

Scheduling of Construction Projects under Resource-Constrained Conditions with a Specifically Developed Software using Genetic Algorithms

Mursel ERDAL*, Recep KANIT

Abstract: The purpose of this study is to develop a genetic algorithm (GA) based software that can perform resource allocation close to optimum and that can determine the critical path by minimizing the project duration according to the resource profile for a present work schedule and resource pool using a programmable objective function. In this context, the methodology of GAs was presented, the software was developed and the performance of this software was tested with a sample project. With the developed software, by minimizing the activity durations in both constrained and unconstrained resource conditions, projects can be scheduled, total duration and the critical path of the projects can be determined. With this software, any construction company will be able to determine how much time would be required to complete a project at the bidding stage by considering its resources and constraints and can take the required precautions. The main difference of this present study is that the developed code performs minimization of schedule duration integrated with resource allocation and levelling. It also determines the critical path of the final solutions. Both renewable and non-renewable resources are included in the code which is not often considered in the literature. By minimizing project duration and optimizing resource allocation, construction projects can become more sustainable, and the environmental impact of the construction process could be minimized.

Keywords: genetic algorithms; optimization; renewable resource; scheduling; sustainability

1 INTRODUCTION

Today it is impossible to envisage an environment with unlimited resources. A management system in which resources are used efficiently is needed in today's conditions where resources and time are limited. A project needs to be planned from start to end to assess time and money optimally and to provide optimum use of work power, equipment-machinery, etc. which are limited [1].

Researchers have been solving resource allocation problems of project planning under constrained resources using classical optimization techniques [2]. However today's project requirements made the content and size of the problems bigger and complicated, so classical solution techniques are insufficient, such that variables are so many, solution clusters are so wide, therefore differentiation of the functions becomes impossible [1]. In this point, solutions are tried to be obtained using an algorithm which is based on J. Holland's [3] study. Holland, who was lecturing about adaptation to nature, started to study GA theory which is based on Charles Darwin's natural theory of evolution [4]. According to natural evolution theory, weak species, which cannot adapt themselves to their habitat, will become extinct. Species which can adapt themselves to their environment can survive. Each generation will take the best properties of previous generations and they evolve by adapting themselves to the natural conditions which can change. The last generation is the one that obtained a better adaptation compared to past generations [5]. The main purpose of natural evolution is to supply necessary development for the living creatures to survive in their environment. All the properties of living creatures are coded in their chromosomes and the properties of the best species are transferred to the next generation. Weak species do not have much chance to survive and to transfer their properties to the next generation [6]. The principles of GA are parallel to the nature's basic principles. A population is generated from the solution cluster of problems. Individuals of this population are formed from the chromosomes which are coded according to a particular system. Individuals of this population are subjected to processes which are the

duplicates of natural breeding, and a new generation is formed. Individuals showing the best performance are kept and they are used for the generation of new individuals. The solution to the problem is approached at each new generation. The best individual of the final population is considered to be the solution to the problem. This solution may not be the exact solution but it is close to the ideal solution [7]. Today GA's are being used in many applications like the plane design to national security models. One of the applications of GA is the resource-constrained project scheduling problems. Many researchers studied on this subject [8-15]. Davis and Patterson [16] solved a project consisting of 2 dummy activities and a total of 27 activities with GA and they compared the solution with the results of other heuristic methods. They determined that the result of GA was superior from that determined from other heuristic methods. Wall [17] applied GA to 1000 number of projects. These projects have dimensions ranging from 10 to 300 activities and 3 to 10 constrained resources. GA was able to obtain the best solution for most of the problems. Sivrikaya-Serifoglu [18] used the GA approach for the minimization of project duration and maximization of net present value. The Result of GA for project duration minimization was compared with the result of local constrained based analysis (LCBA) and it was determined that the solution of GA was better than that of LCBA. Sivrikaya-Serifoglu et al. [19] widened their previous study to incorporate non-renewable resources. Mori & Tseng [20] examined the resource-constrained project scheduling problem with only renewable resources. In their study, an activity number was assigned to each gene of the chromosome, a mode for this activity, scheduling sequence, and start-finish time information coding system was utilized. Mori & Tseng [20] compared their approach with the stochastic approach that was suggested by Drexel and Grünwald [21] and they found that GA was better. Ozdamar [22] used a hybrid GA approach to the problem in which non-renewable resources are present. In this approach Ozdamar [22] defined a 2D chromosome structure with one dimension belonging to activity modes and the second one belonging to rule number defined from

group properties. In this study the best results were found to be obtained within an appropriate CPU time by using the suggested approach. Hegazy [23] solved a project planning problem having 20 activities and 6 resources by using three different software. For the case of the unlimited resource, the present duration was calculated as 32 days with CPM, whereas Microsoft Project and Primavera determined it as 49 days. GA solution was determined as 45 days. Guvenc [24] used the algorithm suggested by Goldberg and prepared software by using Delphi 4.0 programming language. Prepared software was tested on a sample project. Hartmann [25] suggested a GA with both renewable and non-renewable resources. In this study, he performed experimental studies and selected the best GA approaches. Furthermore, he compared the best GA approach with the three heuristic approaches that were reported in the literature. According to this comparison, GA was found the give better results according to the average deviation criteria of the best project time. Bettemir and Sonmez [26] proposed a hybrid methodology based on genetic algorithms (GA) and simulated annealing (SA) to solve resource-constrained project scheduling problem; they determined that the GA-SA hybrid methodology is superior to sole GA. The main difference of our study is that

our code performs minimization of schedule duration integrated with resource allocation and levelling. It also determines the critical path of the final solutions. Both renewable and non-renewable resources are included in the code which is not often considered in the literature.

