Multiobjective Reservoir Optimization Using Genetic Algorithm

Seyma Tiryakioglu
University of Mississippi

Follow this and additional works at: https://egrove.olemiss.edu/etd

Part of the Engineering Commons

Recommended Citation
https://egrove.olemiss.edu/etd/435

This Thesis is brought to you for free and open access by the Graduate School at eGrove. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of eGrove. For more information, please contact egrove@olemiss.edu.
MULTIOBJECTIVE RESERVOIR OPTIMIZATION USING GENETIC ALGORITHM

A Thesis
presented in partial fulfillment of requirements
for the degree of Master of Science
National Center for Computational Hydroscience and Engineering
The University Of Mississippi

by

SEYMA SEHRIBAN TIRYAKIOGLU

May 2018
ABSTRACT

This thesis investigates the optimization of energy generation in a multipurpose reservoir. The objective functions and the constraints for the multiobjective optimization of reservoir operation for maximization of energy production involve the solution of a set of nonlinear equations governing pressure flow in penstock and turbine system and the hydropower generation. The hydropower generation also requires operational rules that define how the reservoir storage volume must be used at different storage levels. For these reasons, the present thesis uses genetic algorithm (GA) functions available in Matlab to perform the multiobjective optimization of hydropower energy production. Multiobjective optimization is based on two objective functions: maximization of total energy production over a specified number of years for which observed data is available, and maximization of annual firm energy production for individual years.

Two separate Matlab codes using different hydropower generation algorithms were written for multiobjective optimization using GA. One of the two Matlab codes disregards the operation of individual turbines when allocating the storage volumes for firm energy and secondary energy and uses a nominal head loss due to the friction. The second Matlab code allocates the storage volumes for firm energy and secondary energy by considering hydropower production by individual turbines and the true head losses for each turbine.

In addition to these two Matlab codes for multiobjective GA optimization of the reservoir operation, a third Matlab code was also written to calculate the energy production using a traditional rule-based method.
These three Matlab codes were applied to calculate the hydroelectric power generation in a multipurpose reservoir. The reservoir operation strategies determined using GA, with and without the consideration of the operation of individual turbines, were then compared with the those obtained using the rule-based traditional method and the results of the original worksheet analysis provided by the State Water Works (DSI) of Turkey. Using prescribed operational policy and rule-set, this study shows that compared to traditional and DSI results, operations conducted using genetic algorithms produce both higher firm energy and a greater total energy production. Further, these results are found to be accurate over a period of 30 years.
## LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GA</td>
<td>Genetic Algorithm</td>
</tr>
<tr>
<td>ICOLD</td>
<td>International Congress on Large Dams</td>
</tr>
<tr>
<td>DSI</td>
<td>Devlet Su Isleri (State Water Works) of Turkey</td>
</tr>
<tr>
<td>GAT</td>
<td>Genetic Algorithm Toolbox</td>
</tr>
</tbody>
</table>
ACKNOWLEDGMENTS

I would like to express my deepest appreciation to my advisor, Dr. Mustafa S. Altinakar and my committee members, Drs. Bahram Alidaee, Yafei Jia, and Vijay Ramalingam. In addition, I thank Nazmi Kagnicioglu, Veysel Yildiz and Murat Yakut who have helped me and cooperated with me during my project work.

Finally, I am extremely thankful to my friends Luc Rebillout, Nazim Sahin and my family for their generous support.
TABLE OF CONTENTS

ABSTRACT ............................................................................................................................... ii

LIST OF ABBREVIATIONS ..................................................................................................... iv

ACKNOWLEDGMENTS ............................................................................................................. v

LIST OF TABLES ..................................................................................................................... viii

LIST OF FIGURES ................................................................................................................ vix

1.  INTRODUCTION ................................................................................................................. 1

   1.1.  Statement of the Problem ............................................................................................. 4

   1.2.  Test Case for Evaluating Developed Methods .............................................................. 7

2.  OPTIMIZATION OF RESERVOIR OPERATION .................................................................. 10

   2.1.  General Information ....................................................................................................... 10

   2.2.  Nonlinear Optimization and Genetic Algorithm Technique ........................................... 11

   2.3.  Genetic Algorithm ......................................................................................................... 13

      2.3.1.  Selection (Reproduction) ........................................................................................... 14

      2.3.2.  Crossover .................................................................................................................. 15

      2.3.3.  Mutation ................................................................................................................... 17

   2.4.  Literature Survey on the Use of Genetic Algorithms for Optimization of Reservoir

         Operations ......................................................................................................................... 18
3. GENETIC ALGORITHM OPTIMIZATION OF RESERVOIR OPERATIONS USING MATLAB

3.1. Genetic Algorithm Programming Using Matlab .......................................................... 21
3.2. Formulation of the Multiobjective Optimization Problem in Matlab ....................... 25
3.3. Description of the Matlab Code: A Study Case ......................................................... 30
3.4. The Concept of Energy Generation at a Dam and the Formulation of Hydropower Generation .................................................................................................................. 34
3.5. Final Considerations for the Matlab Code with Genetic Algorithm ......................... 39

4. RESERVOIR OPERATION USING RULE-BASED TRADITIONAL METHOD (CODE03) ........................................................................................................................................... 43

5. CASE STUDY: KAYRAKTEPE DAM IN TURKEY ........................................................ 48

6. DISCUSSION OF RESULTS AND CONCLUSIONS ......................................................... 57
6.1. Comparison of Results of CODE01 with Operational Data from the DSI ............... 59
6.2. Comparison of Results of CODE02 with CODE03 .................................................... 65
6.3. Sensitivity of the Optimal Results to the Parameters of Genetic Algorithm .......... 69
6.4. Conclusion ..................................................................................................................... 70

7. FUTURE WORK ................................................................................................................ 76

REFERENCES ..................................................................................................................... 78

VITA ..................................................................................................................................... 83
LIST OF TABLES

Table 1  List of input variables for the Matlab genetic algorithm code............................................. 30

Table 2  Characteristics of the Kayraktepe Dam Project................................................................. 50
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Storage allocation zones of a reservoir</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Illustration of reservoir operation</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>Concept of solution space</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>Illustration of global and local maximum points in a nonlinear problem.</td>
<td>12</td>
</tr>
<tr>
<td>5</td>
<td>Flowchart of genetic algorithm</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>Tournament method for “Selection” step</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>Illustrations one point, two points, and uniform crossover methods</td>
<td>17</td>
</tr>
<tr>
<td>8</td>
<td>The interface of Genetic Algorithm Toolbox (GAT) in Matlab</td>
<td>22</td>
</tr>
<tr>
<td>9</td>
<td>Plot options for the genetic algorithm in Matlab</td>
<td>24</td>
</tr>
<tr>
<td>10</td>
<td>Flowchart of the genetic-algorithm Matlab code for the optimization of energy generation</td>
<td>33</td>
</tr>
<tr>
<td>11</td>
<td>Hydraulics of energy generation at a dam</td>
<td>34</td>
</tr>
<tr>
<td>12</td>
<td>Friction coefficient graphs for Haaland (1983) and Altshul (1952)</td>
<td>38</td>
</tr>
<tr>
<td>13</td>
<td>Illustration of reservoir zones for the rule-based traditional method</td>
<td>43</td>
</tr>
<tr>
<td>14</td>
<td>Illustration of storage volume allocation for the case “a.1”</td>
<td>45</td>
</tr>
<tr>
<td>15</td>
<td>Illustration of storage volume allocation for the case “a.2”</td>
<td>45</td>
</tr>
<tr>
<td>16</td>
<td>Illustration of storage volume allocation for the case “a.3”</td>
<td>46</td>
</tr>
<tr>
<td>17</td>
<td>Illustration of storage volume allocation for the case “b.1”</td>
<td>47</td>
</tr>
</tbody>
</table>
Figure 18 Location of dams and their operating status: Red: under construction (Bagbasi, Bozkir, Ermenek), Yellow: in the project development phase (Mut, Kayraktepe), Green: already operating dam (Gezende).

Figure 19 Monthly irrigation volume needs of the agricultural sector.

Figure 20 The stream gaging and meteorological gaging stations.

Figure 21 Monthly precipitation rates.

Figure 22 Monthly average temperatures.

Figure 23 Monthly evaporation volumes.

Figure 24 Time series of monthly inflow volume into the Kayraktepe Reservoir.

Figure 25 Time series of monthly inflow (I) and potential energy (PEV) volumes.

Figure 26 Duration curve for monthly potential energy volume (PEV).

Figure 27 Elevation-volume and elevation area curves for Kayraktepe Reservoir.

Figure 28 Modeling of the Kayraktepe Project.

Figure 29 Comparison of end of the month storage volumes obtained from optimization using CODE01 with the results provided by the DSI.

Figure 30 Comparison of end of the monthly energy production obtained from optimization using CODE01 with the results provided by the DSI.

Figure 31 Pareto front obtained using Matlab for CODE01.

Figure 32 Comparison of the annual total energy production obtained from optimization using CODE01 with the results provided by the DSI.

Figure 33 Comparison of the annual firm energy production obtained from optimization using CODE01 with the results provided by the DSI.
Figure 34  Comparison of end of the month storage volumes obtained with CODE02 (optimization by considering individual turbines) and CODE03 (rule-based traditional method). .................................................................................................................................................................................................................................................. 66

Figure 35  Comparison of monthly energy production obtained with CODE02 (optimization by considering individual turbines) and CODE03 (rule-based traditional method). ............................................. 66

Figure 36  Pareto front obtained using Matlab for CODE02 ............................................................................................................................................................................................................................................. 67

Figure 37  Comparison of the annual total energy production obtained with CODE02 (optimization by considering individual turbines) and CODE03 (rule-based traditional method) .................................................................................................................................................................................................................................................. 68

Figure 38  Comparison of the annual firm energy production obtained with CODE02 (optimization by considering individual turbines) and CODE03 (rule-based traditional method). .................................................................................................................................................................................................................................................. 68

Figure 39  Efficiency curve for Francis Turbine .................................................................................................................................................................................................................................................. 77
1. INTRODUCTION

One of the topics covered by applied mathematics is mathematical optimization. Although various mathematical optimization problems have been studied earlier, the beginnings of modern mathematical optimization can be traced back to the development and formalization of calculus, and variational calculus by Gottfried Wilhelm (von) Leibniz and Sir Isaac Newton [1]. Since then, mathematical optimization methods have been extensively studied and developed by numerous mathematicians and scientists. The methods of mathematical optimization have been used to solve problems in numerous areas of science and engineering.

One of the areas where optimization has been extensively used is the optimal use of water resources and optimal design of water storage structures, such as large dams. Schardong [2] used a multiobjective differential evolution algorithm (MODE) for optimization of a multiobjective reservoir system operation and compared it with genetic algorithm NSGA-II using some test cases. Afsharian Zadeh [3] used particle swarm optimization (PSO) for reliability based optimal design and operation of a cascading hydropower reservoir systems by formulating the maximization of firm energy production while controlling the reliability level of hydroenergy production as a mixed integer nonlinear program (MINLP). Trezos [4] studied the planning of hydropower production of hydroelectric facility in Southern California using an integer programming (IP) approach. Heydari [5] studied optimization of 5 dams in series on the Karun River and the Dez Dam parallel to those five by formulating the optimization problem as a matrix structure for multiobjective optimization of water supply and energy.
production. Mujumdar and Nirmala [6] presented the development of an operating policy model for hydropower generation in a multi reservoir system using a Bayesian stochastic optimization model that considers uncertainties in both forecast uncertainty and inflow uncertainty. Qi et al. [7] presented the multiobjective optimization of hydropower generation in flood control operation using evolutionary algorithms which take into account preferences of the decision maker. Brown et al. [8] provides a short summary of the past efforts and discusses the future directions for water resources system analysis.

The water control structures, such as dams are built to provide a variety of services [6], which may include water supply for irrigation, drinking, and industrial use, flood control, fire protection, hydroelectric energy generation, and recreation, etc. Thousands of structures have been built around the world to address the specific needs of a region or a country. ICOLD (International Congress on Large Dams) has records of 57,835 large dams (with a dam height greater than 15m) in 96 countries [10]. China occupies the first position with 23,842 large dams followed by the United States of America with 9,265 large dams. Turkey occupies the 10th rank with 972 large dams. These structures are generally built at very high costs. Therefore, it is normal to expect that they would be operated in a manner to achieve maximum efficiency to fully meet the specific purposes for which they were designed.

