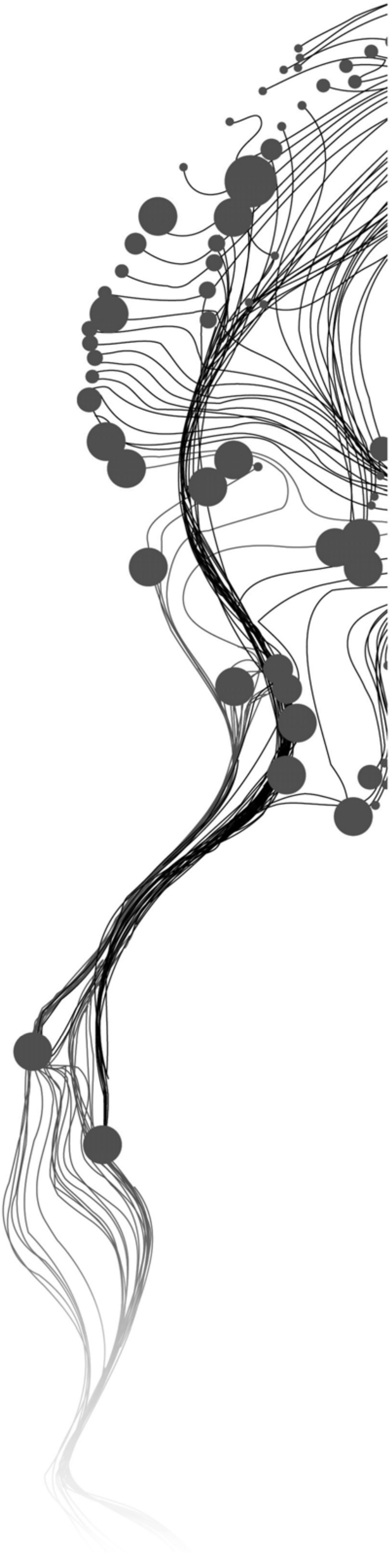
LOCATION ALLOCATION PROBLEM USING GENETIC ALGORITHM AND SIMULATED ANNEALING: A CASE STUDY BASED ON SCHOOL IN ENSCHEDE

MD. SHAMSUL ARIFIN
February, 2011

SUPERVISORS:
Dr. Raul Zurita-Milla
Dr. Otto Huisman



LOCATION ALLOCATION PROBLEM USING GENETIC ALGORITHM AND SIMULATED ANNEALING: A CASE STUDY BASED ON SCHOOL IN ENSCHEDE

MD. SHAMSUL ARIFIN
Enschede, The Netherlands, February, 2010

Thesis submitted to the Faculty of Geo-Information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.
Specialization: Geoinformatics

SUPERVISORS:
Dr. Raul Zurita-Milla
Dr. Otto Huisman

THESIS ASSESSMENT BOARD:
Professor Dr. Menno-Jan Kraak, Chair
Dr. Ir. Sytze de Bruin, External Examiner, Wageningen University

ABSTRACT

Location allocation is a combinatorial optimization problem. Traditional exact method cannot solve location allocation problem efficiently. This problem does not limit itself in a small spectrum rather it has been grown with lots of branches for around a century. Capacity, cost, different type facilities, demands, time, different type of distances and mixing with diverse real world problems have made location allocation problem much complex. Metaheuristic solutions like genetic algorithm and simulated annealing have been exploited for long time to solve location allocation problem. In several researches of location allocation problem, these two algorithms have been bought into one umbrella and have been proved efficient. But location allocation problem with integrated GIS, genetic algorithm and simulated annealing was not much explored. This research has explored location allocation problem by both genetic algorithm and simulated annealing with GIS integration. To achieve this, two case studies based on Enschede schools have been performed. Location allocation problem usually considers nearest distance. Through these case studies, location allocation problem also considers nearest distance with various criteria like capacity, user preference, existing facility etc.

ACKNOWLEDGEMENTS

I would like to thank both of my supervisors for their support and guidance all through this work.
I would like to thank my parents and family for their blessings.
I would like to thank almighty Allah.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

1. INTRODUCTION

1.1. Motivation and Problem statement

Locating a facility into the best place is a decision making problem. The best place depends on criteria like the optimal distance, the capacity of the facility, population density, optimal cost etc. Location allocation can be based on one criterion like optimal distance or adding various combinations of criteria like optimal distance and capacity of the facility together or capacity of the facility or optimal cost together and so on. So, the goal of the location allocation problem's solution is to find the best location or locations to fit one or more facilities which will make the highest utility value from one criterion or multiple criteria.

Bad location of the facility has negative effect to provide services to the beneficiary. Distance from the area of supply and the area of demand should be optimal. If location of the facility is far from populated area (area of demand) beneficiary may not be able or interested to take the service from that facility. This type of facility can be school, hospital, market, hospital, fire service etc. The capacity of the facility also has effect to provide the service. When facilities are created to meet the demand of people, capacity of the facility cannot be ignored. Therefore, the location of the facility should be well distributed such that capacity of the facilities can meet all the demands. So with optimal distance, capacity of the facility needs to be considered in the time of taking decision.

Solution of location problem is important for the decision makers. They need decision support tool which will locate the facility based on several criteria. Type of facility, user preference on facility, different services of the facility, facility opening closing time, facility establish and relocation cost criteria can also be significant for decision maker to take decision. Location allocation problem now-a-days not only sets facility in nearest distance but also it tries to add these non distance based criteria to find optimal solution.

Commercial ARCGIS software has also implemented location allocation problem in the extension of network analyst [1]. It is tightly coupled with analyst and has strong visualization that shows the output result. But this commercial software only deals with single objective location allocation problems which minimizes either total distance or time. For example, finding the locations that can reduce the overall transportation costs of delivering goods to outlet is one single objective type problem. Another single objective type is to find the maximum coverage from the location of police station, fire station, emergency rescue center etc. Their black box implementation doesn't allow any customization and development of inside these problems. ARCGIS itself cannot deal complex objective function with multi criteria like [2] or any non distance based criteria.

Location allocation is a combinatorial optimization problem. If optimal solution needs millions of combination then traditional exact method needs very high computational time to find the optimal one. Openshaw mentioned that applying deterministic method is not feasible because of its extreme computational time to solve this problem in the study [3]. Moreover, various classifications of location allocation problem do not keep the problem in an arena of a simple one. So, even if traditional deterministic methods try simple combinatorial optimization, it cannot deal complex location allocation problems with various criteria. So, methodologies which will provide optimal solutions based on one or more criteria and will not be trapped in local optima are needed. These solutions are metaheuristic.

Church and Murray [4] mentioned that commercial application softwares were not using much varieties of metaheuristic solution which remains true until this time. Many of the metaheuristic approaches are still not used in commercial solutions for location allocation and also in free open source solutions. Particularly, a solution that delineate map from GIS data, can load facility and demand data or can generate facility and demand data randomly on administrative area and then solve location allocation problem using metaheuristic approaches, was not much explored

One of the metaheuristic approaches of solving location allocation problem is genetic algorithm which was first addressed by Hossage and Goodchild [5]. Other similar metaheuristic methodologies are simulated annealing, tabu search or neighbourhood search etc. There are some comparisons among these solutions in the researches [3, 6]. Among all methodologies, the performance of genetic algorithm and simulated annealing better than others. Performance of these two is very near to one another [7]. Among the metaheuristic solutions based on their performance from previous researches, we have chosen genetic algorithm and simulated annealing.

In this context, this study is motivated towards an integrated GIS solution of location allocation problem with various criteria like nearest distance, capacity and user preference of facility and existing network of the facilities by using metaheuristic methodologies. With addition to it, this research will also try to compare between genetic algorithm and simulated annealing performance on location allocation problem.

1.2. Research identification:

1.2.1. Research objectives:

Our main research objective is to solve location allocation problems using metaheuristic solutions. With addition to that we want to observe their performance. We shall achieve these objectives through a case study of school for location allocation in the city of Enschede. The whole objective is divided into three sub research objectives. By completing these sub research objectives step by step we want to finally achieve the entire research objective. Comparing and analyzing different metaheuristic solutions will be the final phase of my research objective.

The sub research objectives are as follows:

1. To determine optimal facility locations by using genetic algorithm and simulated annealing.
2. To put into action capacity and allocation of user with and without user preference into these two metaheuristic solutions.
3. To compare genetic algorithm with simulated annealing.

1.2.2. Research questions

Each sub research objective brings one or more research questions. From first sub research objective the following question can be derived:

1. How to prepare and process GIS data in the model for metaheuristic solutions to find optimal location?

From the second research objective the following research question can come out:

2. What will be the objective function with applying capacitated facility and user preference in order to get optimal location?

For achieving the last sub research objective in order to analyze and evaluate the formulated problem by metaheuristic solutions, firstly genetic algorithm and simulated annealing needs to be optimized though parameters fine tuning. At this stage the research questions that can be originated are given as follows:

3. When can genetic algorithm and simulated annealing be optimized?
4. What are the strengths or weaknesses of genetic algorithm in this problem context with compare to simulated annealing?

1.2.3. Innovation aimed at:

The innovation relies on the holistic view and extensive test of genetic algorithm and simulated annealing for a variety of location allocation problems of growing complexity. Hence road network, capacity of the facility and user preference for facility will gradually be incorporated with model where genetic algorithm and simulated annealing solve location allocation problem. Finally, genetic algorithm will be compared with simulated annealing to identify its strengths and weaknesses on this problem context.

1.3. Thesis structure:

This thesis consists of six chapters which are arranged as follows:

Introduction: Chapter one introduces the research problem with motivation. This chapter also contains research objectives and research questions and structure format of the thesis.

Literature review: Chapter two and three reviews the literature in location allocation problem and metaheuristic solutions like genetic algorithm and simulated annealing. These two chapters also discuss their common terms and associated terminologies. In chapter two location allocation problem was discussed with its types, classification, solution and software. In chapter three, genetic algorithm and simulated annealing were described in detail with their literature and connection with location allocation problem.

Methodology and general implementation: Chapter four elaborates data methodology and general implementation.

Case study: Chapter 5 deals with two case studies based on schools in Enschede and make discussion one their results.

Conclusion: Chapter 6 provides answer about research question, some general achievements and future work.

2. LOCATION ALLOCATION

2.1. Introduction:

An extensive literature review to provide an overview on previous research about previous research of location allocation and the different type of solutions. In this chapter the objective is to understand the terms and trends in location allocation problems. Location allocation is under research for more than a century. It started from Weber's location allocation problem in 1909. If we consider Weber's problem as an extension of the famous mathematician Fermat's distance minimization of a rectangle in seventeenth century, then it is a problem which has been dealt for over three hundred century [8]. Now-a-days the location allocation problem has grown with a lot of types and classifications. In this chapter, I will discuss the pros and cons of location allocation problem and try to find gap where much research light was not shed in the solutions of location allocation problem. Some common GIS software which can solve location allocation problem is also described in the last part of this chapter.

2.2. Some common terms:

There are some terminologies in Location allocation literature. Before going into details in literature review some terminologies should be explained. Facilities, location and customers or demands are referred as basic components of location allocation problems in [9]. In various location allocation problems, the role of those components may differ and can be used to typify that location allocation problem.

2.2.1. Facility

The term facility is used in location allocation problem to define an object whose spatial position is optimized through model or algorithm considering interaction with other pre-existing objects. Some examples of facilities include objects like outlet of chain-store, school, college, hospital, ambulance, fire-truck, ware-houses etc. Facilities can be characterized by their type, number, costs etc. [10].