2 APPLICATION OF GAS TO CONSTRUCTION PROJECTS

2.1 Problem Description an Introduction to Structured Software

In this study, the proposed GA model that is enabled to make resource allocation near optimum value by using an objective function for the current schedule and resource pool was developed in addition to this; it minimizes project duration while determining the critical path; moreover the model can easily be applied to real-life problems with its user-friendly interface. The distinguishing feature of this model is its user-friendly non-rigid structure. That is why the proposed model is applicable for all types of projects so the number of activities, resource quantity, and duration of project are not important. The software interface is shown in Fig. 1.

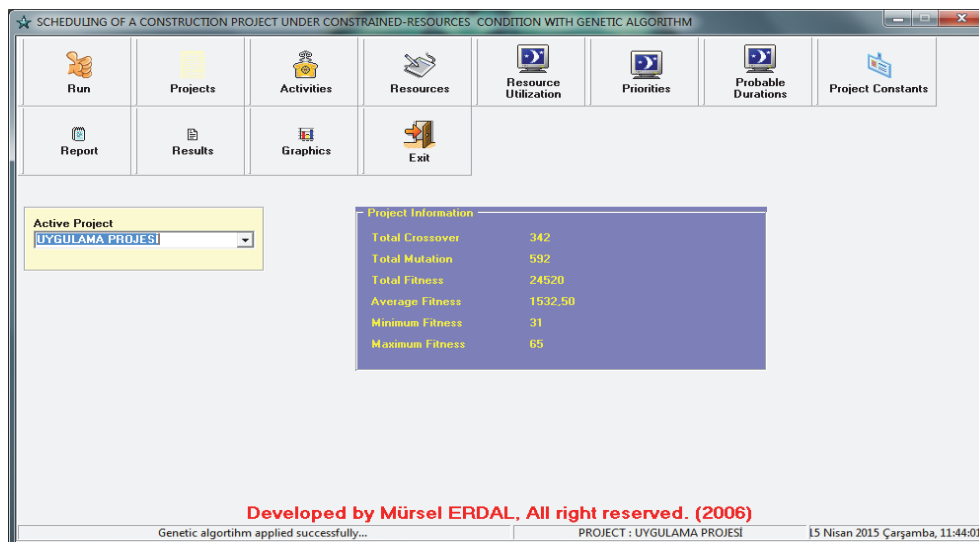


Figure 1 Software interface

Software was developed with Delphi 7.0 and Paradox data tables by using a computer with 2.60 GHz CPU, 4 GB RAM, and Windows 7 operating system. The software has two main parts. Activities, dependencies between activities, resources, and activities resource usage quantities are defined in the first part of the project. Solutions and graphs are shown in the second part after the execution of software. Solutions could be taken as reports and automatically be saved, moreover, solutions can be transferred to excel. Firstly, project information has to be defined for minimization of project duration and calculation of critical path with GA. "Project's" button should be clicked for entering the required information after that "Project Definition" screen appears for the definition of the project. After the project definition step from the "Active Project" selection box, the defined project has to be selected among all the menus that allow data input to become active. "Activity Definition" screen appears

after pressing the "Activity" button. "Activity Definition" option allows input of activity names and activities' actual finish dates. Resources and daily usage quantities are defined via the "Resource Definition" screen. Besides, the software provides an opportunity to decrease activity duration with available unused resources.

After defining the resources, the "Precedence" button clicked, and the "Precedence Definitions" menu opened. This menu includes the definitions about which activity precedes or follows each other. Following the definitions of the priority relations, rather than the ordinary completion periods of the activities, probable completion periods where modes were created should be entered by clicking the "Probable Time Periods" button. After this entry is completed, all of the project data will be entered. Afterward, the screen shows details of the definitions of GA parameters opened by clicking the "Project Constants" box (Fig. 2).

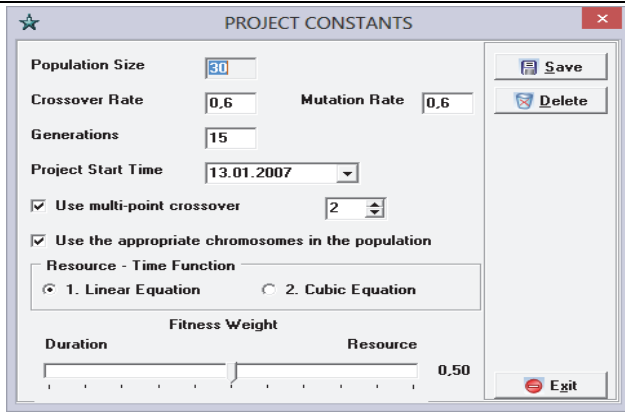


Figure 2 Project invariants

This screen provides the size of the population, crossover rate, mutation rate, number of generations, starting time of the project, number of the crossover points, information regarding whether the most suitable chromosomes will be used in the new population, and the

resource-time relationship. After defining all of these processes, the program runs by clicking the "Run" button.

After running the program, necessary calculations for the minimization of the project time, minimization of the deviations from resource profile, and determination of the critical path according to the new project data will be performed in a few seconds (depending on the size of the project). The information regarding GA will be displayed under the "Project Information" section on the "Home Screen". In the "Project Information" section, the total number of cross-overs and mutations, compatibility, mean compatibility, and the minimum compatibility are displayed. The minimum compatibility value displayed is the total time period of the project. When the program runs, the other buttons on the "Home Screen" where the results are displayed will be also activated. The "Results" screen, where all of the results required by the users are displayed, opens by clicking the "Results" button on "Home Screen" (Fig. 3).