Reservoirs are operated for flood control, navigation, hydroelectric power, municipal and industrial water supply, irrigation, water quality management, recreation, erosion and sediment control, fish and wildlife enhancement [11]. When a dam is being constructed, one or more of the above are selected as the goals. Often, these goals are ranked as primary, secondary and tertiary goals, etc.
The number of goals to be optimized when optimizing the operation of the reservoir determines whether the process is called single or multi-objective optimization. If there is a single goal to achieve (which is expressed as a mathematical function, called an objective function, to be maximized or minimized), the problem is a single objective optimization problem. The single objective optimization aims to find the best operational strategy for a reservoir by maximizing or minimizing this objective function subject to a set of constraints, which may be physical constraints or operational policy constraints.

Multi-objective optimization seeks to optimize multiple objective functions simultaneously. These multiple objective functions may express conflicting goals. Thus, the multiobjective optimization leads to an optimal compromise, whereby the simultaneous optimization of all the objectives does not necessarily lead to the optimal value of all individual objectives. In fact, there the multiobjective solution may lead to several optimal solutions. These solutions are gathered in a region known as Pareto Optimal [12]. The Pareto Optimal is a collective optimal, or a set of nondominant solutions [2], in which none of the objective functions can be further improved without worsening the value of at least one other objective function. Pareto front represent different combinations of tradeoffs between the multiple objectives of the optimization problem [2]. Since the majority of large dams in the world are designed to meet more than one need, such as the combination of flood control and energy production, their operation involves multiobjective optimization.

Single and multi-objective optimization forms can be mathematically written as follows: Consider the vector of n control variables given below:

$$\bar{X} = [x_1, x_2, \ldots, x_i, \ldots, x_{n-1}, x_n] \quad where \quad x_i \in \mathbb{R}^n$$

Multi Objective Optimization problem states that:
Maximize or Minimize: \( \bar{M}(F_j(\bar{X})) = \bar{M}(F_1(\bar{X}), F_2(\bar{X}), ..., F_J(\bar{X})) \) where \( j = 1, ..., J \)

Subject to \( g_k(x_1, x_2, ..., x_{n-1}, x_n) \leq 0 \quad k = 1, 2, ..., m \)

\( h_\ell(x_1, x_2, ..., x_{n-1}, x_n) = 0 \quad \ell = 1, 2, ..., r \)

where \( x \in \mathbb{R}^n \) is a vector of \( n \) decision variables, \( F_j(\bar{X}) \) are the objective functions, \( g_k(\bar{X}) \) are the inequality constraints, \( h_\ell(\bar{X}) \) are the equality constraints. If \( J \), i.e. the number of objective functions, is equal to one, the optimization problem is becomes a single objective optimization

1.1. Statement of the Problem

This research considers the multiobjective optimization of a multipurpose reservoir, which is designed to provide irrigation water, flood protection and energy production. Input data for the test case considered in this thesis includes monthly volumes of inflow, evaporation, and required volumes for irrigation release to downstream for a period of thirty years.

Figure 1 Storage allocation zones of a reservoir.

Reservoir storage is divided into four storage zones (Figure 1). Dead storage, which is also called inactive storage zone, is used for sediment storage. Located above the dead storage zone, active storage zone is divided into two subzones. The lower part is the buffer zone and the
upper part is the main active storage zone. The buffer zone may suffer from water quality problems. The buffer zone may have sediment particles within the storage, which reduces the water quality for reservoir operation. Therefore this zone is not preferred to use for reservoir operation. The main active storage zone is generally the preferred zone for reservoir operation. The flood control storage is where the excess water is stored in the reservoir to protect of the downstream from floods. Induced surcharge zone is the section, which is used for exceptional peak flows. Active reservoir storage is used for reservoir operation.

Reservoir operation concerns decision making with regard to the amounts of storage volume to be allocated and used for different purposes, such as irrigation, hydropower generation, etc. Starting with an initial water storage volume of $S_t$ (in hm$^3$) at the beginning of the month $t$, the storage at the end of the month $S_{end}$ (or $S_{t+1}$) is calculated taking into inflow volume, $I$, and outflow volume, $O$, using the mass (or volume) conservation equation given by Eq (3), which can be recursively applied for all months from 1 to $T$:

$$S_{end} = S_{t+1} = S_t + I - O \quad t = 1 \ldots T$$

(3)

In this recursive equation, the end of the month volume for the month $t$, i.e. $S_{end}$ (or $S_{t+1}$), becomes the initial storage volume for the month $t + 1$.

The volume of water entering the reservoir is defined as inflow volume, which includes the sum of runoff due to rainfall and snowmelt, and the water released from any upstream reservoirs. The outflow volume includes losses due to evaporation, releases for irrigation, drinking water supply, environmental minimum water (the amount of water released each month to support and maintain the local ecosystem), and energy water (passing through turbines to
generate hydroelectric energy). In his research, only evaporation, irrigation and energy water volumes are considered as outflow (Figure 2).

Figure 2  Illustration of reservoir operation.

The present thesis seeks to optimize the hydroelectric energy production using two objective functions:

1. Maximize the total energy production over a period of specified number of years for which the data is available, and
2. Maximize the firm energy production for all individual years of the specified period.

Firm energy is defined as the amount of guaranteed hydroelectric energy produced continuously throughout the year. Since there are two objectives, a multi-objective optimization method is to be used.

The numerical solution of the reservoir routing equation and the calculation of friction losses in the penstocks involved in the computation of the objective functions are complex and nonlinear. Therefore, it was decided to use genetic algorithm techniques that are already available as functions in the Global Optimization Tool of the Matlab software.
Two separate Matlab codes were written for multiobjective optimization of the reservoir operation for hydroelectric energy production using different algorithms for calculating the energy production from an allocated storage volume. These alternative algorithms for hydroelectric energy production are as follows:

1. The algorithm for the allocation of volumes for firm energy and secondary energy or the production of the hydroelectric energy does not consider the operation of individual turbines. The hydroelectric energy generation is computed using a constant nominal head loss value irrespective of the turbine discharge. This code will be referred to as CODE01 from hereon.

2. The algorithm for the allocation of volumes for firm energy and secondary energy and the production of the hydroelectric energy take into account the number of turbines, their characteristics, and the distribution of allocated firm and secondary energy volumes between the individual turbines according to a specified operational policy. For each turbine, the energy produced is calculated by considering the true friction head loss computed based on the turbine discharge and the penstock characteristics. This code will be referred to as CODE02 from hereon.

A third Matlab code, which will be called CODE03 from hereon, was also written to compute the firm and secondary hydroelectric energy generation using a traditional rule-based method to allocate the volumes for firm and secondary energy production. This Matlab code also considers the energy production by individual turbines based on the partition of the volumes allocated for firm and secondary energy and the corresponding friction head losses.

1.2. Test Case for Evaluating Developed Methods
Kayraktepe Dam located in southern Turkey near Mediterranean coast was selected as the test case for this thesis study. Revised planning report and various other data concerning Kayraktepe Dam were kindly provided by the State Water Works Turkey (DSI). Along with the reports and data, the DSI also provided a spreadsheet of reservoir operation study for the computation of the projected hydroelectric energy production, which is composed of firm energy and secondary energy, during a period of 30 years. These computations were carried out within the framework of the revised planning study, which considered the technical and economic feasibility of the irrigation of an area of 7,425 ha using pressure pipe flow and the generation of the hydroelectric power.

The worksheet included monthly volumes of inflow, losses due to evaporation, and various outflows (such as irrigation, drinking, environmental and energy water) for a 30-year (360 months) period extending from October 1984 to September 2013.

The three Matlab codes (CODE01, CODE02, and CODE03) were applied to the test case of Kayraktepe Dam. It is important to note that Kayraktepe Dam and reservoir is a multipurpose facility, whose primary purposes are flood protection and irrigation. Energy production is the secondary purpose.

The two Matlab codes with genetic algorithm optimization (CODE01 and CODE02) in the present study, however, focus on the maximization of the annual firm and total hydroelectric energy production. The third code (CODE03) applies a rule based traditional method to allocate the energy volume, which considers the operation of the individual turbines. For all thee Matlab codes, the monthly volumes required for the irrigation are treated as known time series, which is given as input to the program together with the know time series of losses through evaporation and leakage. All three codes receive these time series as input and accommodate them as outflow
volumes to be satisfied. The use of the remaining available storage volume is then optimized for energy production by maximizing the total energy production over the entire period of study and the individual annual firm-energy production amounts.

This thesis divided into seven chapters. The first chapter presents statement of problem and explains the purpose of the study. The second chapter explains the general information about optimization, genetic algorithm and compiles the literature sources for current study. The optimization model, problem formulation and Matlab programming for genetic algorithm are explained in Chapter 3. Traditional method for reservoir operation is defined in Chapter 4. Chapter 5 presents a case of study with the problem assumptions and input variables for energy optimization. The sixth chapter on the discussion of results and conclusions presents the results of the energy production amounts obtained using the three Matlab codes and compare them with the original energy production calculations. The last chapter discusses the future work.
2. OPTIMIZATION OF RESERVOIR OPERATION

Various optimization techniques have been used for multiobjective reservoir operation. One of the techniques that can be used for solving non-linear multiobjective reservoir optimization problems is the genetic algorithm. In this study, the genetic algorithm will be used for the nonlinear reservoir optimization problem in which monthly total energy production and firm energy maximization are performed. This chapter provides some general information about multiobjective optimization, genetic algorithm technique, and highlights some of the previous work related to the topic of this thesis.

2.1. General Information

The modeling of a multiobjective optimization problem for a multipurpose reservoir, such as the one considered in the present thesis, requires identification of the decision or control variable vector $\vec{X}$, the vector of objective functions $\vec{M}(\vec{F}(\vec{X}))$, in which individual objective functions are defined based on the vector $\vec{X}$, and the set of constraints, which can be inequality constraints $g_k(x_1, x_2, \ldots, x_{n-1}, x_n)$, or equality constraints $h_l(x_1, x_2, \ldots, x_{n-1}, x_n)$, referring to Eq.(2).

Monthly reservoir storage volumes released for energy production (the total monthly volume for firm energy and secondary energy) are chosen as decision variables. The constraints impose lower and upper bounds for the values of the decision variables based on the discharge characteristics of the turbines. In addition, it is required that the reservoir is operated between
specified limits, and the mass conservation is imposed in the form of reservoir routing equation, see Eq. (3) which expresses the conservation of storage volume (water is considered incompressible). Other constraints and/or rules are also imposed on how the turbines should be operated in CODE02) and CODE03.

Multiobjective optimization may produce multiple feasible solutions, within the feasible region that satisfies the constraints. These multiple feasible solutions do not necessarily maximize all individual objectives (Figure 3).

![Figure 3 Concept of solution space.](image)

2.2. Nonlinear Optimization and Genetic Algorithm Technique

In linear programming, the optimal solution, which both local and global, is located at the vertices of the feasible region. The situation is different for nonlinear programming. Nonlinear problem may be more than one local maximum and minimum point. Referring to Figure 4, the function $z = f(x)$ has multiple local maximum points, which are marked as “a”, “b”, “c” and “d”. However, the global maximum is located at point “c”.

11
Nonlinear and linear optimization problems can be formulated as constrained or unconstrained. In constrained optimization formulation, constraints are defined for the decision variables and/or the objective functions. Unconstrained optimization problems incorporate interior [9] penalty functions into the objective function to force the results into the feasible region. Ideally, the penalty function becomes active when a chromosome is not satisfying the objective functions to guide the chromosome into the feasible region. However, for multiobjective problems with multiple constraints defining the right form of penalty function may pose a serious challenge in itself.

![Illustration of global and local maximum points in a nonlinear problem.](image)

Optimization problems related to water resources are complex and nonlinear [2]. There is a nonlinear relationship between the components to be used for optimization. In this case, nonlinear models may be preferred or optimization may be performed by using the linearization method. For example, when energy production is calculated, the relationship between released volume and the head loss is not linear.
2.3. Genetic Algorithm

GA is evaluated as a robust, efficient and quickly converged algorithm, and does not appear to cause any significant change in the results of program re-running. In this study, GA is preferred because of the high percentage of success in constrained, multidimensional and nonlinear problems.