In many location allocation models, one of the properties of facility is the number of new facilities that need to be established in the area of interest. Thus a single facility problem inside the location allocation model needs to establish only one new facility considering the existing facilities. This is very simple instance of location allocation model. Multi facilities inside location allocation model are more common where more than one facility is located simultaneously. [11].

Another important property of facility is type. Facility type includes the capacity of the facility and services of the facility. Facility can be characterized as capacitated and uncapacitated considering how much demand it can meet. If facility can supply an infinite demand then it is uncapacitated and when facility's capacity of supply is limited then it is capacitated. Facility can also be classified depending on number of services it is providing. A facility can provide only one type of service or a group of services. Example of single service facility is food shop which only provides food and example of multiple services is general hospital that provides multiple health supports.

Cost is another property of facility though which location allocation models can be differentiated [10]. Facility cost can be two types. One is fixed cost another is variable cost. Fixed type of cost depends on the establishment expense of facility. Variable cost has a relation with service delivery.

2.2.2. Demand or customer

The second essential component of location allocation algorithm is demand which is also known as customer [12]. A demand or customer is a person who needs accessibility to a service or to a supply of a good [10]. Since location allocation problem is connected with satisfying demand, it is important to know their distribution, quantity and behaviour.

If we consider demand distribution in space, it can be assigned uniformly over the area or network [13]. It can be assigned to specific point (geocoded) over the area [14]. It can be assigned on the centroid of the area [15]. However it can also be assigned randomly to simulate the problem over the area if there is no real data.

Another hindrance of depicting reality into the research is using demand in the location allocation model. In the classical location allocation, demand is used as weighted value in the node or in the smallest unit of continuous space like the researches [16-18] and [19] respectively. According to Murray the distribution of demand is either uniform or irregular [20]. Weighted demand means the aggregation of some customers on one point. Since location allocation is NP hard problem, if the size of demand decreases, it makes the decrement of computational complexity. When the quantity of customer is very large for example a million then it is better to use weighted demand according Erkut and Bozkaya (1999) mentioned by Sadigh and Fallah [21].

The demand of the customer can also be either deterministic or stochastic. In case of deterministic there is prior knowledge of demand while it is used into the model. In other case, demand will vary depend on type or service of facility.

2.2.3. Location or space

The third essential component of location allocation problems is space or location. There are three types of representation of space in location allocation problem. These are discrete, continuous and network based. In discrete space model, it is assumed that there is a prior knowledge of the candidate or potential sites. Since some best locations are selected from pre-selected potential locations, it is also referred as site selection model. Decision makers make the choice of candidate sites due to geographical or economical factor [10]. Examples of these factors are zoning regulation, presence of structure and land availability etc.

Some of the location allocation problems [15, 22, 23] deal space as continuous. One or more continuously varying coordinates determine all the possible site locations. These continuous locations are normally considered in Euclidian space [11]. It is also known is site-generation model. Because there is no presumption of potential sites for the model, rather appropriate site generation is done as output by the model.

Another type of representation of space is network-based. Network space depends on graph-theoretic approach. Model using this approach can solve problem with much larger size [10]. A network with either continuous or discrete space is considered in this type of space or location. Continuous network considers links of network for a continuous set of candidate locations. In discrete network, new facilities are only placed in the nodes [9].

In space there can be some forbidden area where site selection or site generation should not be done in the model. Similarly some areas can not be used due to some restrictions. For example new facility may not be built over water body or park. Site selection or generation in these areas should be avoided due to ineligibility.

2.3. Some location allocation problems from literature:

In the previous section we have discussed essential components of the location allocation problem. In this section we shall discuss two common and widely used location allocation problems, their commonality and differences. These two types of location allocation models are P median problem and covering problem.

2.3.1. P median problem

P median problem is considered one of the most studied, most general and simplest forms in location allocation problem. This problem identifies the median points among the potential points so that total cost can be minimized through objective function [8]. Facilities of this problem mostly include public type like school, hospital, ambulance, firefighting, shelter center etc. One of the objectives of p median problem can be to minimize time, distance etc. If P is equal to 4(P=4) then p-median means 4-median problem. So 4-median problem searches the locations of 4 facility or supply centers. Usually facility in this problem does not consider capacity and provides single service [10] and customer wants to go to the closest facility. Customer or demand is being beneficent having closest facility.

2.3.2. Covering Problem

Covering problem is another type of location allocation problem which also cover big portion in the literature. This problem intends to find facilities which provide customers the access to facility service within a specified distance. Here facilities want to cover maximum customer to reach their target. Solution of this model is suitable for service oriented farms which have multiple facilities as network. This model will work well where accessibility is an important factor for market share and profit. For example this model can be used for wireless tower establishing for network, setting siren alarm for emergency [24], chain store, multiple outlet etc. This model is also applied in switching circuit design, locating defence network, warehouse locating according to [25]. The difference of covering problem with P median and P center problem is covering demand. P median and P center problem both must cover all demands or customers. But covering model may or may not cover all.

Location set covering problem (LSCP) and maximal covering problem (MCP) are dichotomy in covering problem. Toregas (1970, 1971) defined LSCP as a problem that find minimum number of facilities to cover a specified number of demands within specific distance according to [24]. MCP tries to cover maximum demand with specific distance. In both problems distance is adjusted to achieve minimum number of facility in LSCP and maximum number of demand in MCP.

2.4. P median problem in literature:

Since in our research we are using P median location allocation problem we shall keep our discussion fix only on that class. Alfred Weber is considered as father of location allocation problem as mentioned in 2.1. Weber located a single warehouse by minimizing the total travel distance between the warehouse and a set of spatially distributed customers according to [26]. Weber problem was extended from single warehouse (facility)to multiple supply points (facility) by another research [27] in 1963 which was a p-median location allocation problem.

Hakimi considered facilities sited as nodes in the graph network [28]. So his optimal solution of p facilities will consist of only nodes in the graph network. In his model, demand is discrete and optimal solutions of facilities are also discrete. This was considered as great breakthrough in p-median location allocation problem according to [10]. Hakimi's solution reduced the search in graph from infinite number of points inside link to limited set of node. This triggered the location allocation solution to consider discrete space instead of continuous space.

In some of the past researches, facilities, demands were used through nodes or continuous space through synthetic data. In network based location allocation problem, graph network is used in many literatures. In facility location problem a network of discrete nodes were used for facilities and demands which was solved by [5]. Discrete nodes for facility or demand are also used by [18, 29], [16], [30], [31], [32].

Instead of using node as network, in some studies continuous space was used to locate optimal facility. Continuous space was created synthetically by the composition of cells in a study by [19]. Continuous space in facility location is also used through GIS data by [15]. Though in [15] Neema *et al.* used GIS data but they didn't consider road network for distance measurement rather they used Euclidian distance for demand. In another research[22], the same authors suggested that to construct more realistic and practical model some issues like obstruction, network need to be taken an account. Optimal locations found from the solution may exist in unacceptable area. For example, facilities cannot be established over water-body or over some other land use like road, buildings etc. So, for location allocation obstacles need to be identified like the research[29]. If data is in GIS format then a suitability analysis may also help to overcome this issue like the research done by [33]. So the trend of using non GIS data is switching towards GIS data.

For classical p-median location allocation distance is considered as straight line or radial trip from the facility to the customer or supplier. Therefore, the classical location allocation problem ignores route by network when locating facilities [34]. Although route by network is tightly coupled in the location allocation problem in some software like ARCGIS [1] or Flowmap [35] it can be one of the criteria in location allocation problem. This criterion plays important role for location allocation problems like assigning ambulance where an emergency patient needs to be reached in minimum time [14]. But where network changes frequently in different seasons (rural area without having built road) road network is not a good choice for that case. This is also true where location allocation depends on other purpose road (bicycle road) than main road and only main road is available at certain scale of data. Hence adopting road network in all types of location allocation problem is not required.

Very few researches were accomplished using road network. Proximity of road network was used in one research [3]. In two other researches [14, 36], the authors used road network through ARCGIS network analyst by origin destination cost matrix. ARCGIS itself is a commercial application. Network analyst of it needs to be bought separately since it needs separate license. According to our knowledge, still there is no research where origin destination cost matrix was used for road network in location allocation problem by open source software.

Now even location allocation problem is solved by fine tuned solutions with its best performance to find optimal locations, these locations may not be feasible if the optimal location falls over obstacles[29] like water-body, existing buildings etc. So, care should be taken for excluding forbidden area.

In the classical location allocation problem when facility is providing service like school, hospital, market it is always assumed that students, patient or customer will go to the nearest facility. In reality, facility may not be always the nearest one. Some of the facility may be chosen according to its type (i.e. specialized hospital; school depending on medium or religion, market based on commodity etc.). Focus should be given in research considering user preference.

2.5. Classification by Brandeau, Church, Murray and integration of models:

Integrating different location allocation models make them more realistic. For example, when median model becomes capacitated median model it becomes more realistic. In reality each server or facility has limited capacity. This approach is used in the research by [18] and [16]. The problem will become more complex when there will be multiple objectives instead of single one. According to Church in [3], integration of models make the model computationally more complex.

A classification is done to relate and distinguish location allocation researches up to year 1989 by Brandeau and Chiu [26]. They tried to provide an overview of major problems in location allocation and briefly describe the different types and how these relate to each other. They reviewed 54 location allocation problem including standard or commonly used problems such as the median, center, and coverage location problems, as well as less traditional problems. They classified all problems into three general classes through objective, decision variables and system parameters.

Classification through objective was based on optimization of some values through objective function and non-optimization types[26]. Classification through decision variables was based on facility, location service area, number of servers or facilities etc. System parameter type classification was based on topological structure (link, tree, network, Plane, n-dimensional space), travel metric (network-constrained, rectilinear, Euclidian etc.), travel time/cost, demand etc.

Church (1999) identifies four general classes of location models which are median, covering, capacitated, and competitive[3]. The median model and the covering model are described in previous section. Capacitated models consider the capacity of each facility with respect to demand. Competition models help the decision maker to consider other's competitive facility location and readjust own facilities. According to Church the recent trend is integrating multiple facility models; for example integrating p-median model with maximal covering model.

Murray also classified location allocation problems in 2010 [20] while he supported the classification of Daskin (1995). Unlike Church and Brandeau and Chiu's general classes, Daskin has used more specialized classification based on focus of the application (either public sector or private sector), number of facility (Single facility or multiple facilities), space for facility or demand, input information (either static or deterministic), dynamicity (single or for a period of time), type of solution (exact or heuristic or metaheuristic) and consideration of existing facility. With addition to Daskin's classification Murray added that there are also other types for which location allocation model may also differ from Daskin. These types depends on measurement of distance (Euclidian, rectangular, network-based), type of discrete facility or demand space (point, line, polygon, object), distribution of demand (uniform, irregular or other), number of services(single service or multi service),hierarchy of the facility (single or multiple). According to Murray, classification among location allocation problems becomes more complex now-a-days. Thus complexity occurs due to integration of models.

Li and Yeh [3] mentioned according to Church (1999) most traditional methods in solving location allocation problem cannot handle lots of demand points and facility in GIS datasets. To envisage more deep in solving location allocation problem using GIS tool and data we have analyzed some of the literatures of location allocation problem from past two decades. The best fit class is taken among the classifications mentioned earlier.

