invariable and constant. The applied models functioned correctly.

The following comments can be made regarding the proposed genetic algorithm model.

- Owing to its structure, the genetic algorithm's effectiveness in reaching a solution depends on the extent to which it can advance towards the optimum result with the current generation length. In an experiment with 1000 generations, almost all of he individuals approach the optimum solution after 100 generation (about 10% of the total experiment generations). Thus, the model functioned correctly as expected.

- Owing to its multiple chromosome structure, the proposed can level multiple projects.
- The two selected mutation operators allow the maximum amount of resources to decrease rapidly.
- Compared to the others, the type 2 (single-point, piece change) provided better results between the crossing operators.
- As the initial generation is formed randomly, the processes in the project are assigned to times and abundances that are very different from one another; they are thus not constrained by local optimum solutions.
- The listing of individuals within their generations according to their function values considerably affects the calculation period. However, there are cases where the selected listing directly affects the performance of the genetic algorithm (its approach to the solution), and where the calculation period is limited.
- In the algorithm assumptions, elitism and placing emphasis on the selection of good individuals (crossing rate of 25%) prevented significant problems in the determination of optimum solutions by the algorithm.

Based on the size of the problem; the researchers can develop genetic algorithms for planning and leveling studies according to different types of resource utilization and different limitation assumptions.

**References:**

Burgess, A.R., & Killebrew, J.B.. (1962). "*Variation in Activity Level on a Cyclic Arrow Diagram*". T h e Journal of Industrial Engineering, March-April, Vol.13,No.2, pp. 76-83

Dianati, Mehrdad, Song Insop, Treiber Mark. (2002). "*An Introduction to Genetic Algorithms and Evolution Strategies*", University of Waterloo, Ontario, N2L 3G1, Canada.

Emel, Gül Gökay ve Taşkın Çağatan. (2002). "*Genetic Algorithms and their Areas of Application*". *Journal of the Uludağ University Faculty of Economic and Administrative Sciences* Vol:XXI Issue:1 pp.129-152, Bursa.

Eyeci, Y. (1985). "*Workforce Planning for More Then One Construction*" Republic of Turkey Anadolu University Faculty of Science, Department of Civil Engineering, Postgraduate Thesis, pp:5-10, Eskişehir.

Fogel, L.J., Owens, A. J. Ve Walsh, M. J. (1966). "*Artificial Intelligence through Simulated Evolution*", Wiley, New York.

Goldberg, D. E. (1989). "*Genetic Algorithms is Search, Optimization and Machine Learning*". MA: Addison-Wisley, 412 p.

Holland, J.H.. (1975). "*Adaptation in Natural Artificial Systems*"., Ann Arbor, MI: The University of Michigan Pres, 2nd Edition, 199929, Boston, MA: MIT Pres.

İşçi, Öznur. (03.10.2002). "*The Application of Genetic Algorithms in Operational Studies*". Ege University, The Graduate School of Natural and Applied Sciences, Department of Statistics, Doctorate Thesis, Izmir.

Karakaya, A. (2007). "*Modeling the Use of Genetic Algorithms in the Management of Construction Projects for Resource Leveling Purposes*". Dokuz Eylül University, The Graduate School of Natural and Applied Sciences, Department of Architecture, Doctorate Thesis, Izmir.

Kocatürk, Nazan. (1987). "*Cost Scheduling in Construction Projects Using CPM*" A Master's Thesis in Civil Engineering Middle East Technical University, (Supervisor- Assoc.Prof. Orhan Yüksel, Supervisor-Assoc. Prof. Dr. Ayhan İnal) P:xiii+132, Ankara.

Li Heng, Love Peter. (1997). "Using Improved Genetic Algorithms to Facilitate Time-Cost Optimization". *Journal of Construction Engineering and Management*, (9), pp.233-237.

Özdemir, İ. (1988). "*Determining the Most Suitable Investment Period for Investments in which Payments are Periodic and Variable, while Workforce and Machine Power are Constant and Limited*". Anadolu University, The Graduate School of Natural and Applied Sciences, Department of Civil Engineering, Doctorate Thesis, Eskisehir.

Rechenberg, I. (1971): Evolutionsstrategie – Optimierung technischer Systeme nach Prinzipien der biologischen Evolution (PhD thesis). Reprinted by Fromman-Holzboog (1973).

Rennard, Jean-Philippe. (2000). "*Introduction to genetic algorithms: Genetic Algorithm Viewer : Demonstration of a Genetic Algorithm*". http://www.rennard.org/alife

Şen, Z. (2004). "*Genetic Algorithms and the Best Renewal Methods*". Su Foundation Publications, Istanbul.

Tulip, A. (1972). "*Resource Smoothing-Theory and Practice*", Proceeding of 3rd INTERNET72, Stockholm, Vol. 3, pp 355-373.

19. Yüksel, O. (2000). "*Three Problem and Three Algorithm in Project Management with Network Diagrams*". Sf:173, 2nd. Congress of Constraction Management , 15th. of June, Izmir.
p:173, Izmir.

Zbigniew, Michalewicz. (1996). "*Genetic Algorithms + Data Structures = Evolution Programs Third, Revised and Extended Edition*." Springer-Verlag Berlin Heidelberg. Newyork USA, 387