The genetic algorithm was first described by the Holland [13] and later developed by Goldberg [14] and Michalewicz [15]. It is based on natural selection and natural evolution. Goldberg explains the structures that differentiate genetic algorithm from other algorithms with the following four approaches [14].

- GAs do not work with parameters themselves but rather with the group created by coding the parameters.
- GAs focus on the whole set of points rather than limiting their search to a single point.
- The information that GAs use is based on objective function, rather than auxiliary knowledge such as derivatives.
- GAs prefer probabilistic transition rules to deterministic rules.

The genetic algorithm makes a global search and it is population based. While the potential good solutions survive in the genetic algorithm, other solutions are removed from the population. The convenient solutions are determined according to the objective function, which is also called a fitness function. Crossovers and mutations operators of genetic algorithm method are used to create new generations.
In the genetic algorithm, each candidate solution is called a chromosome. The set of all candidate solutions is called the population. The population size is a parameter that defines the number of candidate solutions. As the population grows from generation to generation, poor solutions tend to disappear and good solutions tend to produce better solutions. Firstly, the initial population is randomly generated considering problem constraints. The value that determines how good a chromosome is for an optimal solution is called a fitness value. In the GA, the fitness value of the solution increases its chances of being selected for mutation and crossover process to create a new population. Figure 5 shows the main steps of the genetic algorithm. These steps "Selection", "Crossover", and "Mutation" are briefly discussed in the following subsections.

2.3.1. Selection (Reproduction)

To create a new generation, the chromosomes with high fitness rates are selected as parents. This process is called "selection". The selection of individuals is random process but fitter individuals have high probability from the other chromosomes. There are many selection methods such as the roulette wheel, tournament, and rank selection.
In the roulette wheel method, the probabilities of selection for chromosomes are proportional to the fitness values.

For the rank selection method, the choice of the parents depends on the rank of the individuals, but sorting is not related to fitness values.

In this thesis, the “Tournament” method is preferred. To apply the tournament selection method, a certain number of individuals are randomly selected from the population and selection is made among these individuals by taking into account their fitness values. Figure 6 schematically illustrates how the tournament method works.

![Figure 6](image)

**Figure 6** Tournament method for “Selection” step.

### 2.3.2. Crossover

The crossover is the process of creating a new child from two selected parents. The purpose of crossover is to create a better generation by swapping genes from the parents who are selected in the first step. There is not a new generation until the crossover step is completed.
After swapping genes among the parents, a new and “better” generation is created. The creation of a better generation continues until the desired result is achieved. The criterion that determines whether there is a crossover between selected pairs is the crossover probability. Crossover probability is generally between 0.5 and 1. There is a wide range of crossover methods such as one point crossover, two point crossovers, uniform crossover, and intermediate crossover.

Figure 7, illustrates how the first three methods are applied. One point method randomly selects a single location in the gene sequence of both parents. The selected location is shown by a thick line in the figure. The genes of the parents to the left and right side of the selected location are swapped to create two offsprings. A similar procedure is use for the two point crossover with the difference that two locations are selected in the gene sequence for swapping the genes of two parents. In the uniform crossover method, instead of swapping blocks of genes, the decision to swap or not the genes is decided at the individual gene level based on a probability value, which is generally selected as 0.5.

In the present thesis, the intermediate cross over method is used. The explanations for this method will be provided in Chapter 1.
2.3.3. Mutation

After a certain period of new generation production, chromosomes in the next generation may repeat each other. This may lead to a decrease in the genetic diversity of new populations. The aim of the mutation is to preserve the genetic diversity. The probability of mutation is the operator that determines whether or not mutations will be made on genes. The mutation probability usually takes small values. If this value is 100%, all the chromosomes change, while if it is 0%, the chromosomes remain the same.
2.4. Literature Survey on the Use of Genetic Algorithms for Optimization of Reservoir Operations

Traditional methods do not always offer the optimal solution for reservoir operation. The genetic algorithm is considered a robust and reliable technique for optimization of reservoir operation for single or multiple objectives. Numerous optimization problems for single and multi-reservoirs have been studied in the past using this algorithm. In these studies, several topics such as optimizing energy production, meeting irrigation and drinking water needs; providing ecological, environmental and socio-economic benefits; flood control and so on have been handled using the genetic algorithm.

Esat and Hall have conducted research to show the applicability of the genetic algorithm in multipurpose reservoirs [16]. Maximization of energy production and irrigation water was used as objective functions in this study. Capacity of reservoir release and storage were used as constraints. In this work, the adequacy of the genetic algorithm for water resources systems is demonstrated by comparing with other techniques. Oliveira and Loucks [17] evaluated the operating rules for multi-reservoir systems using genetic algorithm. The performance of the genetic algorithm was sufficient and practicable for different scenarios and problems in their studies. Another optimization research with genetic algorithms was published by Wardlaw and Sharif [18]. Previously published results, which were solved with dynamic programming for the same reservoir problem, were compared with results from the genetic algorithm in this study. The results of the genetic algorithm were evaluated and were determined to be robust and applicable even in complex systems. Optimization studies by genetic algorithm for three reservoirs in the Colorado River Storage Project were implemented by Hincal [19]. Actual operation data and the operation data resulting from the optimization study were compared. The
results showed that the genetic algorithm is competitive and constitutes a robust alternative to other optimization techniques. The studies made for the reservoir optimization with genetic algorithm have been done by using different coding methods throughout the years. Researchers have attempted to obtain optimal solutions using their own software codes or solvers provided by software packages.

Matlab is one of the software programs that offer solvers for different optimization methods used in engineering studies, including tools for the genetic algorithm. There are two solvers for the genetic algorithm, one for single objective optimization and the other for multiobjective optimization.

The genetic algorithm toolbox (GAT) can be used for both multi objective and single objective problems. The applicability of these solvers for reservoir optimization has been tested by various researchers. Hashemi et al. [20] studied reservoir optimization using GAT for different inflow probabilities. The downstream water needs were provided by regulating the reservoir volume balance for different scenarios. GAT gave fast and reasonable results. Another work that is optimized using GAT is by Devisree and Nowshaj [21]. This work was done to maximize the annual amounts of energy as well as irrigation water. The obtained results are compared with the results obtained with the linear programming solution. It was observed that the results were reasonably close to each other and satisfactory. In these studies, models are designed for a period one year, for which they provided good results.

The efficiency of the genetic algorithm has been tested for long periods (different flow scenarios). Research in this area was made by Yousif H. Al Ageeli, Lee and Aziz [22]. Energy generation was simulated for reservoir operation by using traditional method, and he compared it with results obtained by using a single objective genetic algorithm. Their aim is to maximize the
annual electricity generated. In addition, the inclusion and non-inclusion of precipitation and evaporation in optimization were compared by them. They concluded that in the scenario, which includes precipitation and evaporation, 65% more energy was produced and genetic algorithm was successful in different scenarios.

Considering all this, in this thesis, we will optimize the energy production of reservoir by using structural data, inflows and other necessary releases, which are available for a period of thirty years for the selected test case in Turkey. The purpose of this research is to evaluate the applicability and efficiency of the genetic algorithm for the problem of maximizing firm energy and annual energy production. The results obtained using genetic algorithm are then compared with both traditional ruled based reservoir operation results and the original calculations performed by an engineering company, which provided by the courtesy of DSI.
3. GENETIC ALGORITHM OPTIMIZATION OF RESERVOIR OPERATIONS USING MATLAB

Global Optimization Toolbox of Matlab provides a series of solvers to solve different types optimization problems (linear, quadratic, integer, and nonlinear) by maximizing or minimizing one or more objective functions under defined constraints. One of these solvers is the genetic algorithm. In Chapter 2, some examples of the past studies using Matlab solvers were given. This chapter presents the formulation of the optimization problem considered in this thesis and its programming by writing a code in Matlab that uses the existing genetic algorithm solver in Global Optimization Toolbox.

3.1. Genetic Algorithm Programming Using Matlab

This research aims to optimize operation of a multipurpose reservoir from the point of view of maximizing the hydropower generation, which is composed of firm energy and secondary energy parts, while providing required storage volumes for various other purposes, such as irrigation, drinking water, etc. and taking into account losses of storage volume due to evaporation.

The storage volumes allocated for the energy production for each month of the selected study period are the decision variables. In the genetic algorithm framework, they are referred to genes. Thus, each chromosome holds as many genes as the number of months in the study period. Each chromosome represents an individual. The fitness of each chromosome must be
evaluated by calculating the values of the objective functions using the volumes for energy production stored in its genes.

The set of all chromosomes is the population. The initial population with a specific number of chromosomes is generated by assigning the random values to the genes by considering the lower and upper bound values imposed as constraints. The genetic algorithm method uses crossover and mutation procedures to evolve the initial population to improve the fitness of the chromosomes, and thereby reach a set of Pareto optimal chromosomes that satisfy the constraints and have the highest fitness values.

Matlab offers two ways of using genetic algorithm solvers. One method is to use the Genetic Algorithm Toolbox (GAT), which offers a user interface to define the objective function, various types of constraints, and the solution parameters. The second method is calling
the multiobjective genetic algorithm function at command line, referring to Eq. (4). The interface of first method is shown in Figure 8. In the present thesis, the second method was used.

In this thesis, we chose to write a Matlab code to solve the multiobjective optimization of reservoir operation for optimal energy production. In Matlab, the search for the vector of optimal solutions, \( x \), located on the Pareto front of multiple objective functions, subject to linear equality and inequality constraints, and nonlinear constraints, is initiated by calling the following function
\[
x = \text{gamultiobj}(\text{fitnessfcn}, \text{nvars}, A, b, Aeq, beq, lb, ub, options)
\] (4)

Various parameters defined in this equation are briefly explained below:

- The argument \( \text{fitnessfcn} \) in Eq. (4) represents reference to an “.m” file, which is a Matlab script file containing the script defining the objective functions of \( x \) decision variables, which are called fitness functions in the context of genetic algorithm.
- The argument \( \text{nvars} \) in Eq. (4) represents the number of decision variables, which is also the size of the vector \( x \).
- The linear inequality constraints are defined as a matrix relationship given by
\[
A \cdot x \leq b
\] (5)
with \( A \) as the coefficient matrix for the vector of decision variables \( x \), in the expressions of inequality constraints and \( \text{beq} \) as the vector of inequality constants. The coefficient matrix \( A \) and the vector \( b \) are given as arguments in the call shown in Eq. (4).
- The linear equality constraints are defined as a matrix relationship given by
\[
Aeq \cdot x = \text{beq}
\] (6)
with \( Aeq \) as the coefficient matrix for the vector of decision variables \( x \), in the expressions of equality constrains and \( \text{beq} \) as the vector of equality constants. The
The coefficient matrix $A_{eq}$ and the vector $beq$ are given as arguments in the call shown in Eq. (4).

- The lower and upper bounds on the decision variables $x$ are expressed as follows

$$lb \leq x \leq ub$$

Equation (7)

In Eq. (4), the variables $lb$ and $ub$ are the vectors containing the lower and upper bounds of the decision variables stored in vector $x$, respectively.

- The argument options in Eq. (4) stands for various settings that can be specified for the genetic algorithm solver in Matlab. Examples of parameters that can be modified are the population size, the selection method, the methods for crossover and mutation, the stopping criteria, etc. All settings have default values and this argument is optional. If the default values are not acceptable, the user can use the command "optimoptions" to change the settings.

The genetic algorithm tool in Matlab provides numerous plot options as shown in the Figure 9. These plots can be used get a better understanding of the solution provided by the genetic algorithm solver.