Data used in location allocation are two types. Data produced in randomly in rectangular or square space is synthetic data. Another is GIS data which contains spatial information and data is collected or used or produced from real world. We have also made two types classification based on synthetic data or GIS data from the previous location allocation models. In the table 2.1 and table 2.2 we have shown the classifications. All researches were done using metaheuristic solutions. These solutions are heuristic solution and do not trap in local optima.

Year	Researcher	Brandeau& Chiu-optimization	Daskin-facility type	Murray-distance, space,demand	Church-four class
1995	Gong *et el.*[29]	Minimizing distance	Multiple facilities	Euclidian distance	
1997	Gong *et el.*[18]	Minimizing distance	Multiple facilities	Euclidian distance	Capacitated
2003	Salhi *et al.*[23]	Minimizing transportation cost	Multiple facilities	Euclidian distance, continuous space	Uncapacitated
2005	Uno *et al.*[30]	Minimizing distance	Single facility	Euclidian distance	Competitive
2007	Silva[37]	minimize assigning cost of customer & establishing facility cost	Single facility	Euclidian distance, uniform distributed demand	Capacitated
2007	Medaglia *et al.* [31]	minimize waste shifting cost and number of affecting people	Multiple facilities	Euclidian distance, discrete network space	
2008	Jabalameli *et al.* [38]	Minimizing transportation cost	Multiple facilities	Euclidian distance, continuous space	Uncapacitated
2008	Liu *et al.* [39]	minimizing transportation cost	Multiple facilities	Euclidian distance, discrete network space	
2008	Yang [32]	Maximizing flow	Multiple	Euclidian	

		in the network	facility	distance	
2008	Neema *et al.*[22]	Minimizing distance	Multiple facilities	Euclidian distance, continuous space	
2009	Zhao *et al.* [40]	Minimizing setup cost & transportation time	Multiple facility	Euclidian distance	

Table: 2.1 Researches using non-GIS synthetic data using metaheuristic method.

Year	Researcher	Brandeau& Chiu-optimization	Daskin- facility type	Murray-distance, space,demand	Church-four class
2004	Correa *et al.*[16]	Minimizing weighted distance	Multiple facilities, public sector	Euclidian distance, discrete network space	Capacitated
2005	Li *et al.* [3]	Minimizing transport cost & maximizing population coverage	Multiple facilities, public sector	Euclidian distance, continuous space	
2008	Teixeira *et al.* [36]	Minimizing travel distance	Multiple facilities, public sector	Network-based distance, multiple-hierarchy	Capacitated
2009	Li *et al.* [19]	Minimizing travel cost	Multiple facilities,	Euclidian distance, continuous space, random demand	Capacitated
2010	Sasaki *et al.*[14]	Minimizing average travel time	Multiple facilities, Considered existing facility, public sector	Network-based distance, discrete network space	
2010	Neema *et al.*[15]	Minimize all weighted distance of population, air quality, noise level, land use.	Multiple facilities, public sector	Euclidian distance, continuous space	

Table: 2.2 Research using GIS data and metaheuristic method.

Though metaheuristic have been widely used in searching optimal values, there are few studies that ties metaheuristic and GIS together in resource and environment management according to

Li and Yeh [3]. So, it is significant that there are very few researches which had used location allocation problem using GIS tool and GIS data. There is hardly any research that had provided an open source solution which delineated map, used GIS data and solved location allocation problem by metaheuristic solutions and again delineated result using open source software.

2.6. Location allocation solutions

Location allocation problems were solved using exact, heuristic and metaheuristic techniques. In the exact solution method, the problem needs to find a set of locations as solution without using any approximation. In exact method, solution needs to complete counting to get optimal result. Selecting P facility out of N is a location allocation problem. To complete counting of this type of problem we need to consider all combinations of P facility out of N facility. According to the mathematical formula of combination, total number of combination becomes:

$$\binom{N}{P} = \frac{N!}{P! \times (N-P)!}$$

The p median problem dealt by Correa *et al.* [16] was selecting 26 facilities out of 43 facilities to meet the demand of 19710 students. According combination formula total combinations become to go for exact solution 421 billion.

$$\binom{43}{26} = \frac{43!}{26! \times (43-26)!} = 421{,}171{,}648{,}758$$

According to [3], selecting 20 cells (facilities) in a 100×100 cells needs a total combination of 4.03×10^{61}.

$$\binom{10000}{20} = \frac{10000!}{20! \times (100-20)!} = 4.03 \times 10^{61}$$

Similar example was also shown about selecting 25 facilities out of 10000 cells in a research [14]. All the authors of three researches chose metaheuristic method instead of exact solution to solve location allocation problem.

Li and Yeh [3] mentioned that facility location problem and its entire variant including most location allocation and P median problem are defined as NP-hard optimization problem. Karivi and Hakimi proved P median [41] and P center [42]as NP hard problem. NP is a term of complexity in computer science which means solution will be found in polynomial time by a non deterministic Turing machine. NP hard is a class of problems which are at least as hard as the hardest problems in NP. According to Avazbeigi [43] Turing machine is a standard computer model in computability theory introduced by Alan Turing in 1936. Here, polynomial time implies, if the problem has a size of n, computation time of that problem is no greater than a polynomial function of that problem size n.

So, $m(n) = O(n^k)$, where k is some constant that depend on the problem,

O = function of time.

According to Avazbeigi [43] NP hard problems may become not only any type of decision problems and search problem but also optimization problem. He suggested that if the problem is NP hard, solution of that problem should be shifted from exact to heuristic or metaheuristic due to complexity of the problem. Gong *et al.* [18] also mentioned similar. According to Gong *et al.* branch and bound (linear programming) of exact method theoretically can solve location allocation problem. But they also mentioned that due to nonlinearity and large scale of this location allocation problem, branch and bound is impracticable.

These factors have welcomed heuristic and metaheuristic solutions for location allocation problem.

Heuristic uses approximation in solution to get optimal or near optimal result. Khobam and ghadimi mentioned that according to Resende and De Sousa (2004) heuristic produces quick good quality solution but does not guarantee for optimal solution [44]. In other cases heuristic solution may be very far from optimal. First heuristic in location allocation algorithm is Cooper's iterative location allocation algorithm [45]. Greedy adding algorithm, alternating algorithm and vertex substitution algorithm were next heuristic algorithm for solving median problem [8]. According to the author many algorithms and techniques were made based on these three algorithms.

The location allocation objective function is neither concave nor convex but may have many local optima according to Cooper [27]. His dealt problem [27] needs a set of locations as solution which will minimize an objective function value. In mathematics, the value of a function is maximum or minimum when its derivative is equal to zero. But mathematical derivatives for many problems including location allocation problem may not exist. Even though derivatives of the location allocation problem may exist, due to many local optima trivial (not optimal) solutions are possible. According to Gong *et al.* [18] alternative location allocation (ALA), an efficient heuristic from cooper [45] also suffers at terminating at local optima.

Metaheuristics are also approximates solutions but unlike heuristic these can escape local optima. Since location allocation falls in global optimization problem [9], mmetaheuristic is a very good option to solve this type of problem and its all variants. There exist several metaheuristic techniques which are simulated annealing, genetic algorithm, tabu search, variable neighbourhood search, ant colony etc.

2.7. Location allocation in GIS Softwares:

The ARCGIS[1] location allocation analysis layer offers only six different problem types to answer specific kinds of questions which are minimize impedance, maximize coverage, minimize facilities, maximize attendance, maximize market share and target market share.

Classical location allocation problem like p-median problem is same as minimal impedance where in both case the objective is to minimize the sum of distances from demand points to facility. This problem type is used to locate private facility like warehouse and also public facility like library, museum and airport in order to reduce the distance or driving time from facility to demand. By reducing distance the solution wants to reduce the transportation cost.

To implement emergency rescue center location like fire station, police station in order to cover maximum demand, maximize coverage type problem is used. If the objective is to cover 100% demand for emergency support this solution will help to model it.

In minimizing the facility the output of the solution will be the optimal number of facilities to cover all the demand. The difference between minimizing facility and maximizing demand problem is in determining the number of facility. In maximizing demand the number is predefined where in other type the solution finds the minimum number. These two type solution can be used in hierarchical order.

With the assumption of demand weight will decrease in relation to the distance between facility and the demand point, facilities are located. This is designed for implementing private facility like superstore, pizza shop etc.

Maximize market share is to find the location that will maximize the market share in presence of other competitor's location. Example of this problem type is to find three locations that will maximize market share in presence of two competitors. In the contrary, target market share will show the locations of the private facilities to achieve the required market share. Example of this problem type is to find the locations of private facilities to achieve 60% of market share in presence of competitors.

All the problems stated above considers uncapacitaed facility which may not be practical in public facilities like school, hospital etc. or even in private facilities like pizza shop or supermarket. In private facilities the service providing time with respect to enormous demand cannot be infinitive. So, consideration of capacity is very important which is missing in ARCGIS. All the demand points were weighted in the ARCGIS implementation so far [14, 36]. Using separate demand point for each demand in ARCGIS needs to be investigated more while separate demand point is used in some past researches as mentioned by Church[46]. ARCGIS cannot deal location allocation problems with variation of objectives. It can only deal above stated six single objective location allocation problems. No variation of objective is achievable in ARCGIS. For Example, if demands are assigned according to the different criteria of facility ARCGIS will fail. For example 20% demands need facility type A, 30% demands need facility type B, 40% demands need facility type C and 10% demands need facility type D. Demands are random. In this case ARCGIS will fail to provide the solution.

Flowmap is a software [35] which is developed by the Faculty of Geographical Sciences of Utrecht University in the Netherlands. It can also solve similar type of location allocation model like coverage, expansion, relocation, recombination and combined model of expansion and relocation etc. The purpose of coverage and expansion model is same. But in coverage model, flow map used 'Spatial pareto' to reduce permutation of exact (brute force) solution. In expansion model, Flowmap solved four types of problem like "maximize customer coverage", "minimize overall average distance", "minimize overall worst case distance" and "maximize individual market share". In relocation model, flow map gives a solution to improve current solution by relocation of facility. Flowmap reduction model gives exactly opposite solution of expansion model with its four types of problem. Flowmap combined model is a combination of expansion and relocation model.

Like ARCGIS, Flowmap does not support capacity of each facility and types of demand rather it considers infinite capacity of facility. Another software LoLa [47] also solves some type of location allocation problem but without capacity. To read real data in GIS and to delineate solution into view Lola needs to add ARCView and scripting the process in a model [48]. There is no other non commercial solution which solves location allocation through map rendering in input and output using metaheuristic like genetic algorithm and simulated annealing.

2.8. Summary

In this chapter, we have discussed about some common terms of location allocation problem and some general and mostly used type of location allocation problem. Later we have discussed location allocation problem's grown complexity through its classification. We have also discussed existing solution types for location allocation problem and some existing GIS software that provide non-heuristic solution of location allocation problem.

3. GENETIC ALGORITHM AND SIMULATED ANNEALING – METAHEURISTIC

3.1. Introduction:

Combinatorial optimization problem needs to find optimization from millions of combination of solutions. Metaheuristic is a solution to this combinatorial optimization problem. Metaheuristic is a kind of heuristic solution but unlike heuristic it can escape local optimal. Several metaheuristic solutions exist in location allocation arena like genetic algorithm, ant colony optimization, simulated annealing, plant simulation, variable neighbourhood search etc. Among these solutions genetic algorithm and simulated annealing are used in many location allocation researches. As solution, Simulated annealing is a serious competitor to genetic algorithms [7]. The authors suggested that it is worth to compare the results of simulated annealing and genetic algorithms. Both of them are derived from analogy with natural system and they can deal with optimization problem of same type. So, among the metaheuristic solutions genetic algorithm and simulated annealing are chosen for this research.