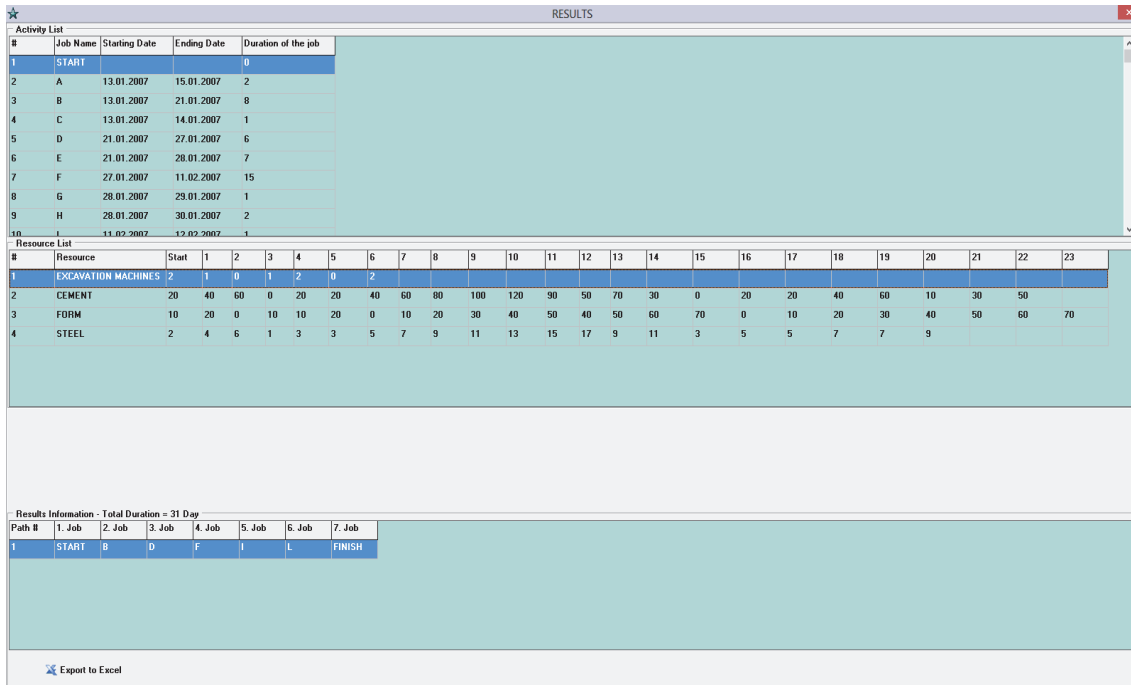


Figure 3 Result screen

The "Results" screen consists of three sections; Jobs, Resources, and Result Information. In the "Jobs" section, Job Name, Starting Date, Ending Date, and Duration of the Job is listed. The "Resources" section shows the status of the resources which will be used during the job. In the "Result Information" section, the total time period, and the critical path of the project are displayed. All of the information on this screen can be exported to Excel by clicking the "Export to Excel" button. Along with User, Report, and Result screens, graphics of the information resulting from the performance of other programs can be viewed by clicking the "Graphics" button on "Home Screen". There are two different graphics on this screen. "Generation-mean compatibility" graphic (Fig. 4a) is based on the compatibility value of each generation. "Generation-optimum compatibility" graphic (Fig. 4b) shows the course of minimization of the time period of the project through

the generation. For the graphics, the information about the related activities, priority relations, estimated ending date of each activity, resource requirements based on the ordinary time period, and the probable ending dates depending on the resource restrictions which were entered into the program, are stored in the program with dynamic strings. Thus, chromosomes were created by simply changing the activities' ending time, which is randomly chosen.

The implementation of the GA model to construction projects is explained in the following section. For the presentation of GA stages and the solution method, the project schedule given in Fig. 5 is utilized. This project consists of 6 activities including start and finish activities. In the schedule, names of activities, expected, optimistic, and pessimistic durations are presented [1].

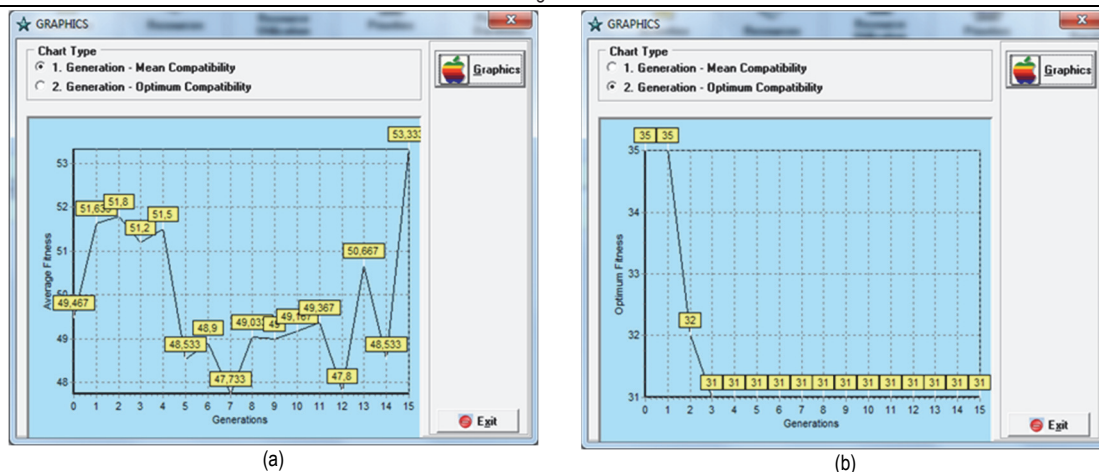


Figure 4 Result screen: (a) Generation-mean compatibility graphic; (b) Generation-optimum compatibility graphic