![Plot options for the genetic algorithm in Matlab.](image)

Figure 9 Plot options for the genetic algorithm in Matlab.
Some criteria must be defined for stopping the iterations of the genetic algorithm by creating a new population. Generally more than one stopping criterion is defined and the library program “gamultiobj” automatically ends the iteration when one of these stopping criteria is satisfied. The time limit can be set to stop program running process. In this thesis, all time limits are defined infinitely. Another stopping criterion concerns the maximum number of iterations (generation) allowed. This limit is defined as $100 \times nvars$ in the program itself. Users can define this value according to their needs. The maximum number of iterations for this study was set as a 3,000, but the sensitivity to this number was also investigated by running the program with different number of maximum number of iterations. The function tolerance shows the average relative change in the fitness function. The limit value for function tolerance is $10^{-4}$. Another stopping criterion is constraint tolerance. Constraint tolerance determines feasibility with respect to nonlinear constraints. The value in “gamultiobj” is $10^{-3}$. In this thesis, the default values of function and constraint tolerance were used.

### 3.2. Formulation of the Multiobjective Optimization Problem in Matlab

In this thesis, the reservoir operation is optimized by satisfying simultaneously two objective functions related to energy production. Before explaining these functions, it would be beneficial to provide information on firm energy, secondary energy and total energy generation in a hydropower plant.

Firm energy is defined as the amount of energy that is guaranteed to be generated at all times. In the present case, it will be assumed that this is a specified amount of targeted firm energy generation defined during the planning and feasibility study of the system comprised of
the reservoir, the dam and the hydropower plant. It is assumed that the firm energy production is continuous 24/7.

Any energy generation in addition to the firm energy is called secondary energy. It is assumed that the secondary energy is provided only when a sufficient storage volume is available. It is also assumed that the secondary energy may be produced during only a limited amount of time based on the availability of storage volume.

The sum of firm energy and the secondary energy produced in a given month is called total energy. It is desirable that the total energy is always equal to or greater than the targeted firm energy. However, if inflow discharge is not sufficient and the reservoir level is low, it may so happen that the targeted firm energy cannot be generated and the total energy produced becomes less than the targeted firm energy production.

Let us consider that, for a given reservoir-dam system, the analysis will be carried out over a period of \( n \) years. The total energy production in GWh for the month \( i \) in the year \( m \) is denoted as \( HP_{m,i} \). Let us also assume that the targeted firm energy amount in GWh is denoted by \( PF \). Then, the two objective functions can be described as follows.

The first objective function (or fitness function), which aims to optimize that total energy production in GWh for the period of \( n \) years, is written as follows:

\[
\text{Maximize: } PE(1) = \sum_{m=1}^{n} \sum_{i=1}^{12} HP_{m,i}
\]  

(8)

It is to be noted that the value of this first objective function is stored in Matlab as the first element of a vector with two elements, i.e. \( PE(1) \).

The optimization aims also to achieve the targeted firm energy amount in GWh over the entire period of \( n \)-years. Considering that the total energy in GWh produced at any month should
normally be equal to or greater than the targeted firm energy, the firm-energy deficiency for the month $i$ in the year $m$ can be defined as:

$$DFIRM_{m,i} = \begin{cases} 0 & \text{if } HP_{m,i} \geq PF \\ PF - HP_{m,i} & \text{if } HP_{m,i} < PF \end{cases}$$

(9)

For an optimal operation of the reservoir the sum of the firm-energy deficiency over the period of $n$-years should be minimized. Therefore, the second objective function (fitness function), expresses the minimization of the total firm-energy deficiency over the period of $n$-years:

$$Minimize: PE(2) = \sum_{m=1}^{n} \sum_{i=1}^{12} DFIRM_{m,i}$$

(10)

It is to be noted that the value of this second objective function is stored in Matlab as the second element of a vector with two elements, i.e. $PE(2)$.

The values of the objective functions $PE(1)$ and $PE(2)$ are calculated for each chromosome in the population. Therefore, in the Matlab code, the PE is treated as a two dimensional results vector with two columns corresponding to $PE(1)$ and $PE(2)$ and as many rows as the size of the population.

The calculation of the vector of objective functions, $(i), i = 1,2$, is programmed as a separate script file, and the name of the script file is provided as the first argument “fitnessfcn” to “gamultiobj” in Eq. (4).

In this study, the time interval is chosen as month. Therefore, it is assumed that the inflow, outflow, and storage-loss volumes are available on a monthly basis for the duration of selected $n$-years. These time series are treated as input for the optimization code. As explained in the previous section, the decision variables for the problem will be to monthly storage volume allocated to the production of the total energy, comprised of firm and secondary energy parts. Therefore, the variable “nvars” in Eq. (4) corresponds to the number of months during the
selected $n$-year period. Moreover, in terms of implementation of genetic algorithms, the variable “nvars” correspond also to the number of genes for each chromosome. Thus, each chromosome, i.e. each member of the population, has $nvars = n \times 12$ genes. The vectors “lb” and “ub” have also nvars elements. They define the lower and upper bounds of the monthly storage values allocated to total energy production. These limits, of course, depend on the characteristics of the turbines.

The storage volume in the reservoir, $S \ (hm^3)$, is a function of the water surface elevation, $z \ (m \ a. \ s. \ l.)$. This relationship can be expressed as follows:

$$S = f(z)$$

(11)

This relationship can be inverted to give water surface elevation in terms of the storage volume:

$$z = g(S)$$

(12)

The mass balance or volume balance since the density of water is constant, for the storage volume in the reservoir is given by

$$\frac{dS}{dt} = Q_I - Q_o$$

(13)

where $Q_I$ is the inflow discharge $(m^3/s)$ into the reservoir $Q_o$ is the outflow discharge $(m^3/s)$ from the reservoir. Assuming that the time interval of $dt = \Delta t$ is one month, and $Q_I$ and $Q_o$ represent monthly average discharges, the discretization of Eq. (13) gives

$$S_{t+1} - S_t = (Q_I - Q_o)\Delta t = (I_t - O_t)$$

(14)

where $I_t$ is the monthly inflow volume, $O_t$ is the monthly outflow volume. $S_{t+1}$ represents the storage volume in the reservoir at the end of the month, and $S_t$ is the reservoir volume at the beginning of the month. This balance equation (also called the reservoir routing equation) must be respected at all times. The monthly outflow volume may include the following:

- $EV_t$ is the monthly storage volume $(hm^3)$ lost by evaporation.
IR_t is the monthly storage volume (hm³) released for irrigation purposes.

EN_t is the monthly storage volume (hm³) allocated for energy production (i.e. the water released through the turbines). In fact, the set of EN_t values for all months of the n-years represent the genes of chromosomes.

Thus, the reservoir volume at the end of the month is given by

\[ S_{end} = S_{t+1} = S_t + I_t - (EV_t + IR_t + EN_t) \] \hspace{2cm} (15)

The input data for the problem includes the lower and upper limits of operation for the reservoir, which are denoted as S_min and S_max, respectively. The corresponding water surface elevations are denoted as z_min and z_max. The operational policy is set such that, the volume in the reservoir should be kept as close as possible to S_max. If after making all possible releases, the reservoir storage volume at the end of the month is greater than S_max, the excess volume is spilled over the spillway and/or bottom outlets to bring the end of the month volume to S_max.

Thus, the over flow condition can be expressed as

\[ Overflow = \begin{cases} 
S_{end} - S_{max} & \text{if } S_{end} > S_{max} \\
0 & \text{if } S_{end} \leq S_{max}
\end{cases} \] \hspace{2cm} (16)

The storage volume in the reservoir can go below S_max in order to produce at least the targeted firm energy amount. However, even the firm energy production cannot be met if the storage in the reservoir at the end of the month goes below the minimum storage value S_min.

Therefore, the optimization code is designed to make sure that S_end \geq S_min.

As mentioned above, the set of EN_t values for all months of the n-years are the nvars decision variables stored as the genes of chromosomes. Thus, referring to Eq. (4), we can write that

\[ x = (x_1, x_2, ..., x_{nvars}) = (EN_1, EN_2, ..., EN_{nvars}) \] \hspace{2cm} (17)
The vectors of lower and upper bounds “$lb$” and “$ub$” Eq. (4) define the lower and upper bounds for the $EN_t$ values. The upper limit is defined by the total volume that can be used to generate energy if all the turbines work at full capacity during the entire month. The lower bound for the $EN_t$ is assumed to be zero, meaning that no energy is produced, which is possible under certain circumstances.

In general, the efficiency of the turbine varies with the flow rate. In this thesis, due to lack of information about the turbines, the turbines were assumed to operate with a constant value of efficiency corresponding to the maximum efficiency, regardless of the flow rate.

3.3. Description of the Matlab Code: A Study Case

The flowchart of the Matlab code is given in Figure 10. The program “MAIN.m” controls the entire operation. It first reads the input values listed in Table 1 by calling the subprogram “inputdatas.m”. Main program initializes the vectors of lower and upper bounds for the decision variable and defines various settings for the Matlab function “gamultiobj”. Then, the “MAIN.m” calls “gamulitobj” to optimize the reservoir operation for energy generation using the method of multiobjective genetic algorithm. The names of two Matlab subprograms “OBJECTIVE_FUNCTION.m” and “constraint.m” are sent to “gamulitobj” as arguments.

Table 1 List of input variables for the Matlab genetic algorithm code.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Units</th>
<th>Number of Elements</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I$</td>
<td>$hm^3$</td>
<td>$n \times 12$</td>
<td>Monthly total inflow volume</td>
</tr>
<tr>
<td>$IR$</td>
<td>$hm^3$</td>
<td>$n \times 12$</td>
<td>Monthly volumes required for the irrigation</td>
</tr>
<tr>
<td>$EV$</td>
<td>$hm^3$</td>
<td>$n \times 12$</td>
<td>Monthly total evaporation volume</td>
</tr>
<tr>
<td>$MROL$</td>
<td>m a.s.l.</td>
<td>1</td>
<td>Maximum reservoir operation level</td>
</tr>
<tr>
<td>Symbol</td>
<td>Unit</td>
<td>Definition</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>MNROL</td>
<td>m a. s.l.</td>
<td>Minimum reservoir operation level</td>
<td></td>
</tr>
<tr>
<td>MROV</td>
<td>hm³</td>
<td>Maximum reservoir operation volume</td>
<td></td>
</tr>
<tr>
<td>MNROV</td>
<td>hm³</td>
<td>Minimum reservoir operation volume</td>
<td></td>
</tr>
<tr>
<td>D1</td>
<td>m</td>
<td>Number of small turbines Penstock diameter for small turbine(s)</td>
<td></td>
</tr>
<tr>
<td>D2</td>
<td>m</td>
<td>Number of large turbines Penstock diameter for large turbine(s)</td>
<td></td>
</tr>
<tr>
<td>MTC</td>
<td>hm³</td>
<td>Maximum monthly total storage volume that can be allocated for energy production</td>
<td></td>
</tr>
<tr>
<td>MNTC</td>
<td>hm³</td>
<td>Minimum monthly total storage volume that can be allocated for energy production</td>
<td></td>
</tr>
<tr>
<td>TC1</td>
<td>hm³</td>
<td>Number of small turbines Total volume that can be used by a small turbine operating continuously at full capacity during a month</td>
<td></td>
</tr>
<tr>
<td>TC2</td>
<td>hm³</td>
<td>Number of large turbines Total volume that can be used by a large turbine operating continuously at full capacity during a month</td>
<td></td>
</tr>
<tr>
<td>v</td>
<td>m²/s</td>
<td>Viscosity</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>m</td>
<td>Number of penstocks Penstock length</td>
<td></td>
</tr>
<tr>
<td>ks</td>
<td>m</td>
<td>Pipe roughness</td>
<td></td>
</tr>
<tr>
<td>IRV</td>
<td>hm³</td>
<td>Reservoir volume at the beginning of the simulation</td>
<td></td>
</tr>
</tbody>
</table>

The library function “gamultiobj” creates the initial population randomly by respecting the lower and upper bounds transmitted also as arguments. The fitness values $P_E(1)$ and $P_E(2)$ are evaluated for all chromosomes by calling the subprogram “OBJECTIVE_FUNCTION.m” and the subprogram “constraint.m”, which defines the operational constraints to operate the reservoir at a volume greater than or equal to $S_{min}$. To compute the objective function values, “OBJECTIVE_FUNCTION.m” calls several other subroutines:

- The storage subroutine (“storage.m”) calculates the storage volume at the end of the month using (15).
- Pool-elevation subroutine ("pool_elevation.m") calculates the pool elevation corresponding to a given storage volume.