Genetic algorithm is a metaheuristic search technique which uses the analogy of natural evolution into search algorithm. It is capable of finding optimal or near optimal and to avoid to be trapped in local optima. Hosage and Goodchild [5] first identified the enormous potential of genetic algorithm over heuristics in applying on certain class of location allocation problem. Since then it has been applying in the realm of location allocation almost three decades. It has also been successfully used in many other disciplines like control, design, scheduling, robotics and machine learning etc. [7].

Simulated annealing is also a metaheuristic random search technique which imitates the analogy of a hot metal in its cooling and freezing into a minimum energy crystalline structure (the annealing) and the search for an optimal in a more general system. Simulated annealing is simple to implement. It also has been applied into wide number of real world problems. In travelling salesman problem and optimal layout of printed circuit board problem, simulated annealing is proved efficient [7].

Before going deep into how location allocation is solved by genetic algorithm or simulated annealing, we shall describe genetic algorithm and simulated annealing and their terminologies in the coming section. We shall also describe genetic algorithm and simulated annealing in location allocation research.

3.2. Genetic Algorithm:

A genetic algorithm is a problem solving algorithm which imitates natural selection or natural genetics. It is a search technique to find optimal or nearly optimal solutions of search problems. In the decade of 1960 John Holland thought and worked with genetic algorithm. But his first publication was appeared with the title "Adaptation in Natural and Artificial System" in 1975 [7]. Holland invented genetic algorithm as metaheuristic search based on "Survival for the fittest" a common ideology of biology. He introduced not only mutation and but also reproduction from biology into the artificial system. Hence the terms Gene, Chromosome, Individual, Population, Crossover and Mutation are used in this search technique. Section 3.2.1 to 3.2.9 explains some common genetic algorithm terms and associated terminologies. Simulated annealing terms and corresponding terminologies are also explained in a later section.

The steps of genetic algorithm:

The main steps of genetic algorithm are simple. Genetic algorithm starts with bottom up approach. This means it starts with a set of solutions and ends with optimal one. The following steps of genetic algorithm are also generic. The same steps can be used in many optimization problems.

Step 1: Create the initial population by producing G set of individuals or Chromosomes.
Step 2: Evaluate the fitness value of each individual in the population
Step 3: Repeat (creating new generation of population)

- *a. Selection of parent from individuals in population*
- *b. perform recombination or mutation to generate new individual*
- *c. add new individuals into the population*
- *d. remove individual considering low fitness or randomness*

Go to step 3 until termination criteria are satisfied.

In the first steps, genetic algorithm initializes solution randomly and generates population. Then it measures the fitness value of each individual of population through objective function. From third step, it performs recombination and mutation to generate new population. The fitness value is checked for all individuals. The individual with higher value is evolved. This process will be continued until it is stopped by any criteria. Finally an individual with best fitness value will be selected as solution. The figure 3.1 also shows genetic algorithm in block diagram.

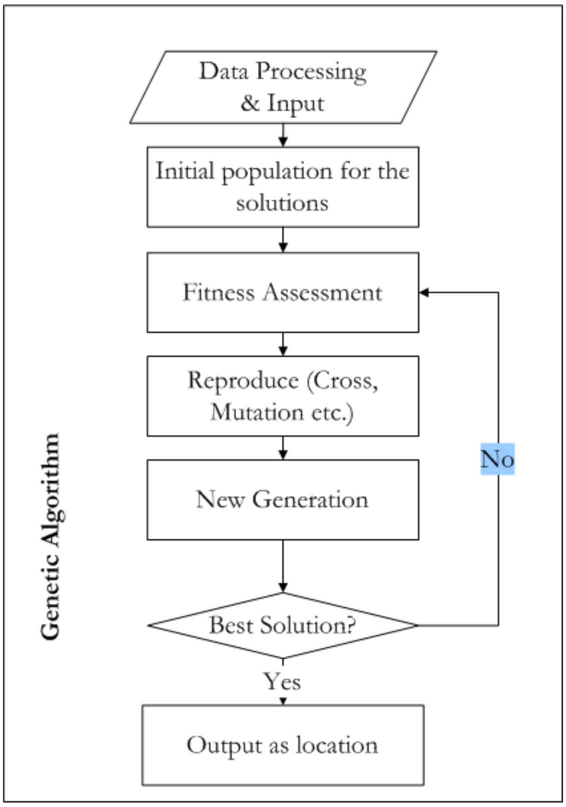


Figure: 3.1 Genetic algorithm block diagram

3.2.1. Individual and Chromosome:

An individual in genetic algorithm is a single solution. It has two forms[7]. Of the two forms one is chromosome or genotype and another is phenotype. Phenotype is the expression of chromosome that is used inside some of the models. For example if the chromosome is in integer then phenotype in the model can be binary format. Sometimes individual and chromosomes are used synonymously in the literature like [3]. In this research, we shall also use them as synonymously.

Chromosome is the raw genetic information that is dealt by genetic algorithm. A chromosome is encoded as bit of strings. This string can be string of binary number, integer or float [7]. So inside algorithm it may use as strings or array. It can be used as same string or used as different strings so that it fits the model. For example, in figure 3.2 a chromosome may consists of integer number. But inside the algorithm it can be used in binary format like figure 3.3. A chromosome comprised of a sequence of genes. According to figure 3.2 and 3.3 these genes can be integer and binary respectively. An important step for implementing genetic algorithm is to design chromosome according to the problem domain. Each chromosome must define a solution of the problem.

1	2	3	4	5	6	7	8	9	10	11	12
Gene1	Gene2	Gene3	..	..						..	Gene12

Figure: 3.2 An example of a chromosome or individual in integer format

0001	0010	0011	0100	1001							1100
Gene1	Gene2	Gene3	Gene4	Gene5	..	..				..	Gene12

Figure: 3.3 An example of a chromosome or individual in binary format

For location allocation problem chromosome can be expressed by location variable x and y. For example in one literature[3] chromosome is used as follows

Chromosome = $[x_1y_1\ x_2y_2\ x_3y_3\ x_4y_4 \ldots \ldots \ldots \ldots\ x_ny_n]$

In the above presentation, the chromosome can be represented also in binary format. For example, if 19 locations need to be selected out of n locations, then the summation of all binary 1 will be equal to the 19. Binary 1 means it is selected as location and binary 0 means this location is not selected. Similar approach was taken by Domínguez-Marín et al. [49] which has been shown in the following figure.

1	1	0	0	1							1
x_1y_1	x_2y_2	x_3y_3	x_4y_4	..						..	x_ny_n

Figure: 3.4 Binary relation with point location

3.2.2. Chromosome design:

Some of implementations like [3, 15] [13] and [23] used binary number to design the chromosome in their genetic algorithm solutions. All the authors created their chromosome from the point of optimal locations.

$$\text{Chromosome} = [x_1y_1\ x_2y_2\ x_3y_3\ x_4y_4 \ldots \ldots \ldots \ldots\ x_ny_n]$$

If we consider x coordinate and y coordinate in UTM format integer in order to convert into binary then what we shall get is as follows:

x coordinate, 259245 = 00111111010010101101

y coordinate, 474742 = 01110011111001110110

In total there are 38 binary digit for x cordinate and y cordinate without considering decimal precision. So, lenght of array or string for gene in chromosome is already 38. If there is 20 optimal location that means 20 genes then legth of array 760 and this is the reason the reseearchers limits their optimal location maximum 10 in [3]. Though comber *et al.* [13] limits their optimal location 27 but his length of chromosome was also only 10 which means they only used 10 genes into chromosome. Comber *et al.* mentionaed that chromosome length will be too long if optimal location is more than 15 [13]. Limiting of their chromosome length is due to binary consideration of the chromosome format.

Chromosome design in genetic algorithm is also important. Different real world problems which were solved by genetic algorithm used different types of chromosome [7]. It is not necessary to always design chromosome into binary format. Classical traveling salesman problem was solved by genetic algorithm where chromosome was integer instead of binary [50]. Similar approach was also taken to solve bus stop optimization in location routing problem [51]. In this location allocation model we shall use simple chromosome which will be constituted from integer value of the index number of potential location. In the figure 3.5 the value 0 means index of first point location in the potential facility. Here chromosome length is same as the number of potential facility. The number of optimal facility will be used to traverse chromosome to calculate fitness value of each chromosome.

Chromosome=	6	5	1	7	2	9	8	4	3	0

Index	Point	
	X coordinate	Y coordinate
0	259245.630	474742.530
1	258793.750	472166.090
2	258095.412	472679.849
3	263401.000	470368.190
4	263296.880	470779.000
5	262857.250	471485.280
6	262063.324	471036.376
7	261856.506	470865.392
8	258550.149	467843.558
9	259000.330	470341.550

Figure: 3.5 Index of point is used in chromosome in genetic algorithm in the model

3.2.3. Population:

Populations are collections of individuals. There are two important characteristics of population inside genetic algorithm. One is in defining the size of the population in the design of genetic algorithm and another is to create the initial population in the beginning of genetic algorithm. In figure 3.6 the simple example of population is given. This population consists of five individuals. Each individual has gene value of integers.

1	6	19	22	2	Individual 1
3	13	9	8	18	Individual 2
20	4	12	16	25	Individual 3
0	14	7	17	20	Individual 4
5	23	15	24	11	Individual 5

Population

Figure: 3.6 An example of population of initial solution

Initial population is chosen randomly in most cases. In an ideal case, the initial population should be large enough such that it explores the whole search space. It should have diversity of data from the search space otherwise it will only explore a small part of the search space and may fail to reach at the global optimal. However in the problem, the complexity of the problem will depend on the size of population. Depending on the complexity, the size of population may increase or decrease. The population size should not be very big. Otherwise the completion time of the algorithm will also increase[7].

3.2.4.

3.2.5. Recombination and Reproduction and Mutation:

Recombination, Reproduction and mutation are known as genetic operators. Recombination and mutation are two different processes for breeding by the individual or individuals from the population. Recombination is also known as crossover because it imitates the crossover technique from biology between two chromosomes and produces a new chromosome. Recombination occurs between two individuals at one single point or multiple points at the individuals. In classical simple genetic algorithm single point recombination is used. The following figure is an example single point recombination. From the 1st individual the 1st three genes are taken and from the 2nd individual the last two genes are taken to generate the offspring 1. Similarly offspring 2 is also created. So crossover occurs after 3rd point in both individuals.

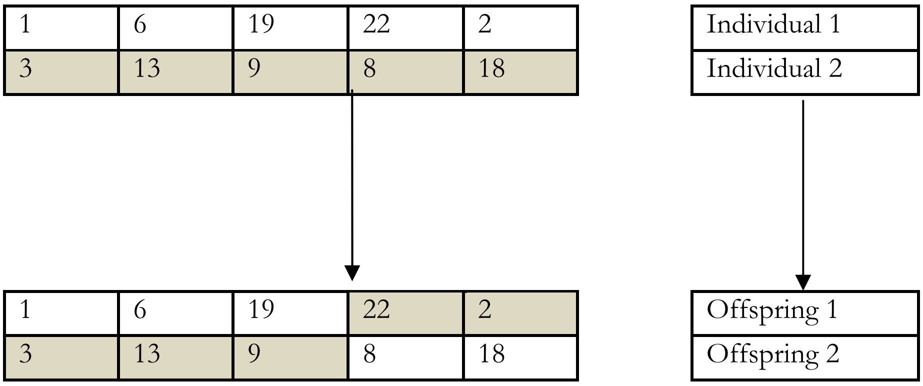


Figure: 3.7 Recombination or crossover on a single point

3.2.5.1. Chromosome Crossover

Chromosome crossover mechanism was followed to recombine to chromosome. Suppose we have two parent chromosomes to recombine. One chromosome is defined as parent 1 and another chromosome was taken as parent 2. From these two parents new child chromosome will be created according to the following algorithm. This algorithm is modified from a genetic algorithm implementation of travelling salesman problem [50]. Here initial random number will only be generated from the range between 0 and the optimal number of facility.