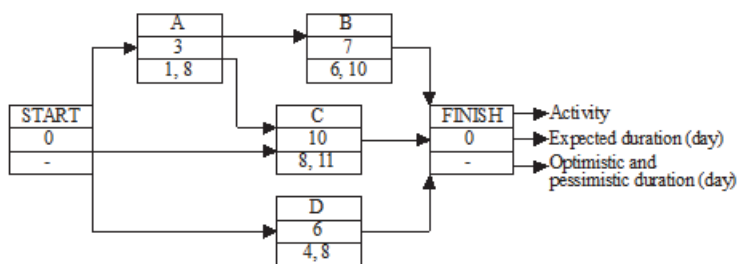


Figure 5 Project network of sample project

2.2 Assumptions

To apply GA to construction projects some assumptions have to be made. These are:

1. The project consists of several activities, which may have more than one mode of execution.
2. Activities may require multiple resources.
3. Activity resource needs are defined according to their expected durations and a linear relation is used between resource and time to calculate the resource needs of modes (Fig. 6). The linear relationship presented in Fig. 6 is a simplification of a real-life case. A more complex function may be utilized for time and resource consumption relationships. If the availability of a renewable resource is more than one, computer code first assigns the resource to an activity that has an early start or shorter duration, then the resource is shared among the other activities.

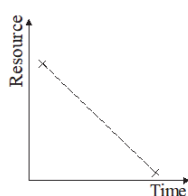


Figure 6 Assumption of linear relationship between time and resource consumption

4. The start time of each activity is dependent on the completion of some other activities ("finish-to-start" precedence relations).
5. Activities cannot be interrupted.
6. Resources can be renewable and non-renewable.

2.3 Objective of Method

The proposed method has multi-objective performance measures. The method provides both total project duration minimization and minimization of deviations of available resource profiles according to constrained resources or unconstrained resources cases. In addition to the determination of total project duration, software and the proposed methodology can also determine the critical path. To accomplish this goal the required mathematical notation of objective function is given in Eq. (1).

$$Min \left[((\gamma) \times t_n) + \left((1-\gamma) \times \left(\sum_{t_i \in A_{ij}} b_k - r_{ik} \right) \right) \right]; \quad (1)$$

$$k=1, 2, \dots, m; i=1, 2, \dots, n$$

The definitions of the symbols of the equation are shown below.

- t_n - The finish time of the last activity (total duration),
- r_{ik} - The amount of resource k required by activity i at time t_i ,
- b_k - The total availability of resource k at time t_i ,
- A_{ij} - The set of activities in progress at time t_i ,
- m - The number of different resource types,
- n - The number of activities,
- γ - The weight factor $1 \leq \gamma \leq 0$.

2.4 Algorithm of Method

For the solution of the problem, GA suggested by Goldberg [4] was used. The flowchart of the algorithm of the operation is presented in Fig. 7.

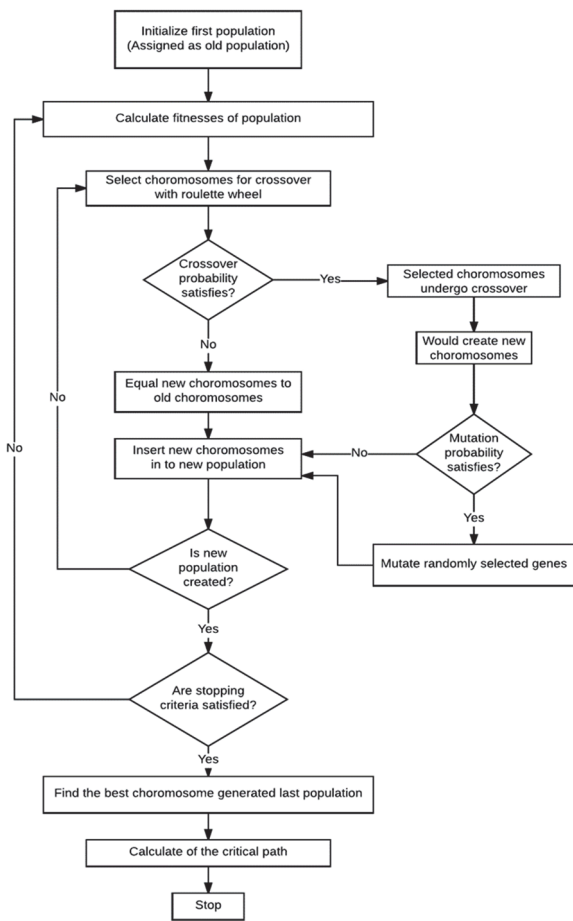
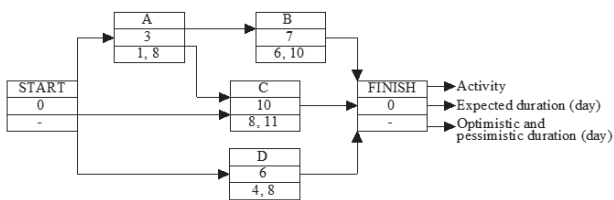


Figure 7 Flowchart of algorithm

2.5 Generation of Chromosomes

In the project planning problems, the work program itself presents a solution; for this reason for the generation of chromosomes permutation coding was used. The technique of population coding is utilized in the software; however, directions of activities are coded instead of the order of activities. The presentation suggested by Mori & Tseng [20] was given in Fig. 8 and this approach was used through the study.