- The subroutine for friction losses ("losses.m") calculates the friction loss in the penstock based on the turbine discharge. It is used only when the operation of the turbines are considered (CODE02 and CODE03).

- The energy subroutine ("energy.m") calculates the hydroelectric energy produced.

"gamultiobj" checks whether the stopping criteria is reached. If the stopping criteria is not reached, a new population is created by going through the steps of selection, crossover and mutation and a new iteration loop is executed.

If one of the stopping criteria is reached, the control is returned to the “MAIN.m”, which calls “lastdata.m” and prepares output for further analysis and plotting.

How the “OBJECTIVE_FUNCTION.m” and the various subroutines are called to accomplish specific tasks by them, requires looking into the hydraulics of hydroelectric energy generation at a dam, which is discussed in the next subsection.
Figure 10  Flowchart of the genetic-algorithm Matlab code for the optimization of energy generation.
3.4. The Concept of Energy Generation at a Dam and the Formulation of Hydropower Generation

Figure 11 assumes that the hydroelectric energy is generated using a Francis turbine, which is a reaction-type turbine suitable for medium head and medium discharges [23]. The water is delivered into the spiral casing of the turbine passing through a penstock of length $L_p$ having a diameter of $D_p$ and an absolute roughness of $k_s$. The available potential energy for the production of the hydropower is equal to the gross head $H_G$, which is the vertical distance between the energy head in the pool and the energy head in the tailrace canal.

![Figure 11 Hydraulics of energy generation at a dam](https://commons.wikimedia.org/wiki/File:M_vs_francis_schnitt_1_zoom.jpg)

Scroll casing has openings to pass the flow into the runner, which is the rotating component of the Francis turbine [24]. A cutaway section of the Francis turbine is shown in the insert in Figure 11. The scroll casing is designed to deliver the flow into the runner, the rotating part of the turbine, with constant velocity over its entire inner periphery. The angle of flow into

---

1 The figure of the Francis turbine in the insert is taken from https://commons.wikimedia.org/wiki/File:M_vs_francis_schnitt_1_zoom.jpg. The copyright belongs to Voith-Siemens, Germany.
the runner is controlled by stay (or guide) vanes which help to transfer momentum to the blades of the runner. Runner has specially designed curved blades mounted between an upper and lower circular plate. The tangential component of the impact force of the water on the runner blade rotates the runner and the central shaft, which is connected to it. The upper end of the shaft is connected to the generator. The rate of flow through the Francis turbine is controlled by a series of wicket gates actuated by a special mechanism.

The power generated by the Francis turbine can be calculated from [22]

\[ HP = \eta \rho g Q H_{eff} = \eta \gamma Q H_{eff} \]  \hspace{1cm} (18)

where \( HP \) is hydroelectric power produced (Watt), \( H_{eff} \) is the net head available for turbine (m), \( Q \) is discharge from the turbine (m³/s), \( \eta \) is overall turbine efficiency, \( \gamma \) is specific weight of water (N/m³), \( \rho \) is density of water (kg/m³), \( g \) is gravitational acceleration (m/s²).

Although the potential energy available for the hydropower generation is given by the gross head \( H_G \), due to friction losses in the penstock, the head that can be used for the energy production is the net head denoted by \( H_{eff} \). The expression for the effective head can be derived by writing the equation of conservation of energy (Bernoulli equation) between the reservoir pool and the tailrace canal as follows [23]:

\[ Z_u + \frac{p_u}{\gamma} + \frac{V_u^2}{2g} = Z_d + \frac{p_d}{\gamma} + \frac{V_d^2}{2g} + h_f + H_{eff} \]  \hspace{1cm} (19)

The air pressure of the surface of the reservoir and the tailrace canal is atmospheric:

\[ \frac{p_u}{\gamma} = \frac{p_d}{\gamma} = 0 \]  \hspace{1cm} (20)

Assuming that the surface velocities at in the reservoir (upstream) and the tailrace canal (downstream) are small enough to be neglected, we write:
\begin{equation}
\frac{V_u^2}{2g} = \frac{V_d^2}{2g} = 0 \tag{21}
\end{equation}

Eq. (19) can now be simplified to yield an expression for the effective head used in the energy equation, 

\begin{equation}
H_{eff} = Z_u - Z_d - h_f - h_s = H_G - h_f - h_s \tag{22}
\end{equation}

where \(Z_u\) is the upstream water surface elevation, \(Z_d\) is the downstream water surface elevation, \(h_f\) is the linear head loss due to friction, \(h_s\) is sum of all singular losses.

Note that the difference between \(Z_u\) and \(Z_d\) is the gross head \(H_G\). Thus the effective head is found by subtracting linear friction losses, \(h_f\), and local singular losses, \(h_s\) (such as inlet losses, bed losses, and outlet losses) from the gross head. Usually, the penstocks are designed to minimize local losses. Therefore, in the present study, it will be assumed that the local losses are negligible; i.e. \(h_s \cong 0\).

The Darcy Weisbach equation can be used to express the friction loss in a penstock [26]

\begin{equation}
h_f = f \frac{L_p}{D_p} \frac{V_p^2}{2g} = f \frac{L_p}{D_p} \frac{Q^2}{2gA_p^2} \tag{23}
\end{equation}

where \(f\) is the friction coefficient (Darcy Weisbach coefficient), \(D_p\) is pipe diameter (m), \(L_p\) is pipe length, and \(A_p = \pi D_p^2 / 4\) is the cross sectional area of the circular penstock.

The friction coefficient \(f\) can be calculated using Colebrook and White equation [27]. However, this is a transcendental equation that can only be solved by an iterative procedure. Several approximate explicit equations have been proposed by various researchers. These are summarized in Genić [25], who recommends the equations by Haaland (1983) and by Altshul (1952). Haaland’s equation, which is valid in the range \(4 \times 10^3 < Re < 10^8\) and \(10^{-6} < \varepsilon < 5 \times 10^{-2}\), is given as
\[ f = \frac{1}{-1.8 * \log \left( \left( \frac{\varepsilon}{3.7} \right)^{1.11} + \frac{6.9}{Re} \right)^2} \]  

Altshul’s equation, which is valid in the range \( 4 \times 10^3 < Re < 10^8 \) and \( 10^{-4} < \varepsilon < 3 \times 10^{-2} \), is given as

\[ f = 0.11 \left( \frac{68}{Re} + \varepsilon \right)^{0.25} \]  

In equations of Haaland and Althsul, the relative roughness is defined as

\[ \varepsilon = k_s / D \]  

and the Reynolds number of the flow in the penstock is defined as,

\[ Re = \frac{V_p D_p}{\nu} = \frac{Q D_p}{A_p \nu} = \frac{Q D_p}{\pi D_p^2 \frac{\nu}{4}} = \frac{4Q}{\pi D_p \nu} \]  

The equations of Haaland and Althsul plotted together in Figure 12. As it can be seen, the values of friction coefficients predicted by the two equations are quite close. For the sake of simplicity and computational efficiency, in the present thesis, it is preferred to use the equation of Altshul.

Inserting Eq. (25) in Eq (27), we get,

\[ f = 0.11 \left( 68 \frac{\pi D_p \nu}{4Q} + \varepsilon \right)^{0.25} = 0.11 \left( \frac{17\pi D_p \nu}{Q} + \varepsilon \right)^{0.25} \]  

By inserting Eq. (28) in Eq. (23), one obtains the expression for the friction head loss

\[ h_f = 0.11 \left( \frac{17\pi D_p \nu}{Q} + \varepsilon \right)^{0.25} \frac{L_p}{D_p} \frac{Q^2}{2g A_p^2} \]  

The subprogram “losses.m” uses Eq. (29) to calculate the head loss due to friction. The expression for the effective head is then obtained by inserting Eq. (29) in Eq. (22):
\[ H_{eff} = Z_u - Z_d - 0.11 \left( \frac{17\pi D_p \nu}{Q} + \varepsilon \right)^{0.25} \frac{L_p}{D_p} \frac{Q^2}{2gA_p^2} \]  

(30)

Figure 12 Friction coefficient graphs for Haaland (1983) and Altshul (1952).

By inserting Eq. (30) in Eq. (18), the expression for the hydropower generated by a turbine is obtained:

\[ HP = \eta \gamma Q \left( Z_u - Z_d - 0.11 \left( \frac{17\pi D \nu}{Q} + \varepsilon \right)^{0.25} \frac{L}{D} \frac{Q^2}{2gA^2} \right) \]  

(31)

In the model, following physical constants are defined:

- The total turbine efficiency is accepted to be approximately 0.918. It is assumed that there is constant efficiency in the turbines for each flow rate.
- The density of water is taken as 1000 \( kg/m^3 \).
• Viscosity of water is taken as $1.00 \times 10^{-6} \text{m}^2/\text{s}$.

• Gravitational acceleration is taken as $9.81 \text{ m/s}^2$.

The tailrace elevation $Z_d$ requires computation of the discharge rating curve for the cross section at the downstream of the dam. In the present thesis, however, it is assumed that the variation of the water surface elevation with the discharge is negligible, and the user will provide a constant value as input data.

3.5. Final Considerations for the Matlab Code with Genetic Algorithm

In Section 3.2, it was shown that the computation of end of the month storage volume computed using Eq. (15). The hydropower generation is computed using the pool elevation corresponding end of the month storage volume. In order to do that, the program needs the relationship between the reservoir pool elevation and the storage volume, which is normally available for a given reservoir as stage-volume curve. This point will be discussed later in more detail when presenting the test case data. In the Matlab code, subprogram “storage.m” calculates the end of the month storage volume using Eq. (15). This value is then used by the subprogram “pool_elevation.m” to calculate and return the corresponding pool elevation.

The genetic algorithm code requires at least one stopping criteria to stop the generation of new population and terminate the program. Since this is a multiobjective optimization, there can be more than one optimum solution. The fitness values of the population must be analyzed to select the best solutions.

In the program, this is achieved by the subprogram “lastdata.m”, which is called by the “MAIN.m” when the genetic algorithm encounters the stopping criteria. The selection of the best chromosome is achieved using a weighted sum of the index values for the two objective
functions and the total amount of storage volume released from the reservoir. The subprogram “lastdata.m” performs the following operations:

- For each chromosome in the population, the sum of the total energy produced over the \(m\)-year study period, \(TPE\), is calculated. Based on this total, an integer index value, \(I_{TE}\), is assigned to the chromosome. The integer index value corresponds to the quotient obtained by dividing the \(TPE\) by a suitable interval value, for example 100 GWh in the present case.

- For each chromosome in the population, the sum of the total firm produced over the \(m\)-year study period, \(TFE\), is calculated. Based on this total, an integer index value, \(I_{FE}\), is assigned to the chromosome. The integer index value corresponds to the quotient obtained by dividing the \(TFE\) by a suitable interval value, for example 5 GWh in the present case.

- For each chromosome in the population, the total overflow volume released to downstream over the \(m\)-year study period, \(VOF\), is calculated. Based on this total, an integer index value, \(I_{OF}\), is assigned to the chromosome. The integer index value corresponds to the quotient obtained by dividing the \(VOF\) by a suitable interval value, for example 5 hm\(^3\) in the present case.

- The best chromosome is then calculated using the following formula:

\[
Best\ Chromosome = TPE \times w_{TPE} + TFE \times w_{TFE} + VOF \times w_{VOF}\tag{32}
\]

where \(w_{TPE} = 0.50\), \(w_{TFE} = 0.25\), and \(w_{VOF} = 0.25\) are the weighting coefficients. Since, the present study focuses on the energy production; the weighting coefficient \(w_{TPE}\) is chosen to be greater than the other two. The weight coefficient values are highly subjective and problem dependent. They can be defined according to the order
of importance specified in the problem, which may be based on the opinion of subject
experts or the preferences of the decision makers

The Matlab genetic algorithm toolbox provides numerous options for custom tailoring
various options and the methods used for selection, crossover and mutation. The options and
method used in CODE01 and CODE02 are briefly described below.

- Population size is an important parameter, which decides number of individuals
  (chromosomes) in each generation. In general, a larger population size is better to
  find the best individuals in a feasible region. However, as the size of the
  population increases, the speed of the algorithm slows down. In the present thesis,
  the population size was set to 1,000.