So, 0 ≤initial random number ≤Optimal facility number

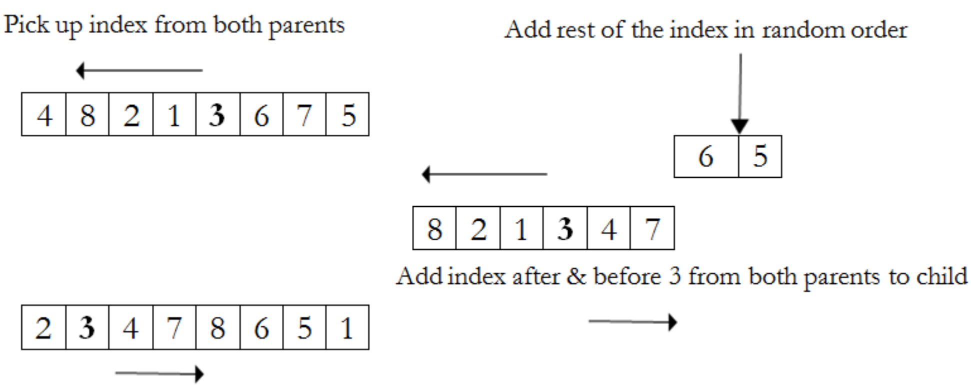


Figure: 3.8 Parent chromosomes crossover create child chromosome

1. Pick an initial random gene between 0 to the number of optimal facility at first parent chromosome. In figure 3.8, randomly picked gene is 3.

2. Locate that picked gene at second parent. In the figure, locate 3 in the second parent.
3. Start creating the new child chromosome by inserting the value of the random gene as first gene in child. In figure create child chromosome with value 3.
4. Go to left direction from the first parent and read the gene value. If new gene value does not exist in child add it into child. In the figure, go to 1 which is left to the index 3. Add it to child chromosome.
5. Shift into right direction from the second parent and read the gene value. If new gene value does not exist in child add it into child. In the figure in the second parent, take 4 which is the right index of 3 and add it to child chromosome.
6. After finishing the reading from first and second Parent if the chromosome length is less than actual length then fill the rest of genes in random order with all not included genes into child chromosome. In the figure, 6 and 5 were not included in the child chromosome, so include these two indexes randomly.

3.2.5.2. Chromosome Mutation

Mutation acts by changing value or values of a single individual. If recombination creates better offspring by exploiting current solution then mutation helps diversity or randomness by exploring the whole search space. According to Jaramillo [52] mutation in genetic algorithm prevents the solution of being trapped in local optima. He considered it as a secondary mechanism of genetic algorithm while the first mechanism is chromosome crossover. Unlike chromosome crossover's two chromosomes, chromosome mutation needs one chromosome. A single gene will be changed the according to the following random value.
Here, 0 ≤ initial random number ≤Optimal facility number

If optimal facility number = 4, then the following chromosome will be changed in the highlighted area. If the initial random number = 2, then in the mutated chromosome will be second index or third gene will be changed with any other gene in that chromosome. This scenario is shown in bold format of mutated chromosome in the following figure 3.9.

Chromosome=	6	5	1	7	2	9	8	4	3	0
Mutated Chromosome=	6	5	**3**	7	2	9	8	4	**1**	0

Figure: 3.9 Mutation in the chromosome

3.2.5.3. Reproduction-Injecting elite chromosome into population

Reproduction produces clone of single or group of good individuals. Reproduction prevents from loosing good individual through recombination. The individual that has the best fitness value among all individuals in a population is called elite individual. Elite individual of one generation is local elite individual. Elite individual of all generation is global elite individual. Local elitism in genetic algorithm is injecting at least one elite individual into next generation in order to keep alive the best individual in the next generation. This local elitism is passing current elite individual into the next generation. Li and Yeh [3] used global elitism in their algorithm. Global elitism is also used by Salhi and Gamal [23] in their genetic algorithm implementation of location allocation algorithm and made robust genetic algorithm implementation. But in order to find the relation between local optima and elitism we shall use both types of elite individual in our model and compare the result as analysis.

3.2.6. Selection of parent:

Parent for recombination is selected from population through this process. The purpose of this process is to have the better offspring and to lead genetic algorithm to global optimal solution. Though this process randomly pick individuals, still the higher the fitness value the more the chance is for the individual to be selected. Well diverse randomness in gene and good fitness value in chromosome is important to reach in global convergence. There are several techniques [7] of choosing parent like the Roulette wheel method, the random selection method, the ranking selection method and the tournament selection method used in genetic algorithm. Normally only one selection technique is used in a problem. Sometimes other metaheuristic search method like local search, simulated annealing and neighbourhood search is used to search and select the parent to produce better offspring. Then this type algorithm is called hybrid genetic algorithm.

a. Roulette wheel selection:

All individual's fitness value creates the wheel. The space (circumference) in the wheel is proportionate to the fitness value. The higher the fitness value the better the chance is to be selected of an individual. Similar to gambling wheel, the weaker individual may win and hence can take participation in creating offspring. To prevent biasness in the selection process, the individual should not be ordered according to the high fitness value in the wheel. In figure 3.10 we see there are 5 chromosomes which are arranged on a wheel proportional to their fitness value. These chromosomes are not ordered in a sorted way. Now like the analogy of gambling wheel, any of chromosomes can be selected when it will be rotated. But the higher the space (fitness), the higher the chance is.

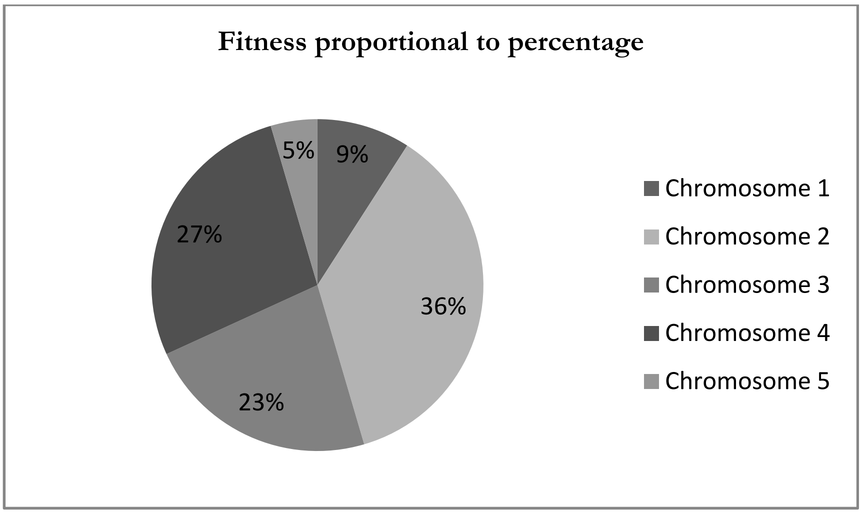


Figure: 3.10 Roulette Wheel for Chromosomes

b. Random selection:

In this selection method, individual for recombination is selected randomly from the population. There is no influence of fitness value here. It is simple to implement but may produce population of weak fitness.

c. Ranking selection:

Ranking selection ranks the entire individuals in the population according to the fitness value from 1 to N as integer range and select individual after ranking. The weakest individual will get the rank of 1 and the strongest fitness value will get the value of N. After ranking there are also many ways for selecting individuals. These techniques are also based on randomness. Ranking selection overcomes a problem of space proportionate to fitness in roulette wheel. The problem of roulette wheel is if the fittest individual has 90% fitness then its space (circumference) on the wheel will be 90%. Then chance for other individual's winning is very less and it will lose diversity. The randomness of ranking selection yields slow convergence but prevents early convergence.

d. Tournament selection:

Tournament is arranged from randomly selected individual to select the individual with highest fitness value from random selection. The winner will get chance in the mating pool. In the repeated way, winners of the tournament will fill the mating pool and will drive the population with higher average fitness. Hence it will finally lead genetic algorithm to the global optimal solution.

3.2.7. Generation:

Generation is state of population which comes from removing the weak individuals from population and adding new individual from population. Iteration inside step 3 of the algorithm (at section 3.2) creates a new generation. A new generation will be created while parents are selected and then perform genetic crossover or mutation. Replacement happens to replace unfit less fit chromosome.

3.2.7.1. Creating next generation

There are many ways to create next generation. When genetic algorithm creates next generation, it creates equal number of individuals according to population size. Parent selection methods from 3.2.5 are used to select the parent chromosomes from current generation. Parent chromosomes are father chromosome and mother chromosome. These two new chromosomes will create two new child chromosomes. Then crossover of parent chromosome will only occur if a random number that appears is less than crossover rate. The rule becomes:

if generated random number is between 0 and 1 < Crossover Rate
then
CrossOver between parent chromosomes create two child chromosomes.

else
child1 chromosome = Father choromosome;
child2 chromosome = Mother chromosome;

After crossover two child chromosomes will be created and both of them will be mutated if a random number that appears is less than mutation rate. The rule is as follows:

if generated random number is between 0 and 1 < Mutation Rate
then
child1 chromosome =Mutate (child1 chromosome);
child2 chromosome = Mutate (child2 chromosome);

After creating all individual for next generation, elitism may be applied for next generation.

3.2.8. Replacement:

Replacement is the process in genetic algorithm which replaces individual by other one in the population. When two parents create two more offspring the size of the population will be increased if all are kept. So either some individuals or all previous ones need to be replaced. There are few replacement methods which are used in genetic algorithm which are random replacement, both parent replacement and weak parent replacement etc.

3.2.9. Termination criteria:

There are several termination criteria for stopping genetic algorithm which are given as follows:

Maximum generation: after evolving maximum generation genetic algorithm will be stopped.

Specific time: Genetic algorithm will run up to a specific time. Then it will stop.

Unchanged fitness value: If fitness value is unchanged for a specified number of generations then genetic algorithm will stop.

3.3. location allocation by genetic algorithm:

Goodchild and Hosage first solved location allocation problem by genetic algorithm in 1986 [5]. Till then it is almost three decades where genetic algorithm has been used successfully in location allocation problem. They used genetic algorithm to solve the p-median problem. Their goal was to locate p facilities of n nodes such that the total distance between each node and its closest facility is minimized. This problem achieved nearest facility as single objective

Instead of single objective some location allocation problems need to achieve multiple objectives. Jones said that about 70% of papers describing the use of metaheuristic for multi-objective problems are about genetic algorithm According to [53]. Multiple objectives were solved by using genetic algorithm in a research for locating multiple facilities with three objectives like maximizing the population coverage, minimizing the total transportation costs and minimizing the proximity of the road by [3]. Market locations with the lowest transportation cost to the nearest market where the crops can be sold at maximum price was solved by Taylor and Pitaksringkarn [2]. This research also addressed multiple objectives by genetic algorithm.