Some Possible Solutions (Chromosomes)

START	A	B	C	D	FINISH
0	3	6	11	6	0
0	8	10	10	4	0
0	1	7	8	4	0

Figure 8 Generation of chromosomes in multi-mode project programming

The probable ending dates of the job (number of the modes) are entered as limitless in the program. The user can insert as many ending dates as she/he wants. Since any ending time chosen for an activity differs from the ordinary

time, the program performs the necessary calculations by adapting the resources according to the expected time.

2.6 Calculation of Fitness Values

The fitness values of individuals were calculated from the total project duration and deviations of required resources from the available resources.

Total duration fitness:

For the calculation of the duration fitness of an individual Eq. (2) was used.

$$f_{dur} = \frac{t_{max} - t_n}{t_{max} - t_{min}} \tag{2}$$

where:

- f_{dur} - The duration fitness,
- t_{min} - The probable minimum project duration,
- t_{max} - The probable maximum project duration,
- t_n - The finish time of the last activity (total duration)

Resource fitness:

For the calculation of the resource fitness of an individual Eq. (3) was used. Histogram showing the parameters in Eq. (4) was presented in Fig. 9.

$$f_{res} = \frac{A_2 - A_1}{A_2} \tag{3}$$

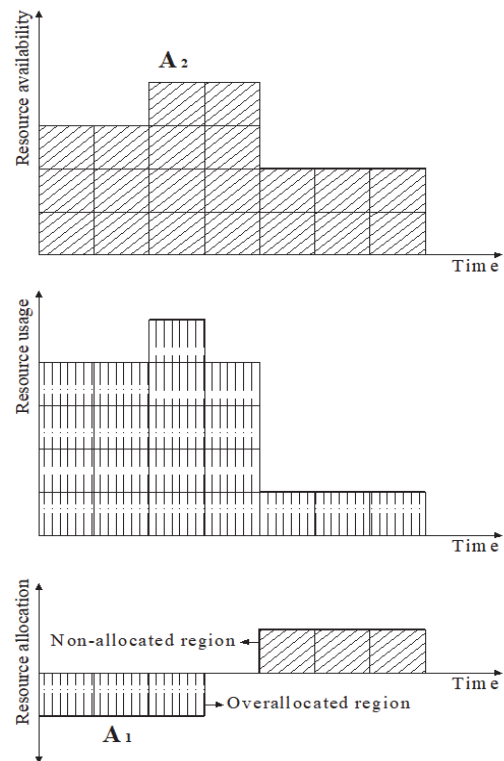


Figure 9 Flowchart of algorithm

For "m" number of resources, equation can be arranged as follows:

$$f_{res} = \frac{\sum_{i=1}^m (A_2 - A_1)}{\sum_{i=1}^m A_{i2}} ; i(1, m) \tag{4}$$

where:

A_1 - The exceeding amount in the histogram,

A_2 - The area of the available amount,

f_{res} - The resource fitness.

By combing total duration and resource fitness, resultant fitness function can be described as shown in Eq. (5).

$$f = \gamma \times f_{res} + (1 - \gamma) \times f_{dur} \quad (5)$$

2.7 Selection of New Chromosomes

For the generation of new chromosomes roulette wheel selection technique which was suggested by Goldberg [4] was used. The problem of illegal solutions is overcome by using genetic operator algorithms defined by Mori and Tseng [20]. These operators guarantee that precedence constraint specified in the scheduling problem remains satisfied.

2.8 Crossover Operation

Two new individuals are generated using crossover operation in which particular parts of two randomly selected individuals from the population are cross changed [27-29]. Implemented computer code both one cut point and multi-cut point (up to 4) crossover can be used. The software package was designed to allow the selection of crossover method and crossover ratio by the user. A simple one-point crossover operation is shown in Fig. 10. This is point crossover operation used for the solution of the sample problem.

2.9 Mutation Operation

In GAs, some genetic information may be lost early in limited populations. Genes that form the chromosome may all be the same. Such a chromosome cannot be changed by the crossover operation. In such cases, randomly selected genes are changed by outside interference [30]. The software package was designed to allow the user can change the mutation ratio. A simple presentation of mutation operation is shown in Fig. 11.