- The selection is performed using the tournament method, which is a highly
  recommended method [20]. It is also possible to write one’s own code for the
  selection but this option was not chosen.

- The crossover fraction value specifies which percentage of children will be
  generated by cross over. In the present study, crossover fraction was chosen as
  0.8. This means that 80% of the children are created by crossover and 20% are
  created by mutation.

- The intermediate crossover method was used in the present study. This method
  uses the following formula to obtain the children

  \[
  \text{child} = \text{parent1} + \text{rand} \times \text{Ratio} \times (\text{parent2} - \text{parent1})
  \]

  (33)

  The new individual is created with parent1 and parent2, and the ratio is weighted
  average of the parents [29].
Given that the code is designed to solve a constrained nonlinear optimization problem, the adaptive feasible method is the only option available for mutation. In this method, the directions that are adaptive to the last successful and unsuccessful method are randomly selected with a step length that satisfies the bounds and linear constraints.

The Matlab uses the option Pareto fraction to control the number of individuals on the Pareto front as a function of the total population. In the present study, the default value of 0.35 is used. This means that the number of individuals on the Pareto front is equal to 35% of the total population.
4. RESERVOIR OPERATION USING RULE-BASED TRADITIONAL METHOD

(CODE03)

Reservoir operation is done by creating and implementing specific strategies. The amount of inflows which are rainfall, snowmelt and upstream reservoir releases and the amount of demand water are considered and an operation plan is created for the reservoir. This plan includes traditional strategies based on predetermined rules which mediate the amount of water used in the reservoir.

In this study, the reservoir volume used for the operation of the existing dam was divided into three main zones, as illustrated in Figure 13. The amount of firm energy and secondary energy that can be produced in a given month is decided based on in which zone the end of the month storage volume is located.

![Figure 13 Illustration of reservoir zones for the rule-based traditional method.](image)

It will be assumed that the maximum storage volume, $S_{max}$, minimum storage volume, $S_{min}$, and the storage volume to produce the targeted firm energy amount, $V_{firm}$, are given as
input data. Then, the allocation of storage volumes for firm energy and secondary energy are decided using the procedure described below.

First a temporary end of the month storage volume, $STEMP$, is computed using inflow and outflow storage volumes for that month, but without allocating any storage volume for energy generation. The allocation of the storage volumes for firm energy and secondary energy are then decided based on in which zone $STEMP$ is located.

a) $STEMP > S_{max}$: the reservoir pool elevation is in zone “a”

When $STEMP$ is in zone “a”, there are three possibilities:

a.1) $STEMP - S_{max} \geq SMAXTUR$

The allocation of storage volumes for this case is illustrated in Figure 14.

In this figure $STEMP$ is reservoir storage at the end of the month without energy water (hm$^3$). $S_{MAX}$ is the maximum reservoir operation storage (hm$^3$). $SMAXTUR$ is maximum turbine storage (hm$^3$). $SBEG$ is initial reservoir storage (hm$^3$). Initial reservoir storage can start at or below the maximum reservoir storage. Both firm energy and secondary energy are produced. Secondary energy is obtained by subtracting firm energy amount from the total monthly produced energy. All turbines are operated at maximum capacity during the month, and the amount of water that exceeds the turbine capacity is spilled. In this case, the amount of water remaining at the end of the month in the reservoir is equal to the maximum reservoir level.
a.2) \( VFIRM \leq STEMP - SMAX < SMAXTUR \)

The allocation of storage volumes for this case is illustrated in Figure 15. The amount of water to be spilled is zero. The targeted amount of firm energy is produced. Secondary energy is produced if storage volume remains above the maximum storage volume after the firm energy is generated. If there is not water for the secondary energy, it is preferred to keep the maximum reservoir at the maximum pool level. The end-of-month reservoir level equals the maximum reservoir level.

Figure 14  Illustration of storage volume allocation for the case “a.1”.

Figure 15  Illustration of storage volume allocation for the case “a.2”.
a.3) \( VFIRM > STEMPSM - SMAX \)

The allocation of storage volumes for this case is illustrated in Figure 16. Only firm energy is produced, which brings the pool level below the maximum reservoir level. The end of the month reservoir volume is below the maximum reservoir volume. There is no overflow from the spillway.

\[
STEMPSM > SMAX
\]

![Figure 16](image)

Illustration of storage volume allocation for the case “a.3”.

b) \( SMIN + VFIRM/2 < STEMPSM - SMIN \leq SMAX \): the reservoir pool elevation is in zone “b”

When STEMPSM is in zone “b”, the overflow volume is always null and the secondary energy is never produced. The two cases for the production of firm energy are described below.

b.1) \( STEMPSM - SMIN \geq VFIRM \)

The allocation of storage volumes for this case is illustrated in Figure 17. The total amount of energy produced is equal to the firm energy. The end-of-month reservoir level is between SMIN and SMAX.
Figure 17  Illustration of storage volume allocation for the case “b.1”.

b.2) \( STEMP - SMIN < VFIRM \)

In this case, the end of the month equals the reservoir level \( SMIN \). The full amount of firm energy cannot be produced, but at least half of the firm energy can be produced. The limit \( (SMIN + VFIRM/2) \) helps to ensure that the turbines are not operated for very low flow rates.

c) \( SMIN + VFIRM/2 > STEMP - SMIN \): the reservoir pool elevation is in zone “c”

In this case, no energy is produced and the volume \( STEP \) is stored to increase the storage volume in the reservoir.

Once the firm energy and secondary energy volumes are assigned, if any, the program uses predefined rules to determine how the turbines will be used to produce the energy. These rules are problem specific and will be discussed in the application to the test case.
5. CASE STUDY: KAYRAKTEPE DAM IN TURKEY

Kayraktepe Dam and reservoir system was selected as case study for this thesis. This section describes the characteristics of the system and the input data available for the optimization study.

The Kayraktepe Dam and Hydropower plants project is located on the Goksu River in the Mediterranean Region of Turkey. The Goksu River is born in the Taurus Mountains. Its northern branch arises in the Geyik Mountain and the southern branch in the Haydar Mountain. The two branches join to form Goksu River near southern Mut district. The Ermenek stream joins the Goksu River from the right side just upstream of the Kayraktepe Dam. The Goksu River water released from Kayraktepe Dam flows into the Mediterranean Sea.

Figure 18 Location of dams and their operating status Red: under construction (Bagbasi, Bozkir, Ermenek), Yellow: in the project development phase (Mut, Kayraktepe), Green: already operating dam (Gezende).
There are several dams upstream of the Kayraktepe Dam project. Their location is shown in the map reproduced in Figure 18. One dam is in project development phase (Mut Dam), three are currently under construction (Bagbasi Dam, Bozkir Dam, and Ermenek Dam), and one dam is already operating (Gezende Dam).

![Figure 19 Monthly irrigation volume needs of the agricultural sector.](image)

The Kayraktepe Dam is part of the Mediterranean-region development project. It is designed as a multipurpose facility. It will provide the drinking water for the Mersin province located downstream of the dam. The flow rate of drinking water planned to be supplied to the city is 5 m$^3$/s. Kayraktepe Dam is located in a highly productive agricultural region. The volume of irrigation water needed for the agricultural sector is plotted in Figure 19. One of the purposes of the Kayraktepe project is to provide agricultural irrigation, which expected to contribute to the economic development of the region. Irrigation will be operated with a pressurized pipe system. Nine pumps will be used for the pumping operation.

Kayraktepe is also designed to generate hydropower energy. It is planned to equip the hydropower plant with four vertical axis Francis turbines. Two larger turbines have a nominal flow rate of 140 m$^3$/s whereas the two smaller turbines have a nominal flow rate of 35m$^3$/s. The installed capacities for the large and small turbines are 113.02 and 27.86 MW, respectively. The
turbines are supplied by 90m-long penstocks. The penstock diameter is 5.40 m for large turbines and 2.90 m for small turbines. The main characteristics of the Kayraktepe Dam Project are summarized in Table 2.

Table 2 Characteristics of the Kayraktepe Dam Project.

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drainage Area</td>
<td>10,065.02 km²</td>
</tr>
<tr>
<td>Annual Average Water</td>
<td>26,640 hm³</td>
</tr>
<tr>
<td>Purposes</td>
<td>Irrigation, Flood Control, Drinking Water</td>
</tr>
<tr>
<td>Type of Dam</td>
<td>Roller Compacted Concrete</td>
</tr>
<tr>
<td>Thalweg Elevation</td>
<td>25.00 m a.s.l.</td>
</tr>
<tr>
<td>Minimum Operating Elevation</td>
<td>90.00 m a.s.l.</td>
</tr>
<tr>
<td>Maximum Operating Elevation</td>
<td>120.00 m a.s.l.</td>
</tr>
<tr>
<td>Flood Control Level</td>
<td>124.00 m a.s.l.</td>
</tr>
<tr>
<td>Elevation of Crest</td>
<td>125.00 m a.s.l.</td>
</tr>
<tr>
<td>Minimum Storage Volume (at 90 m elevation)</td>
<td>423.79 hm³</td>
</tr>
<tr>
<td>Active Volume Capacity</td>
<td>1,155.69 hm³</td>
</tr>
<tr>
<td>Total Storage Volume</td>
<td>1,579.48 hm³</td>
</tr>
<tr>
<td>Lake Site (at 120 m elevation)</td>
<td>59.29 km²</td>
</tr>
<tr>
<td>Thalweg Height</td>
<td>100.00 m</td>
</tr>
<tr>
<td>Width Crest</td>
<td>10.00 m</td>
</tr>
<tr>
<td>Irrigation Area</td>
<td>7,425 ha (gross)</td>
</tr>
<tr>
<td>Irrigation Type</td>
<td>Pressure Pipe</td>
</tr>
<tr>
<td>Tailrace canal elevation (constant)</td>
<td>28.3 m a.s.l.</td>
</tr>
</tbody>
</table>
The aim of the test-case application is to optimize the total energy generation and firm energy generation while minimizing the releases to the downstream through the overflow spillway. Maximization of energy production and minimization of spillway releases are conflicting goals.

The optimization problem requires inflow data into the reservoir over a sufficiently long period of time. There are several hydro meteorological gaging stations in the Mediterranean region where the Kayraktepe Dam is located. Some of these observation stations are under the responsibility of the General Directorate of Meteorology (MGM), some of them under the responsibility of the General Directorate of State Hydraulic Works (DSI). Figure 20 shows the locations of hydro meteorological stations in the region. It is important to note that the stations marked as a red circle are inactive.

![Figure 20](image)

Figure 20  The stream gaging and meteorological gaging stations.

The monthly precipitation rates, monthly average temperatures and monthly evaporation volumes are plotted in Figure 21, Figure 22, and Figure 23 based on the data from the hydro meteorological gages.
Figure 21 Monthly precipitation rates.

Figure 22 Monthly average temperatures.

Figure 23 Monthly evaporation volumes.
The total inflow volume into the reservoir comes from rainfall, snowmelt, and the releases from the dams located upstream. The time series of monthly inflow volumes entering into the reservoir are plotted in Figure 24. The time series extends from October 1984 to September 2013 and has a length of 30 years (360 months). As it can be seen, between 2007 and 2009 there is relatively long period with significantly small discharges. This period could prove be a critical period for the optimization of the reservoir operations.

![Figure 24 Time series of monthly inflow volume into the Kayraktepe Reservoir.](image)

The targeted storage allocation for the firm discharge can be estimated from the duration curve for monthly potential energy volume. Potential energy volume is obtained by subtracting the sum of monthly evaporation and irrigation volumes from the monthly inflow volume

\[
PEV = I - EV - IR
\]

(34)

The PEV values are calculated for all months in the record. The resulting time series are plotted in Figure 25 together with the monthly inflow volumes \(I\). Since the monthly evaporation and irrigation volumes are generally very small compared to the inflow discharge, the two curves show very little difference.

The 30-year observation period includes \(N = 360\) monthly potential energy volumes. The PEV values are sorted from the largest to the smallest value. The values in the sorted list are assigned a rank value. The smallest value has a rank of \(R=1\) and the largest inflow value has a
rank of R=N. Then, the following formula is used to assign an exceedance probability to each PEV value using the following formula:

\[ p = \left( \frac{R}{N} \right) \times 100 \]  

where \( p \) is exceedance probability in percent, which also corresponds to the percent time during which a given discharge is equaled or exceeded.