In a study [31], waste dumping locations were allocated with a low cost operating network and maximum distance from population living area. Minimizing the distances from parks to highly populated areas, highly air polluted areas, noisy areas and areas without parks were considered to find parks' locations by Neema *et al.* [15]. Researches [14, 15, 31] dealt with multiple objectives through genetic algorithm method. Multiple objectives give the flexibility of using different objective on same location allocation problem. From these researches it proves that genetic algorithm can successfully solve multiple objectives.

Sasaki *et al.* [14] solved optimal facility for minimal distance considering the amount of demand. But they didn't consider the capacity of the service center at the time of demand consideration. Even if genetic algorithm can handle multiple objective, when location allocation problem handles capacity it can be computationally burdensome. Some types of facility location problem were compared in the study of Jaramillo [52] where the performance of capacitated facility problem's solution was not good. Its computational time was higher among all types location allocation problem solved using genetic algorithm. A hybrid genetic algorithm was used in location allocation problem which dealt with the capacity of the facility by Gong *et al.* [18]. The problem space was not spatial but simulated from random number. In spatial context, an improved genetic algorithm with Hilbert curve was used for capacitated facilities by Li *et al.* [19]. But the approach of genetic algorithm with Hilbert curve by Li *et al.* [19] was faster in capacitated facility location problem.

The performance of genetic algorithm also depends on fine tuning its own parameters. Genetic algorithm has sensitivity in its parameters. These parameters are the values of genetic operator. Li and Yeh [3] also weighted these parameters in their research but didn't mention any methodology to fix their rate for genetic algorithm to optimize result. However Jaramillo took a different approach of using genetic operator's rate. He used dynamic approach where genetic operators' value changed over time.

In 1999 Church mentioned that only small data was used in location allocation problem before use GIS data. But even GIS data was used it was small in other researches. A recent research by [14] was about locating 27 ambulances in 35 locations. They used shortest distances from main road network weighted by emergency case number. The evaluation of objective function was based on minimizing the network distance. Neema *et al.* solved multiple objectives by genetic algorithm, they used only 90 centroids as demand points [15]. Li and yeh did not even use demand as point rather they used demand density because of raster data[3]. Surprisingly Sasaki *et al.* had 21,211 geocoded demand points but they also didn't utilize the benefit of such amount of geocoded data in their problem[14]. Rather like Neema *et al.* they used only weighted demand using centroid from small census area. User preference was also not considered in the above researches.

However, Correa *et al.* solved location allocation by 19710 geocoded demand points [16]. They solved location allocation problem to select 23 facilities out of 43 potential ones using this amount of demand points. So, still there are not many researches which dealt location allocation problem with a large GIS dataset.

3.4. Simulated annealing

Simulated annealing is a random search optimization method for solving combinatorial optimization problem. Simulated annealing is able to solve this problem. Simulated annealing concept was grabbed from the formation of crystals into solid during cooling. The physical cooling phenomenon discovered long ago is that the slower the cooling the more perfect the crystal formed. From cooling procedure, when temperature goes down the energy of the physical system naturally converge to minimal. For the random movement of the physical system, the probability for a specific system configuration depends on energy and temperature [7]. This probability is described by Gibbs law as follows:

$$p = e^{\frac{E}{kT}} \qquad \text{..............................Equation 3-1}$$

Where p = probability,

K=boltzman constant,

E= Energy of the system,

T = Temperature.

Using same analogy from the previous formula, Kirkpatrick *et al.* [54] showed how a annealing process of solid can be simulated for optimization process.

Simulated annealing is a random search optimization method that starts with an initial random solution. Then simulated annealing continues to seek better solution through iteration. Inside iteration, it randomly chooses a new solution in the neighbourhood of the previous solution inside iteration. If the fitness function of the new solution is better than the fitness function of previous solution, new solution is accepted as new current solution. If the fitness function value does not get better, new solution will only prevail with a probability of following:

$$p = e^{-\frac{f(x)-f(y)}{T}} \qquad \text{.................................Equation 3-2}$$

Which is similar to Gibb's law and where f(x)-f(y) is the difference of fitness function between old and new solution. Using this probability for inferior solution helps simulated annealing to be trapped in a poor local optimum. Because it is not only accepting better solutions but also with certain probability it accepts inferior solutions. Here T is temperature similar to the temperature of annealing of solid. Murray and Church [4] said that when temperature T is high most perturbation is allowed in simulated annealing. If the value of T is high, there will be higher chance of accepting moves of fitness value. But if T value approaches to zero most moves of fitness value will be rejected [55]. This is the reason why in Simulated annealing starts with high temperature so that it can avoid premature trap of local optimal. In each temperature, simulated annealing tries certain number of moves and temperature reduces gradually and slowly.

A cooling rate of temperature is used to reduce initial high temperature by iteration. In the following algorithm it was set with δT. δT can be any fraction between 0 and 1. But the higher the value is, the higher the number of iteration is. For example, 0.999 creates higher iteration than 0.90. In two researches this value was used between 0.90 and 0.98 [3]. Since simulation annealing is an iterative process, the number of iteration has also a strong impact over the quality of solution.

The algorithm of simulated annealing is given below:

Step 1: Initialization

Select an initial solution s from S;

Select an initial temperature T_0;

Set absolute temperature $T_a = 0.001$;

Set cooling rate of temperature $\delta T = 0.90$; // δT *can be any fraction between 0 to 1*

Step 2: Improving the solution.

Set repetition counter $n = 0$;

Repeat the following operations until $T_0 \geq T_a$;

Generate, randomly, a solution s' from the neighbourhood N(s);

Calculate $L = f(s') - f(s)$;

If $L \leq 0$ *then set* $s := s'$

else select a random number, X, from U(0, 1), if $X < \exp(-L/T)$ *then*

set $s := s'$

Set $n = n + 1$;

Set $t = t + 1$ *and* $T = T * \delta T$

Stop if the stopping condition is met or Go to Step 2.

In the first step of algorithm, some variables like initial temperature, absolute temperature, cooling rate etc are initialized. These variables fix the number of iteration and imitate the decreasing of temperature of metal.

In the second step candidate solutions randomly are generated from neighbourhood solutions pace. Only the solution which is best than previous one will be selected as current solution. If current solution is worse than previous one then it can only be selected with certain probability. Second step will continue until initial high temperature meets with absolute temperature.

3.5. Location allocation by Simulated Annealing:

location allocation was first solved by simulated annealing in 1986 [4]. But Murray and Church mentioned that first research of simulated annealing result for location allocation took much computation time with compare to other alternative solutions. Selim and Asultan in 1991 tried with different approach like clustering with simulated annealing in (p median) one of the location allocation problems [56] and find good result as their algorithm converges to global optimal. Similarly Kincaid [57] in 1992 also used simulated annealing for obnoxious facility location and found good result in computation time. In 1995, Murray and church reanalyzed simulated annealing solution strategy for two types of location allocation problem like p median and maximal coverage problem. According to them these two problems of location allocation are related but structurally different. They used interchanged procedure in simulated annealing. This interchanged procedure is the transformation of one solution into better solution. They found better result using simulated annealing in terms of identifying optimal solutions and mean solution value for some standard non GIS location allocation problem based on operation research (OR) library dataset [58]. In 2000, Chiyoshi and Galvao [59] and in 2004 Levanova and Loresh [60] also tested simulated annealing against OR library dataset and was successful to find optimal solution of some of those standard problems.

In 2005 Arostegui *et al.* compared among simulated annealing, genetic algorithm and tabu search with various facility location problems under time limitation, solution limitation and unrestricted condition[6]. They found both simulated annealing and genetic algorithm performances depend on problem type and criterion. Simulated annealing is simple to implement and gives comparable result with genetic algorithm. Simulated annealing is also compared with genetic algorithm and tabu search by Li and Yeh [3]. The author claimed the performance of genetic algorithm is much better than simulated annealing. In simulated annealing if current state is better than previous state than current state is accepted. Otherwise, current state will only be accepted with certain probability. Both options cannot be true at the same time. But Li and Yeh used both option true at the same time. However, according to our knowledge their research was not only last which compared simulated annealing with genetic algorithm in location allocation but also first that integrated GIS in the comparisons. After this research, a solution with tightly integrated GIS in location allocation and implemented metaheuristic like simulated annealing is yet to be done.

3.6. Summary:

In this chapter, we have discussed about two metaheuristic algorithms genetic algorithm and simulated annealing and their ins and outs. Between these two algorithms simulated annealing is easy to implement and has less parameter than genetic algorithm. On the other hand, genetic algorithm has lots of option in its implementation. From past researches, we have seen that both algorithms have been used and compared in location allocation problem successfully. But as open source solution for location allocation problem these two metaheuristics are not much explored.

4. MATERIALS, METHODS AND IMPLEMENTATION

4.1. Introduction:

In this chapter we shall discuss the working process of location allocation model. What are the inputs, what are the outputs, how GIS will be integrated with searching algorithm like genetic algorithm and simulated annealing will be described in the upcoming sections. Before entering into methodology we shall discuss in the next section about data and its collection and how data needs to be prepared to fit for location allocation model.

4.2. Data:

For this research, students are considered for primary school from an age from 4 to 12 in Enschede city according to the municipality website of Enschede [61]. Based on this data, in chapter 5 several location allocation problems were formulated as case studies.
The information required for this research has been provided through shape file. Shape format is also taken as widely used format for GIS. For these reasons this format is used in my prototype as data input format. An assumption of data format is made that data will be provided or converted in UTM format. This assumption makes the model simple.

To use data as input in our case study, we need data for facility with its capacity and data for demand. Since we consider distance in our model through road network then we also have to have road network data. In the following subsections we discuss about facility data, demand data and road network.

4.2.1. Facility:

For applying into location allocation model, data from Enschede will be used. Here schools are geocoded as public facility. There were 39 public schools according to the data of 2003. There are also private schools in Enschede. For this research we consider only public schools. With each public school the capacity of that school in 2003 is also provided in the shape file as an attribute.

4.2.2. Demand:

Demand is obtained from the population where age group is in the range of 4 to 12. Data was downloaded from municipality website for the year of 2003 [61]. There are 68 neighbourhoods in Enschede. Primary school student locations are distributed in each neighbourhood.