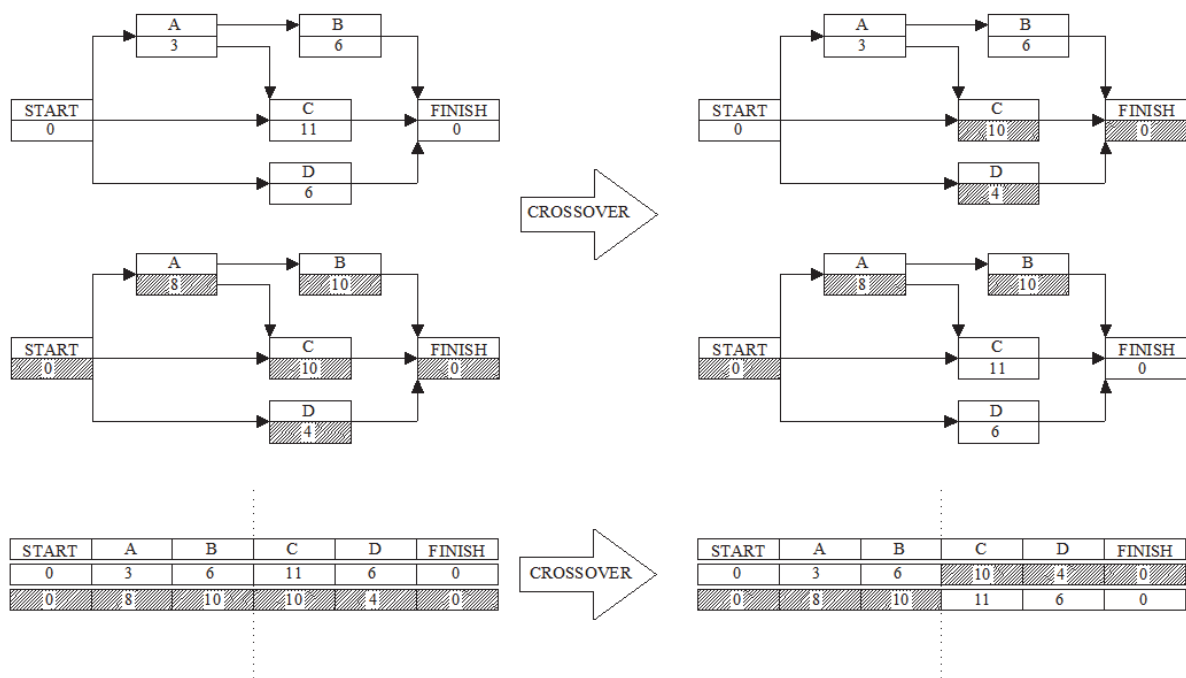


Figure 10 Simple one-point crossover operation

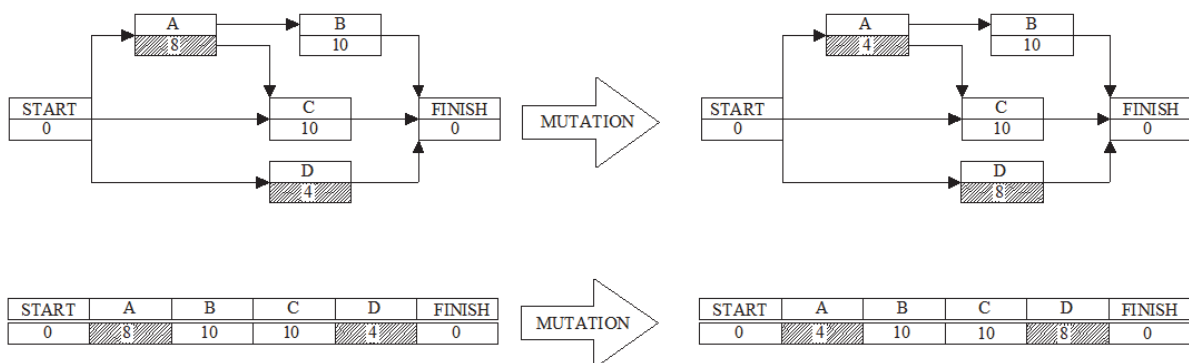


Figure 11. Mutation operation in project programming

3 SAMPLE PROBLEM

To test the software package a project schedule with 14 activities was used (Fig. 12).

Project details are listed in Tab. 1 and available resources are listed in Tab. 2. According to the critical path method solution under unlimited resources assumption, the total duration of the project is calculated as 22 days.

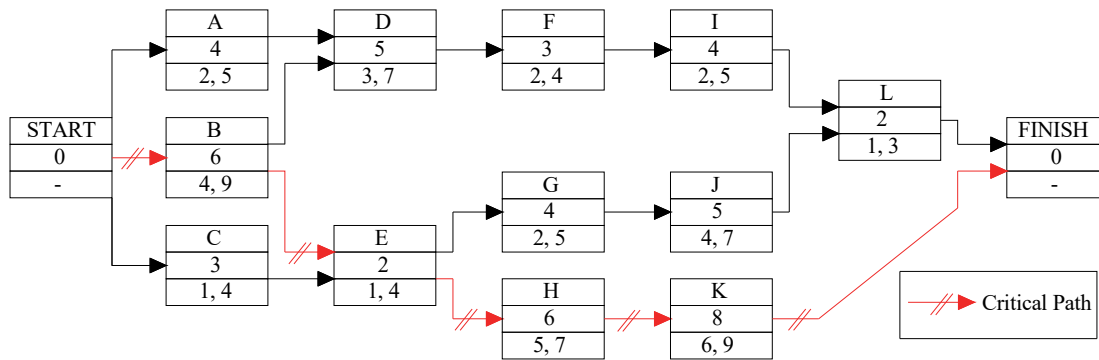


Figure 12 Scheduling of test problem

Table 1 Project information

No	Name of job	Expected duration of job / day	Pessimistic and optimistic duration / day	Predecessor	Resources	Amount of resources needed
1	START*	0	-	-	-	-
2	A	4	2, 5	START	Excavation Machine	1 number
3	B*	6	4, 9	START	Excavation Machine	2 number
4	C	3	1, 4	START	Excavation Machine	1 number
5	D	5	3, 7	A, B	Formwork	20 m ²
6	E*	2	1, 4	B, C	Formwork	70 m ²
7	F	3	2, 4	D	Steel, Cement	5 t, 60 bag
8	G	4	2, 5	E	Cement	30 bag
9	H*	6	5, 7	E	Cement	40 bag
10	I	4	2, 5	F	Formwork	10 m ²
11	J	5	4, 7	G	Cement	30 bag
12	K*	8	6, 9	H	Steel	8 t
13	L	2	1, 3	I, J	Cement	50 bag
14	FINISH*	0	-	K, L	-	-
Total project duration						
22 day						
Critical path						
START - B - E - H - K - FINISH						