![Monthly Inflow and Potential Energy Volumes](image)

Figure 25  Time series of monthly inflow (\( I \)) and potential energy (\( PEV \)) volumes.

The duration curve for the potential energy volume (\( PEV \)) for Kayraktepe Project is then obtained by plotting the PEV values as a function of the probability \( p \), as shown in Figure 26. For a given potential energy volume \( PEV \) on the vertical axis, the value read on horizontal axis from the curve gives the probability that the PEV value will be equaled or exceeded, or also the percentage of the time that the monthly potential energy volume will equal or exceed the value \( PEV \).

The targeted firm energy amount is decided based on the duration curve for monthly potential energy volume. Generally, the targeted firm energy is based on a monthly potential
energy volume with a probability in the interval 90% to 97%. In this thesis, the targeted firm energy volume is chosen to be 155.14 hm$^3$, which corresponds to a probability of 95.6%.

![Duration Curve for Monthly Potential Energy Volume](image)

**Figure 26** Duration curve for monthly potential energy volume (PEV).

The data provided by the DSI includes also the storage volume and lake surface area for the reservoir impounded by the Kayraktepe Dam. The storage volume and lake surface area for the Kayraktepe Reservoir are plotted together in Figure 27. The following third order polynomial was fitted to the useful portion of the elevation-volume curve highlighted in yellow.

$$Z_u = 4.062 \times 10^{-9} \times S^3 - 2.339 \times 10^{-5} \times S^2 + 5.91 \times 10^{-2} \times S + 6.89 \times 10^{-1}$$ (36)

where $Z_u$ is the elevation (m a.s.l.) and $S$ is the storage volume in the reservoir at elevation $Z_u$. This polynomial was programmed into the subprogram “Pool_Elevation.m” in order to obtain the elevation for a storage volume provided as the argument.
Figure 27  Elevation-volume and elevation area curves for Kayraktepe Reservoir.
6. DISCUSSION OF RESULTS AND CONCLUSIONS

The test case of Kayraktepe Dam was simulated using the three Matlab codes developed in this thesis research:

1. CODE01, which implements the multiobjective genetic algorithm without considering the operation of individual turbines.
2. CODE02, which implements the multiobjective genetic algorithm by considering the operation of individual turbines.
3. CODE03, which implements the rule-based traditional method to calculate the amount of firm energy and total energy that can be produced.

The Kayraktepe Dam is in the project development phase. The water needs of the region where the Kayraktepe dam is located and the data taken from observation stations before the dam’s construction are determined for 30 years. The planning of the dam to test the project efficiency was made considering these 30 years. This thesis aims to show an increase in total and firm energy production gained by using genetic algorithm compared to the traditional method and DSI operation results.

The schematic illustration of the Kayraktepe Project is shown in Figure 28. The reservoir impounded by the dam receives a monthly volume of inflow produced by runoff, due to rainfall and snowmelt, and the releases from the upstream dams. Every month, a small part of the storage volume is lost to the evaporation (Figure 23). Moreover, part of the storage volume is allocated to satisfy the monthly irrigation volume needed by the agriculture sector (Figure 19).
The targeted firm energy amount for the Kayraktepe Project was chosen to be 155.14 hm³. Assuming that the firm energy is generated 24 hour per day during every day of a month having an average duration of 30.5 days, this volume corresponds to a turbine flow rate of 58.87 m³/s, which is 1.68 times the nominal flow rate of the small turbines. The report provided by the DSI specifies that the smaller turbines should be preferred for generating firm energy and the larger turbines should be used for secondary energy and peak demands. Therefore, in CODE02 and CODE03, it will be assumed that the firm energy is generated by the two turbines, one at full capacity and the other at partial capacity. Any excess volume available for the secondary energy will be used by first the smaller turbine working at partial capacity and then the two larger turbines. The total flow rate of all turbines combined is 350 m³/s.

Figure 28  Modeling of the Kayraktepe Project.
The maximum and minimum reservoir operation levels for the Kayraktepe Dam are given as 120 m a.s.l. and 90 m a.s.l., which correspond to storage volumes of 1579.48 hm$^3$ and 423.79 hm$^3$ (see Table 2).

If the end of the month storage volume is greater than 1579.48 hm$^3$, i.e. if the reservoir elevation is greater than 120 m a.s.l., the excess volume needs to be released as overflow from the spillway. The optimal reservoir operation strategy should release as little overflow as possible.

The calculation of the effective head requires the pool elevation in the reservoir and the tailrace canal elevation. The third order polynomial fitted to the elevation volume curve is implemented in the subprogram “Pool_Elevation.m”. For a given volume, it provides the corresponding pool elevation value. For the Kayraktepe Project, the elevation in the tailrace canal is assumed to have a constant value of 28.3 m a.s.l. (see Table 2) regardless of the energy water released from the turbines.

The application of the Matlab codes assumes that the reservoir should be operated at the maximum level of 120 m a.s.l. as much as it is possible. It is also assumed that, the reservoir level cannot go below the minimum elevation of 90.0 m a.s.l.

In this section, the results obtained from the energy optimization using the genetic algorithm (CODE01 and CODE02) are compared with the results obtained by the rule-based traditional method (CODE03) and the operational data computed by the engineering firm hired by the DSI.

6.1. Comparison of Results of CODE01 with Operational Data from the DSI
In this subsection, the results of optimization obtained with CODE01, which does not consider the operation of individual turbines, are compared with those computed by the engineering company hired by the DSI.

Figure 29 compares end of the month storage volumes obtained from optimization using CODE01 with the results provided by the DSI. During the first 161 months approximately the storage remains at its targeted maximum value for the results with CODE01 whereas the results provided by DSI show small deviations from the targeted maximum storage. The differences, however, are relatively small since the inflow volumes are sufficiently high during this period. After the 161th month, the results from DSI begin showing large deviations from the targeted maximum storage. The reservoir volume sharply drops after day 276 and reaches its minimum value around the day 290. It continues to operate at this minimum storage value almost until the end of the study period. The results obtained with CODE01 show significantly smaller deviations from the targeted maximum storage. The very last value of the storage volume obtained with CODE01 also shows a significant drop at the very end of the study period. However, this is an artificial effect due to the fact that there is no inflow data available beyond the month 360 and the optimization program is trying to generate as much energy as possible.
Figure 29 Comparison of end of the month storage volumes obtained from optimization using CODE01 with the results provided by the DSI.

Figure 30 compares the monthly total energy production obtained from optimization using CODE01 with the results provided by the DSI. For the first 161 months the two plots show similar monthly total energy production, since the inflows are sufficiently high. Between the months 161 to 298, the results provided by DSI show a steady monthly total energy production slightly lower than 50 GWh. Although DSI has a lower reservoir level, there are places that generate more energy, e.g. month 271. The reason is that genetic algorithm always considers subsequent months, so if there is not enough water to produce more energy, genetic algorithm prefers to store water volume for the next month. The results obtained with CODE01 fluctuate more but the total energy production is greater. This is due to the fact that, the genetic algorithm converges to a strategy that prefers to keep storage level constant. During low flow periods, after providing the firm energy, lesser amount of volume is used to generate secondary energy to increase the storage volume. This yields a higher storage volume for the subsequent months, which in turn results in higher annual total energy production which is clearly shown in Figure 32.
Figure 30  Comparison of end of the monthly energy production obtained from optimization using CODE01 with the results provided by the DSI.

Figure 31 shows the tradeoff between the two objective functions for CODE01. Objective function 1 represents the maximization of total energy generation during the 30 years, and objective function two express the minimization total firm energy deficiency during the 30 years. In Matlab, maximization is expressed in negative values and minimization in positives. Therefore, the figure has negative values on horizontal axis and positive values on the vertical axis. The results obtained for objective function one are between $1.722 \times 10^4$ GWh and $1.718 \times 10^4$ GWh. These values are very close to each other, indicating that the chromosomes have very similar total energy generation for entire period. The total firm energy deficiency is between 160 GWh and 360 GWh for 30 years. In this graph, the chromosome with the most energy production and the least deficiency will be selected as the best solution.
Figure 31 Pareto front obtained using Matlab for CODE01

Figure 32 compares the annual total energy production obtained from optimization using CODE01 with the results provided by the DSI for the entire study period of 30 years. This curve shows a clear decreasing trend due to the fact that the inflows are decreasing throughout the 30-year study period. The difference between the results from DSI and the results using CODE01 become significant after the year 25. DSI is producing more power than CODE01, even though CODE01’s storage is higher at year 18. Because CODE01 aims to keep reservoir storage at the maximum targeted reservoir storage by allocating less volume for energy production if inflow is not sufficient, in this case, CODE01 produces less energy and stores water volume. The annual total energy productions provided by the DSI continue to further decrease while the results with CODE01, maintain an almost constant value. The sudden increase at the end of the series is probably due to the effect mentioned in while discussing the end of the month storage volumes plotted in Figure 29.
The annual average energy production over 30 years is 575.44 GWh with CODE01, and 539.66 GWh based on the results provided by the DSI. Thus, the operation of the reservoir according to the optimal strategy provided by CODE01 gives a 35.78 GWh higher annual average energy production, which is significant. Considering the selling price of the energy and the number of years of operation, the difference can be measured in tens of millions of dollars of additional income.

![Annual Energy Production (GWh)](image)

Figure 32  Comparison of the annual total energy production obtained from optimization using CODE01 with the results provided by the DSI.

Figure 33 compares the annual firm energy production obtained from optimization using CODE01 with the results provided by the DSI for the entire study period of 30 years. One of the two objective functions of CODE01 aims to keep the firm energy production equal to or higher than the targeted firm energy, by minimizing the firm energy deficiency. The plot in Figure 33 shows that CODE01 is able to provide a higher firm energy production, especially after the year 25, despite the significant decrease in inflow volumes. Between the years 11-16, DSI is producing more firm energy compared to CODE01, even though CODE01’s reservoir level is
higher. CODE01 limits energy production because of the inflows deficiency between these years. CODE01 always try to keep reservoir storage at the maximum targeted storage volume.

![Figure 33](image)

Figure 33 Comparison of the annual firm energy production obtained from optimization using CODE01 with the results provided by the DSI.

6.2. Comparison of Results of CODE02 with CODE03

In this subsection, the results of optimization obtained with CODE02, which considers the operation of individual turbines, are compared with those computed using CODE03 that implements a rule-based traditional method.

Figure 34 compares end of the month storage volumes obtained with CODE02 and CODE03. Both CODE02 and CODE03 can keep the reservoir volume at its targeted maximum value. Both Matlab codes yield quite similar results except during the sudden drop in the month 298, which corresponds to the year 2008. It should be noted that the year 2008 was already highlighted as a critical period in Chapter 4. During this period, CODE03 is able to keep the reservoir at a slightly higher storage volume. However, the influence of this difference on the annual energy production is very small. This may be due to the fact that the rules with regard to
the operation of the turbines are the same and act as an additional nonlinear rule-based constraint.

Figure 34  Comparison of end of the month storage volumes obtained with CODE02 (optimization by considering individual turbines) and CODE03 (rule-based traditional method).

Figure 35 compares monthly total energy productions obtained with CODE02 and CODE03. The monthly total energy productions computed by both codes are very similar.

Figure 35  Comparison of monthly energy production obtained with CODE02 (optimization by considering individual turbines) and CODE03 (rule-based traditional method).
Figure 36 shows the tradeoff between the two objective functions for CODE02. Objective function 1 represents total energy production and 2 represents total firm energy deficiency during the 30 years. Total energy production in the last population varies between $1.5925 \times 10^4$ GWh and $1.5922 \times 10^4$ GWh. The total firm energy deficiency values on the vertical axis are quite small. This indicates that the targeted firm energy value is almost guaranteed for all chromosomes. The chromosome chosen as the best solution was the one which has the highest total energy production and the least total firm energy deficiency.