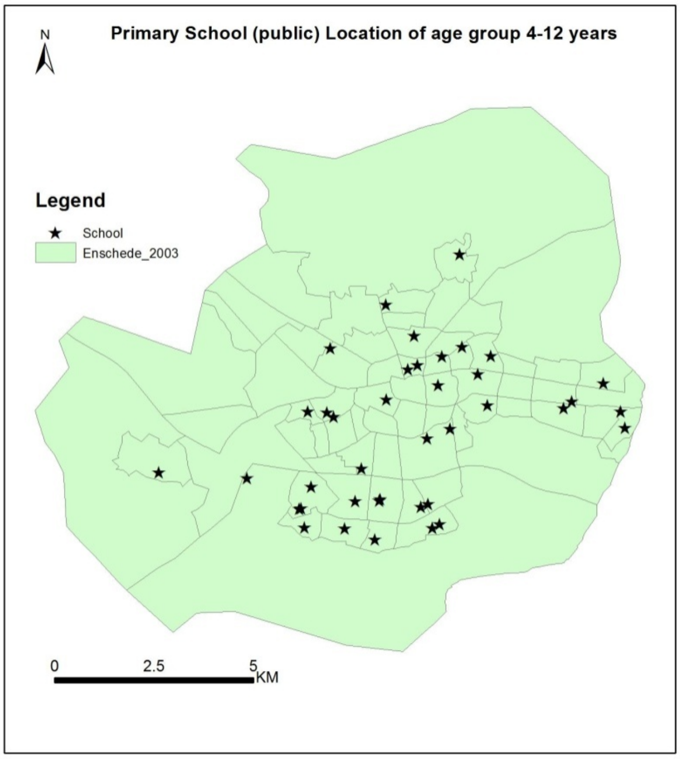


Figure: 4.1 Existing Primary School Location in Enschede

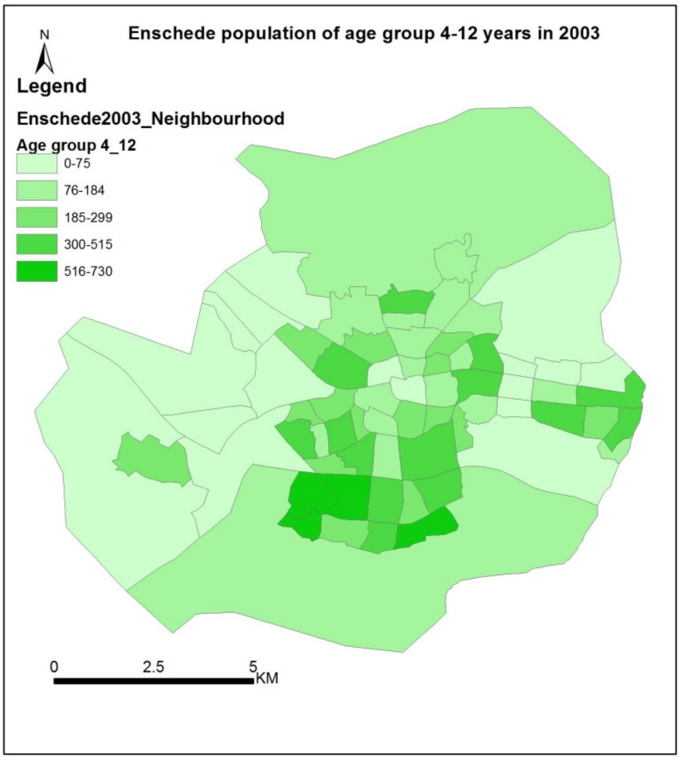


Figure: 4.2 Demand of age group 5-14 in Enschede

Demand data was downloaded in Microsoft Excel format. To add extra attribute of demand data in the polygon shape file of Enschede a join operation was done through Neighborhood ID.

Neighborhood ID is taken from website. The polygons are representing neighbourhoods of Enschede. In appendix 1, data is available in table.

4.2.3. Road network:

For road network main road network of Enschede is provided. Assumption of data quality is all roads should be connected so that distance through network can be computed. The following figure shows the road network of Enschede used in our model. Some of the main roads in North Enschede were not used since these are connected in data.

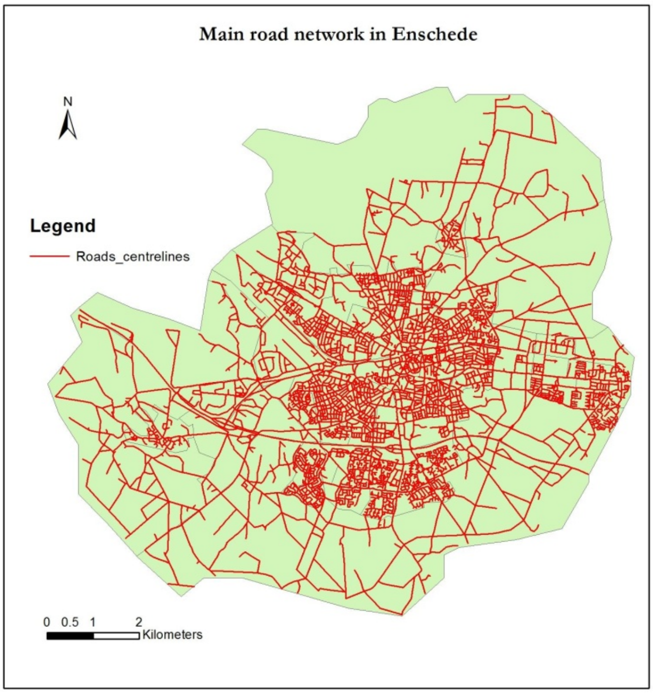


Figure: 4.3 Enschede main road network for model

4.3. Methodology:

Selecting best schools in Enschede based on distance is a P median location allocation problem. In the methodology, we shall part by part analyze the entire problem. Spatial locations of facility and demands are considered in our model For GIS data; it is assumed that all information is in the shape file. After having input into our model and setting parameter of the model, GIS data will be viewed as map in our model. Either genetic algorithm or simulated annealing will process all the information from GIS data. Optimal locations will be delineated as map and all relevant information will be written as a text file. In the figure 4.3 we have shown general methodology of our working procedure. We shall divide the location allocation model into six major parts which are given as follows:

I. Input into model,
II. Objective function of the model,
III. Genetic algorithm in the model,
IV. Simulated Annealing in the model,
V. Tools for map display and genetic algorithm-simulated annealing,
VI. Output of the model.

Loading data for both input and output and displaying them into map are almost same. So for similarity the description will be given under one title. But in the figure 4.3 they are shown in different process. Each part of location allocation model will be described from the section 4.4 to 4.8 with theirs subparts.

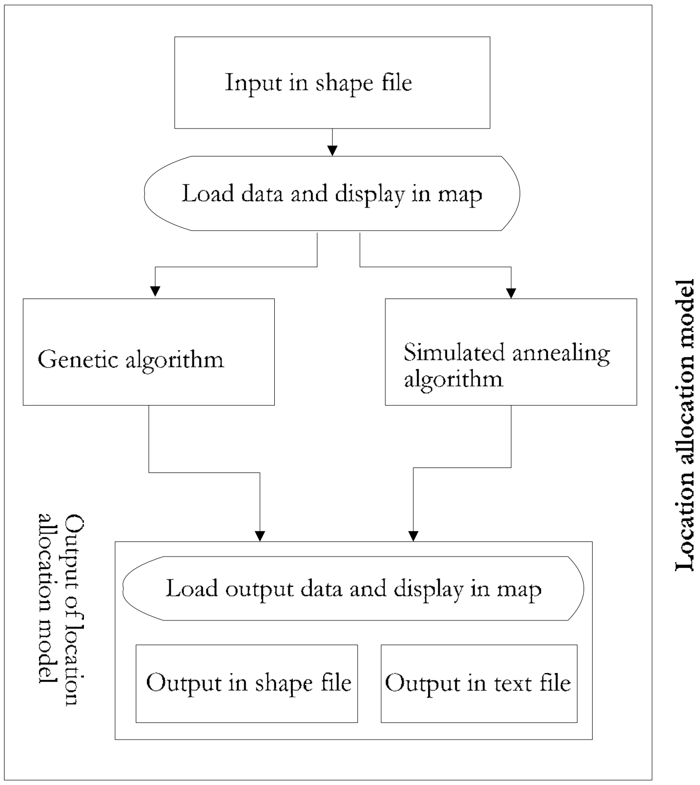


Figure: 4.4 General working procedure of our methodology

4.4. Input into model:

There are two shape files for facility and demand as input. One more shape file may be used to exclude demand point from specific area like park, water-body etc. Some parameters for genetic algorithm or simulated annealing etc are also given as input into location allocation model. If network distance is used instead of Euclidian distance then there will be one more input for network distance which origin destination cost matrix. So, as external file input in our model facility and demand input is obligatory but exclusion area input and cost matrix input are optional according to model output requirement. Origin destination cost matrix will be generated using open source software gvSIG [62].

4.4.1. Facility and demand input of Location allocation model:

This location allocation model can take two types of input for facility. One is fixed input through point shape file another is random input for facility for polygon shape file. Similarly it can handle 3 types of demand input which are fixed input from point shape file, random input from polygon shape file and centroid of area from polygon shape file. For getting optimal location the whole input was can be taken from the combination of following table 4.1.

	Fixed point		Random
Facility	Capacitated, point shape file	Uncapacitated ,point shape file	Uncapacitated ,polygon shape file
Demand	Centroid, polygon shape file	From point shape file	From polygon shape file

Table: 4.1 Input combination of facility and demand

Here, in the row of the facility only one column will be selected for type of facility and similarly only one column will be selected for demand. So, we can create nine combinations between facility and demand from the table 4.1.

Among all these combinations, one combination will be chosen for input. For example, for facility "Capacitated fixed point from point shape file" is chosen and from demand "Random point from polygon shape file" is chosen. All these nine combination are given in the table 4.2.

Facility	Demand
Capacitated fixed point from point shape file	Fixed point From point Shape file
	Fixed point Centroid from polygon Shape file
	Random Point from polygon shape file
Uncapacitated fixed point from point shape file	Fixed point From point Shape file
	Fixed point Centroid from polygon Shape file
	Random Point from polygon shape file
Random Point from polygon shape file	Fixed point From point Shape file
	Fixed point Centroid from polygon Shape file
	Random Point from polygon shape file

Table: 4.2 All combination between facility and demand

If facility is chosen from "Random Point from polygon shape file" the number of potential facility can be taken from user input. For example user may want 100 potential facilities from all polygons in shape file. User has the privilege to give input in this way. Facility can also be loaded from fixed point. In this case type of facility can be considered either capacitated or uncapacitated. In case of "Capacitated fixed point from point shape file" capacity can be chosen from the attribute of the shape file.

Similarly if demand is chosen like "Random Point from polygon shape file" user can also give input for example 1000 random demands. More over, random demand point can be generated from shape file's attribute. Appendix 1 Table has a column "Total4-12" which is total number of children for an age group 4 to 12 for a specific area. Same number of random demand point can be generated for that area. For centroid demand point only one point will be used in per neighbourhood area. The weight for that demand point will be chosen from shape file's attribute like the column "Total4-12" of appendix 1 table.

4.4.2. Exclusion of area for demand and facility point as input into model

When demand point will randomly be generated then there might be some area where demand or facility point generation may not be useful. These areas may be park, field, water-body etc. If objective is to keep demand in the safe zone then industrial area and gas station with some buffer can also be added. However in this research we shall only consider field, park and water-body. To exclude area, polygon shape file of water body, park and play ground will be used into location allocation model.

4.4.3. Other parameters input into model

Number of potential facility, number of optimal facility, number of demand, capacity will be some parameters for location allocation model. There will be some separate parameters as input which will be used in genetic algorithm and simulated annealing respectively. Before running the model these types of parameter needs to be set. Setting exact parameter helps to have optimized result in less time. Specific input parameter of genetic algorithm and simulated annealing will be described in the section 4.6.1 and 4.7.1 respectively.

4.5. Objective function of the model:

Usually in location allocation problem, optimization takes place to minimize distance or transportation time or travel cost etc which has been shown at table 2.1 and table 2.2. Sometimes instead of implementing many objectives only one objective is implemented. If only nearest school is considered as single objective, function f will also try to minimize the function value.

$\min f = w_1 f_1$

$$fitness\ value\ of\ objective\ function = \frac{Very\ big\ constant}{\min f} \qquad \text{..........................Equation 4-1}$$

Here, the lower the value of function f the higher the fitness value becomes.