Table 2 Amount of available resources

No	Resource name	Resource amounts	Resource type
1	Excavation Machine	2 number	Renewable
2	Cement	20 bag/day	Non-renewable
3	Steel	2 t/day	Non-renewable
4	Formwork	10 m ² /day	Non-renewable

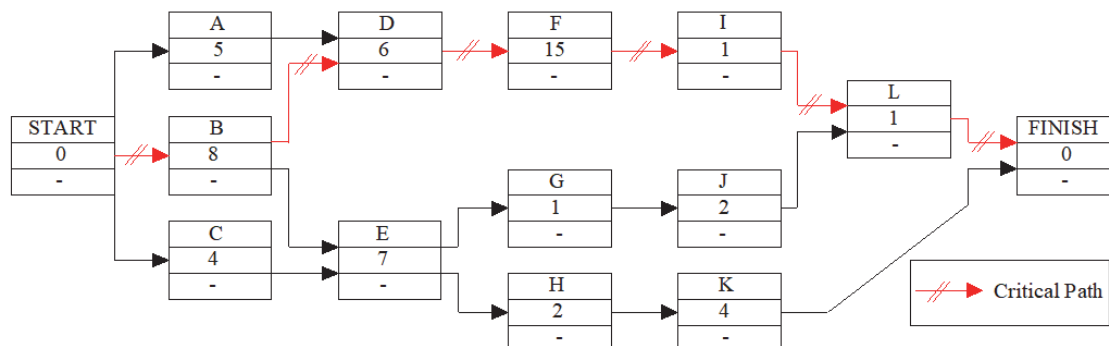


Figure 13 Scheduling of new project after the GA solution

By using resource constraints and information given in Tab. 1 and Tab. 2, activity durations and critical path were calculated using GA and the results are listed in Tab. 3.

According to the GA solution total project duration was calculated as 31 days. In addition, the critical path was also changed (Fig. 13). Tab. 3 lists the duration of each

activity and critical path activities. During calculation, the same weight (0.5) was assigned to research and time variables.

Table 3 Amount of available resources

No	Name of Job	Duration of the job / day
1	START*	0
2	A	5
3	B*	8
4	C	4
5	D*	6
6	E	7
7	F*	15
8	G	1
9	H	2
10	I*	1
11	J	2
12	K	4
13	L*	1
14	FINISH*	0
Total project duration		
		31 day
Critical path		
		START - B - D - F - I - L - FINISH

4 CONCLUSIONS

In this study, a software package that can perform resource distribution using a programmable objective function for a work program was developed. This software can also determine the critical path by minimizing total project duration using GA. When the availability of resources is limited, the concept of critical path loses its original meaning. Some activities may be delayed due to insufficient resources so that the critical path ceases to exist. However, "the critical path" term is used as the path having the largest duration. To test the developed software a project schedule consisting of 14 activities was used. According to the critical path method under unlimited resource assumptions, the total project duration was calculated as 22 days. When resources were constrained total project duration was calculated as 31 days using GA and the critical path of the project was also changed.

As a result, with the developed software any construction company will be able to determine how much time will be required to complete a project at the bidding stage by considering its own resources and constraints and can take necessary precautions. In this regard, it is thought that GAs can provide effective solutions to construction projects for better and more efficient construction plans. In addition, the minimization of the construction process could help minimization of the negative environmental impact of construction.