The annual total and firm energy productions computed using CODE02 and CODE03 are compared in Figure 37 and Figure 38, respectively. As it can be seen, in both plots, the values calculated by the two codes are very similar to one another. Nevertheless, the average annual energy production computed by CODE02 is 1.2% higher than that produced by CODE03. CODE02 yields an annual average energy production of 521.17 GWH whereas CODE03 yields 514.75 GWh. Since the rules implemented in CODE03 forbids generation of secondary energy
when the reservoir volume is less that the maximum reservoir volume, the annual total energy plotted in Figure 37 shows more fluctuations. Note that the annual firm energy plotted in Figure 38 represents the minimum total energy produced during the year. This value may be greater than the targeted form energy.

Figure 37  Comparison of the annual total energy production obtained with CODE02 (optimization by considering individual turbines) and CODE03 (rule-based traditional method)

Figure 38  Comparison of the annual firm energy production obtained with CODE02 (optimization by considering individual turbines) and CODE03 (rule-based traditional method).
Different assumptions were used in programs written for the genetic algorithm (CODE01 and CODE02). The effect of these assumptions on the program results is the following:

- Turbine operation is the most important factor that has an impact on the results. The rules used for turbine operation act like nonlinear constraint in the program. This situation led us to obtain different amounts of energy generation from the CODE01 and CODE02 programs. Optimization by considering individual turbines reduced annual average energy production by 54.33 GWh.

- Another difference between CODE01 and CODE02 is the use of true head loss. The head loss was assumed to be constant (2.2 meters) for the CODE01 program. This is the accepted value in DSI calculations. However, CODE02 considered true friction head losses. These losses vary from 0.5 to 1 m depending on the energy water volume during the month. This contributed to the increase in energy production.

All other values and assumptions used in the program are considered to be the same in both CODE01 and CODE02 programs.

6.3. Sensitivity of the Optimal Results to the Parameters of Genetic Algorithm

In Matlab, sensitivity analysis is performed with the aim of testing the adequacy of the options used. These options relate to the stopping criteria and initial population which are defined in Chapter 1.

The options defined for the genetic algorithm are tested as follows: the maximum generation number, which means the number of iterations to reach the last population, is defined as 100 *nvars in the Matlab program. We tested and checked maximum generation variables in
the program with 3000, 5000, and 10, 000. This increase for the maximum number of
generations did not cause any changes in the results. This means that the value defined for the
maximum generation in this project, which is 3000, is suitable.

Secondly, function tolerance variables were checked. Starting with $10^{-1}$, the function
tolerance was reduced by 10 and tested for different values up to $10^{-6}$. The monthly distribution
of the energy production in the last population was not satisfactory for $10^{-1}$, $10^{-2}$ and $10^{-3}$.
However, the results obtained for the other defined function tolerance values were almost
identical. This indicates that the optimum population was reached.

Another option is population size. As the number of chromosomes increases, it provides
diversity in the population obtained. Population size was defined and tested as 1000 and 2000.
Because the increase in population did not affect the results, we used 1000 for population size.

In addition to all of these, the test of the initial population was made as follows. The
volumes of energy water obtained from the traditional method were placed in the initial
population of the genetic algorithm. It is seen that Matlab creates successful initial population for
the optimization problem.

6.4. Conclusion

The present thesis considered the problem of multiobjective optimization of hydroelectric
energy production in a multipurpose reservoir providing irrigation water for the agricultural
sector and potable water supply for municipalities.

The optimization of energy production was formulated by considering two objective
functions. One objective function aimed at maximizing the total hydroelectric energy production
and the second objective function aimed at minimizing the firm energy deficiency over a number
of operational years.
Two different Matlab codes were written for solving the formulated multiobjective optimization problem using the genetic algorithm solver in the Global Optimization Toolbox of Matlab. The Matlab code CODE01 did not consider the characteristics of individual turbines. The net head was calculated simply by subtracting a constant average head loss from the difference between the end of the month reservoir elevation and tail water elevation. The Matlab code CODE02, on the other hand, considered the characteristics of individual turbines and how the storage volume allocated for energy production will be distributed between multiple turbines by taking into account the preferences of use of turbines for firm energy and secondary energy production. Moreover, for each turbine the true friction loss was calculated based on the turbiined flow rate. The net head available for energy production was then calculated by subtracting the true friction head loss from the difference between the end of the month reservoir elevation and tail water elevation. It is important to note that the optimization problem considered in the present thesis refer to a deterministic optimization based on the known inflow and outflow volumes from the dam. This type of problem is generally encountered in planning and development studies and help to determine the economic feasibility of the planned multipurpose dam-reservoir system.

The case of Kayraktepe Dam and Reservoir located south of Turkey was considered as a test case. The State Water Works of Turkey (DSI) has kindly provided the data, which consisted of reports of planning studies giving information about the facility and its operation, as well as monthly volumes of inflows into the reservoir, monthly volumes of evaporation losses from the reservoir, and monthly volumes of various outflows (irrigation, environmental water, drinking water, etc.). The data provided by the DSI included also a spreadsheet, which calculated the
energy production using the traditional rule-based method to allocated storage volumes for firm energy and secondary hydroelectric energy generation.

In order to be able to make a better comparison with the spreadsheet provided by the DSI, a third Matlab code, called CODE03, was also written by programming a different version of the rule based traditional approach. Although, it did not include any optimization, CODE03 also adopted the approach used in CODE02 for calculating the energy production by considering the characteristics of individual turbines.

The results of CODE01 were compared with the results in the spreadsheet calculations provided by the DSI. It is important to note here that this comparison is not an easy task given the fact that the operational policies for energy production are not exactly the same although both approaches do not consider the energy production by individual turbines and calculate the energy production using a constant friction head loss regardless of the flow rate. CODE01 uses GA optimization whereas the spreadsheet uses a rather simple rule based approach, which aims to produce as much energy as possible by allocating the inflow and storage volume to energy production. Thus, the comparison is only possible in a loose and qualitative manner. Therefore, the following conclusions must be interpreted taking into account this fact:

- One of the objectives of the reservoir operation implicitly included in the operation rule-set is to keep the reservoir storage at the maximum storage level. Referring to Figure 29, the rule-based strategy adopted in CODE01 achieves this goal better by operating the reservoir near maximum storage level. Only during a few draught events (e.g. month 298), the reservoir level deviates slightly from the maximum storage level. Comparatively, the rule set adopted in the spreadsheet attempts to produce more energy without any consideration of the
future inflows. This results in a serious drop of the reservoir storage level starting from month 280. The reservoir operates near minimum level from month 294 to the end.

- The monthly total energy production calculated by CODE01 fluctuates more but especially starting from month 294 to the end, the monthly energy production is higher than that computed by the spreadsheet.

- The pareto front presented in Figure 31 shows that the objective 1 related to maximization of total energy produced over the period of 30 years has a very narrow range. The difference between the two ends of the axis is only 70GWh, which is not significant. The variation for the Objective 2 concerning the maximization of annual firm energy for individual years has a much larger range. This simplifies the selection of the best solution among those in the Pareto front.

- The annual average energy production calculated by the CODE01 is 35.78 GWh higher than that given by the spreadsheet provided by the DSI. Considering the selling price of energy and the number of years of operation this may translate into considerable additional income. During the first 15 years, the inflows are larger and both methods calculate about the same amount of total energy production for the 30 years. After the year 15, differences are observed as the flow discharges become smaller. The significant differences occur after the year 25, during lower inflows. Since it can keep the reservoir storage level closer to the maximum storage level, CODE01, calculates higher annual total energy production, which leads to a higher total energy production of the 30 years. Thus
the strength of CODE01 resides in its better management of the reservoir during low flow periods, which stresses reservoir operations.

- Referring Figure 33, between years 10 to 20, CODE01 seems to produce a lower firm energy. However, this is due to the definitions of firm energy between the CODE01 and the spreadsheet. The minimum annual energy from CODE01, although lower than the firm energy computed by the spreadsheet, meets the minimum firm energy requirement corresponding to the target volume of 155.14 hm$^3$. The strategy chosen by CODE01, however, pays after the year 24. From year 24 to 30, the discharges are low and the reservoir operation is significantly stressed. During this period, CODE01, maintains a much higher energy production rate whereas the spreadsheet solution shows a huge drop in the firm energy. In year 25, it cannot even meet the firm energy demand.

The comparison of the results of CODE02 and CODE03 is more meaningful as these two codes use much similar operational policies and rule-set.

- Referring to Figure 34, both codes keep the reservoir storage level close to maximum value.

- Referring to Figure 35, the monthly total energy productions are very close to each other.

- Pareto front calculated using CODE02 is presented in Figure 36. The range of objective 1 related to maximization of total energy produced over the period of 30 years has a very narrow range. The difference between the two ends of the axis is only 4 GWh, which is not significant. The variation for the Objective 2
concerning the maximization of annual firm energy for individual years is 0.12 GWh, which is also rather small.

- Referring to Figure 37, the annual energy production over 30 years obtained with CODE02 and CODE03 are very similar. Nevertheless, the total average energy production computed by CODE02 over a period of 30 years is 6.42 GWh higher than that calculated by CODE03. Although the difference is small, it may still correspond the several millions dollars of additional income. The small difference in total energy production between CODE02 and CODE03 indicates that the policies and rule sets defined for the operation of the reservoir are strongly constraining the feasible space.

- Referring to Figure 38, the firm energy production computed by CODE02 and CODE03 are quite similar although CODE03 provides a much smoother curve compared to CODE03.

These results show that the use of genetic algorithm in deterministic multiobjective optimization of reservoir operation for maximizing the energy production was successful. The results also show that the choice of operational policies and rule sets are extremely important as they may strongly constrain the feasible space. The present study omitted various details due to lack of information, such as the efficiency of the turbines based on turbine flow rate. These can be easily implemented in CODE02, which is the most complete optimization code that considers the operation of the individual turbines.
In this thesis, energy optimization was carried out using genetic algorithms to improve both the results of DSI operation and traditional operation method. The optimization using CODE, in addition to all these, the following recommendations can be used in future work to improve our model and achieve more realistic results:

- In this thesis, a constant total turbine efficiency corresponding to the maximum theoretical turbine efficiency was used. This is not realistic. The efficiency of the turbines varies according to the turbine discharge. A typical efficiency-flow curve for a Francis turbine is shown in Figure 39. Peak efficiency occurs between 80% and 95% flow for Francis turbine. Efficiency is 0 when flow falls below 8% [17]. Therefore, the turbine efficiency in the energy equation should depend on the discharge function as shown in the following equation.

\[ P_i = [\eta_i(QS_i)] \gamma H_{net_i} QS_i + [\eta_i(QF_i)] \gamma H_{net_i} QF_i \]  

where \( P \) is hydroelectric power produced (Watt), \( H_{net} \) is the net head available for turbine (m), \( QS_i \) is discharge for secondary energy (m\(^3\)/s), \( QF_i \) is discharge for firm energy (m\(^3\)/s), \( \eta \) is overall turbine efficiency, \( \gamma \) is specific weight of water (N/m\(^3\)). In order to implement the consideration of the turbine efficiencies, it would be necessary to have the data on the turbines. The consideration of the turbine efficiencies would be even more important in case of stochastic optimization using a daily time step
The multiobjective optimization process used observation data available for 30 years. This is a deterministic optimization suitable for development and feasibility types studies, but it cannot be use for real time operation of the reservoir. The operational optimization of the reservoir can be achieved based on meteorological forecasts. This would lead to a stochastic multiobjective optimization. Such a stochastic optimization cannot be accomplished using monthly flows. It is necessary to work at least with daily flows. It may be unnecessary to couple the optimization program with a watershed simulation module if it is intended to use rainfall forecasts.

The last recommendation is related to turbine operation. In our thesis, we operated our turbine according to specific rules. For firm energy generation we used two small turbines. For secondary energy and peak flows we preferred two large turbines. This rather a simplistic approach. If the characteristics of the turbines are known and if the time step is daily, better optimization of turbines would make a difference. For these types of problems, the optimization problem can be formulated to include an inner optimization to optimization the use of individual turbines.
REFERENCES


doi:10.3103/s0967091213040049


VITA

SEYMA SEHRIBAN TIRYAKIOGLU

Education:

Bachelor of Civil Engineering, Karadeniz Technical University (June 2014), Turkey.


Academic awards and honors:

Honor Student, Department of Civil Engineering, Karadeniz Technical University, 2010-2014.

Scholarship Award from DSI, 2014-2018