5 REFERENCES

- [1] Erdal, M. (2007). Scheduling of a construction project under constrained-resources condition with genetic algorithm. *Ph. D. Thesis, Gazi University Institute of Science and Technology*, Ankara, (in Turkish).
- [2] Haupt, R. L. & Haupt, S. E. (1998). *Practical Genetic Algorithms*, New York: John Wiley & Sons Inc.
- [3] Holland, J. H. (1975). *Adaptation in natural and artificial systems*, Ann Arbor: The University of Michigan Press.
- [4] Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*, New York: Addison-Wesley Publishing Company Inc.
- [5] Gen, M. & Cheng, R. (1997). *Genetic Algorithms and Engineering Design*, New York: John Wiley & Sons Inc. <https://doi.org/10.1002/9780470172254>
- [6] Gen, M. & Cheng, R. (2000). *Genetic Algorithms and Engineering Optimization*, New York: John Wiley & Sons Inc. <https://doi.org/10.1002/9780470172261>
- [7] Man, K. F., Tang, K. S., & Kwong, S. (1999). *Genetic Algorithms*, Great Britain: Springer-Verlag London Limited.
- [8] Chaudhry, I. A. & Usman, M. (2017). Integrated process planning and scheduling using genetic algorithms. *Technical Gazette-Tehnički vjesnik*, 24(5), 1401-1409. <https://doi.org/10.17559/TV-20151121212910>
- [9] Zheng, X. L. & Wang, L. (2015). A multi-agent optimization algorithm for resource constrained project scheduling problem. *Expert Systems with Applications*, 42(15-16), 6039-6044. <https://doi.org/10.1016/j.eswa.2015.04.009>
- [10] Izadkhah, H. (2019). Learning based genetic algorithm for task graph scheduling. *Applied Computational Intelligence and Soft Computing*, Article ID: 6543957. <https://doi.org/10.1155/2019/6543957>
- [11] Janes, G., Perinic, M., & Jurkovic, Z. (2017). Applying improved genetic algorithm for solving job shop scheduling problems. *Technical Gazette-Tehnički vjesnik*, 24(4), 1243-1247. <https://doi.org/10.17559/TV-20150527133957>
- [12] Tabataba'i Nasab, S. M. & Kaveh, M. (2016). A parallel simulated annealing (PSA) for solving project scheduling problem with discounted cash flow policy in pricing strategy of the project suppliers. *Technical Gazette-Tehnički vjesnik*, 23(6), 1555-1563. <https://doi.org/10.17559/TV-20140425121514>
- [13] Montoya-Torres, J. R., Gutierrez-Franco, E., & Pirachican-Mayorga, C. (2010). Project scheduling with limited resources using a genetic algorithm. *International Journal of Project Management*, 28(6), 619-628. <https://doi.org/10.1016/j.ijproman.2009.10.003>
- [14] Yassine, A. A., Mostafa, O., & Browning, T. R. (2017). Scheduling multiple, resource-constrained iterative, product development projects with genetic algorithm. *Computers & Industrial Engineering*, 107, 39-56. <https://doi.org/10.1016/j.cie.2017.03.001>
- [15] Perez, E., Posada, M., & Lorenzana, A. (2016). Taking advantage of solving the resource constrained multi-project scheduling problems using multi-modal genetic algorithms. *Soft Computing*, 20, 1879-1896. <https://doi.org/10.1007/s00500-015-1610-z>
- [16] Davis, E. W. & Patterson, J. H. (1975). A comparison of heuristic and optimum solution in resource-constrained project scheduling. *Management Science*, 21(8), 944-955. <https://doi.org/10.1287/mnsc.21.8.944>
- [17] Wall, M. B. (1996). *A genetic algorithm for resource-constrained scheduling*. Ph. D. Thesis, Mechanical Engineering Massachusetts Institute of Technology, USA.
- [18] Sivrikaya-Serifoglu, F. (1997). A new uniform order-based crossover operator for genetic algorithm applications to multi-component combinatorial optimization problems, *Ph. D. Thesis, Boğaziçi University, Institute of Science and Technology*, İstanbul.
- [19] Sivrikaya-Serifoglu, F., Ulusoy, G., & Sahin, S. (2000). A GA approach for the maximization of reduced cash flow in resource constrained project scheduling. *Industrial Engineering Journal*, 11(2), 2-12, (in Turkish).
- [20] Mori, M. & Tseng, C. C. (1997). A genetic algorithm for multi-mode resource constrained project scheduling problem. *European Journal of Operational Research*, 100(1), 134-141. [https://doi.org/10.1016/S0377-2217\(96\)00180-4](https://doi.org/10.1016/S0377-2217(96)00180-4)

- [21] Drexler, A. & Grünewald, J. (1993) Non preemptive multi-mode resource-constrained project scheduling. *IIE Transactions*, 25(5), 74-81.
<https://doi.org/10.1080/07408179308964317>
- [22] Ozdamar, L. (1999). A genetic algorithm approach to a general category project scheduling problem. *IEEE Transactions on Systems Man and Cybernetics Part C-Applications and Reviews*, 29(1), 44-59.
<https://doi.org/10.1109/5326.740669>
- [23] Hegazy, T. (1999). Optimization of resource allocation and levelling using genetic algorithms. *Journal of Construction Engineering and Management*, 125(3), 167-175.
[https://doi.org/10.1061/\(ASCE\)0733-9364\(1999\)125:3\(167\)](https://doi.org/10.1061/(ASCE)0733-9364(1999)125:3(167))
- [24] Guvenc, B. (2000). *A genetic algorithm for multi-mode resource constrained project scheduling*. M. Sc. Thesis, METU, Department of Civil Engineering, Ankara.
- [25] Hartmann, S. (2001). Project scheduling with multiple modes: A genetic algorithm. *Annals of Operations Research*, 102, 111-135. <https://doi.org/10.1023/A:1010902015091>
- [26] Bettemir, O. H. & Sonmez, R. (2015). Hybrid genetic algorithm with simulated annealing for resource-constrained project scheduling. *Journal of Management in Engineering*, 31(5), 04014082.
[https://doi.org/10.1061/\(ASCE\)ME.1943-5479.0000323](https://doi.org/10.1061/(ASCE)ME.1943-5479.0000323)
- [27] Chang, C. K., Christensen, M. J., & Zhang, T. (2001). Genetic algorithms for project management. *Annals of Software Engineering*, 11, 107-139.
<https://doi.org/10.1023/A:1012543203763>
- [28] Kim, K.W., Gen, M., & Yamazaki, G. (2003). Hybrid genetic algorithm with fuzzy logic for resource-constrained project scheduling. *Applied Soft Computing*, 2(3), 174-188.
[https://doi.org/10.1016/S1568-4946\(02\)00065-0](https://doi.org/10.1016/S1568-4946(02)00065-0)
- [29] Suresh, M., Dutta, P., & Jain, K. (2011). Analysis of an EPC project: a solution to the resource constrained project scheduling problem using genetic algorithms. *International Journal of Industrial and System Engineering*, 8(2), 251-269. <https://doi.org/10.1504/IJISE.2011.041372>
- [30] Alcaraz, J., Maroto, C., & Ruiz, R. (2003). Solving the multi-mode resource-constrained project scheduling problem with genetic algorithms. *Journal of the Operational Research Society*, 54(6), 614-626.
<https://doi.org/10.1057/palgrave.jors.2601563>

Contact information:

Mursel ERDAL

(Corresponding author)
Gazi University,
Faculty of Technology,
Department of Civil Engineering,
06500, Ankara, Turkey
E-mail: merdal@gazi.edu.tr

Recep KANIT

Gazi University,
Faculty of Technology,
Department of Civil Engineering,
06500, Ankara, Turkey
E-mail: rkanit@gazi.edu.tr