4.5.1. Single objective using nearest school

Mixed integer programming minimizes or maximizes linear function with consideration to constraint. ReVelle and Swain in 1970 first proposed an approach of mixed integer programming to solve P median problem [63]. Their formulation allows all nodes of both potential facility and demand for P median problem which was exact method. Then mixed integer programming was made hybrid with genetic algorithm by Correa *et al.* [16], Medaglia *et al.* [31], Zhao *et al.*[40] Neema *et al.* [15]. In our model we shall also follow genetic algorithm with mixed integer programming approach. The objective to assign some users (demand) to nearest school, those demands will use optimal distance from facility to demand.

Considering the demand for nearest school the objective function will be as follows:

$$minimize\ f_1 = \sum_{i=1}^{p} \sum_{j=1}^{n} d_{ij} w_j$$

................................Equation 4-2

$$C_i \geq A_i \qquad \text{................................Equation 4-3}$$

Where,

P= number of facility,

n= total number of demand points,

x_i, y_i =Location of facility point i,

x_j, y_j =Location of demand point j,

$d_{i,j}$ =Distance between point i and point j (distance can be either Euclidian or network based)

w_j = Weight of demand at point j,

C_i = The maximum capacity of i facility,

A_i = The allocated demand on i facility.

4.5.2. Single objective using nearest school and demand distributions

In calculation of minimum distance there will be some other influencing criteria like capacity, allocation and different types of demand etc. So far we have considered that demand from users/ students are only one type which is minimum distance. But there could be some other type of demand from users or students. Some of the students may want to study in school (facility) type

'A' (e.g. based on religious belief), some want type 'B' (e.g. based on special type educational method) and some want type 'C' (e.g. based on textbook). Beside these, some student may not get admission in nearest school and the demand of getting typed school of own choice (facility) due to capacity of that school. Then they will be assigned randomly to any other school. The following equation shows all the demands.

All Demand = Demand using optimal distance (for nearest school)
+Demand using type of facility (for types like textbook, religion, culture etc. of school)
+Demand assigned randomly when capacity of nearest school or capacity of a chosen type school is full.

....Equation 4-4

Considering the demand for nearest school the objective function will be as follows:

Now if divide total demand n according to equation no 4.2 then we have equation 4.2 as follows:

All Demand = n-m-s, Demand using optimal distance (for nearest school)
+m, Demand using type of facility (for types like textbook, religion, culture etc. of school)
+s, Demand assigned randomly when capacity of nearest school or capacity of a chosen type school is full).

In this case objective function from equation 4.2 becomes as follows:

$$f_2 = minimize \sum_{i=1}^{p} \sum_{j=1}^{n-m-s} d_{ij} w_j + \sum_{i=1}^{p} \sum_{j=1}^{m} d_{ij} w_j + \sum_{i=1}^{p} \sum_{j=1}^{s} d_{ij} w_j$$

4.6. Genetic algorithm in the model

Genetic algorithm, its terms and working principle are described in chapter two. For genetic algorithm chromosome design is important. Similarly designing proper fitness function is also important. The input parameter for genetic algorithm from the model should be used as input into genetic algorithm.

4.6.1. Input parameter of genetic algorithm

There are five input parameters for genetic algorithm which are mutation, crossover, population size, generation and consequent repeated number as stopping criteria. In each generation a population from a given size is created. Best chromosomes are chosen from population for crossover. Crossover is executed according to given crossover rate. Some chromosomes will be mutated according to mutation rate. When genetic algorithm will be stopped will depend on two parameters. One is stopping criteria another is generation number. If a fixed outcome is produced from genetic algorithm for some 'n' times (n is a positive integer number smaller than generation number) then genetic algorithm will stop. Otherwise, genetic algorithm will run up to a specific generation number which is taken as input. Stopping criteria works same as stack. When genetic algorithm starts to produce same fitness value it is being stored into stack. If the element number of stack is same to the number of stopping criteria, genetic algorithm will be stopped.

4.6.2. Incorporating facility and demand points as GIS data:

Facility and demand points are provided as GIS data. Integration of location allocation problem and genetic algorithm is done by using facility and demand points in our location allocation model. Optimal facility location is used to create chromosome. Both demand and optimal facility location points are used to calculate the distance in fitness function of genetic algorithm. Instead of using original point only index is used as chromosome which is also shown in figure 4.4. Individuals or solutions (optimal facility location) will be different subsets of chromosome. Total number of gene in each individual will be same as the number of optimal facility location. Figure 4.4 shows an example of how demands of 100 points, potential facility and chromosome of 8 points and individual of 5 points out of 8 points use index. All index holds corresponding points in the corresponding array which have corresponding x and y coordinate.

The individual in figure 4.4 has 5 genes which are 8, 2, 7, 3 and 4. Individual which is shown in dark colour is only used in calculation of fitness value and hence fixes the fitness of that chromosome.

1	2	3	4	5	6	7	..	..	..	..	100	Demand Points
1	2	3	4	5	6	7	8					Potential Facility Points
8	2	7	3	4	6	1	5					Also Chromosome
8	2	7	3	4	6	1	5					Individual / Solution

Figure: 4.5 Demand, potential facility points, chromosoe, individual representation

4.6.3. Fitness calculation from chromosome

Fitness value of the chromosome will be calculated according to minimum distance between individual and demand. For each demand a gene among all genes from individual which has minimum distance will be selected. Same process will be done for all demand to get total minimum distance. This process is shown in figure **Error! Reference source not found.**5. The figure shows for 100 demand points how minimum distance is calculated between demand and individual. For example, the minimum distance between demand 1 and a gene among genes 8, 2, 7, 3 and 4 needs to be deduced firstly. This process will be continued up to demand no. 100 to measure total minimum distance.

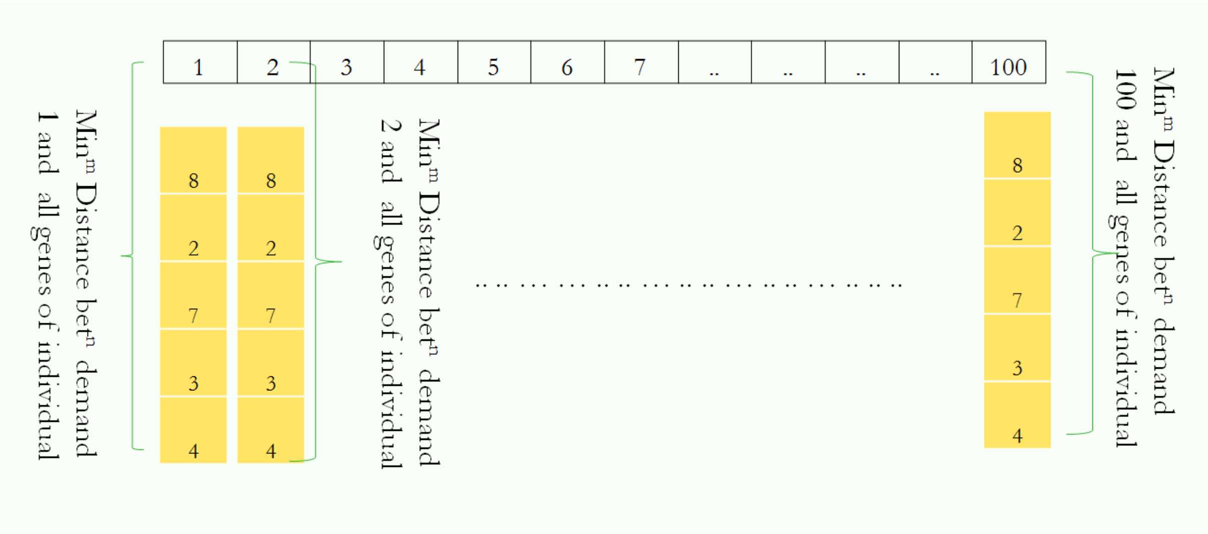


Figure: 4.6 Minimum distance between each demand and all genes in individual

Each chromosome will have its own fitness value through calculation shown in the above figure. In genetic algorithm population, best chromosome will be selected in each generation and will be passed to next generation according applied elitism.

4.7. Simulated annealing in the model

Unlike genetic algorithm, simulated annealing is simple to implement. It has also fewer input parameters than genetic algorithm. Its ‘current order’ has almost same structure of genetic algorithm’s chromosome. But unlike genetic algorithm’s chromosome its ‘current order’ does not take participate in recombination and reproduction. Rather next solution generates simply from random generation from ‘current order’ solution.

4.7.1. Input parameter in simulated annealing

There are three types of input for simulated annealing algorithm which are temperature, cooling rate and absolute temperature. The parameter temperature is the initial (the highest) temperature. This temperature will be slowed down gradually at cooling rate which is another parameter. Then this temperature will be stopped when it will reach at absolute temperature. Absolute temperature is another parameter. All these three parameters actually determine how much iteration will be going on without stopping criteria. If stopping criteria is applied then it will work in a similar way of genetic algorithm.

4.7.2. Current order design:

Initially current order will be designed randomly where each index of potential location will appear only once. In iteration of simulated annealing current order will be changed randomly at one or more than one position of that to generate next order.

4.7.3. Fitness value calculation in each order of simulated annealing

Fitness value of simulated annealing is similar to the fitness value of genetic algorithm. Instead of chromosome of genetic algorithm in simulated annealing current order is used to determine the fitness value. To calculate fitness current order will be traversed according to the number of optimal location. The higher the fitness value the better the current location is. When simulated annealing will stop, it is expected that optimal location with the highest fitness value will be achieved.

4.8. Tools for map display and genetic algorithm-simulated annealing.

To display potential facility and demand in map an open source tool SharpMap [64] will be used where source code for that tool is available in internet. The output of genetic algorithm and simulated annealing as optimal location will also be displayed with the help of same tool. For genetic algorithm a genetic algorithm library [50] will be used and modified to fit for location allocation. Similarly for simulated annealing a simulated annealing library [65] will be used and made fit for location allocation algorithm. The shape file will be created for optimal location with another open source tool which is NetTopologySuite [66]. So in the location allocation model different open source tools are used which are given in the figure below and the model is developed in c sharp.

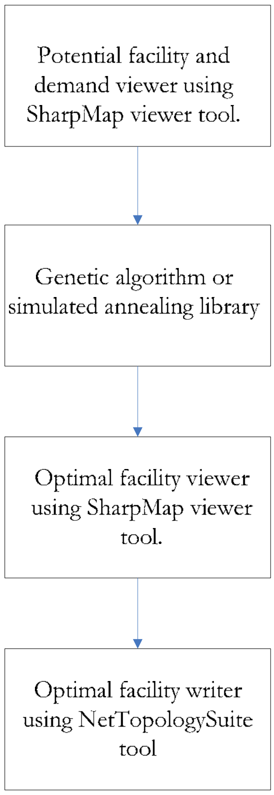


Figure: 4.7 Input and output display in map with different tool.

4.9. Output of the model:

In the location allocation model, there will be two type of output. One output will create optimal location in shape file so that this can be used by other GIS software for further process and analysis. Other type of output is by text file. It will store all chromosomes in population by fitness sorted order as genetic algorithm output. For simulated annealing output it will store current order with simulated annealing fitness value. It will also store all input parameters and execution date etc for both genetic algorithm and simulated annealing.

The location allocation model will solve several types of location allocation problem like to select optimal location for 'P' number location out of 'N' location, to relocate facility from existing ones, to find extra facility considering existing ones and to satisfy demand coverage problem which has been shown in the following figure 4.8

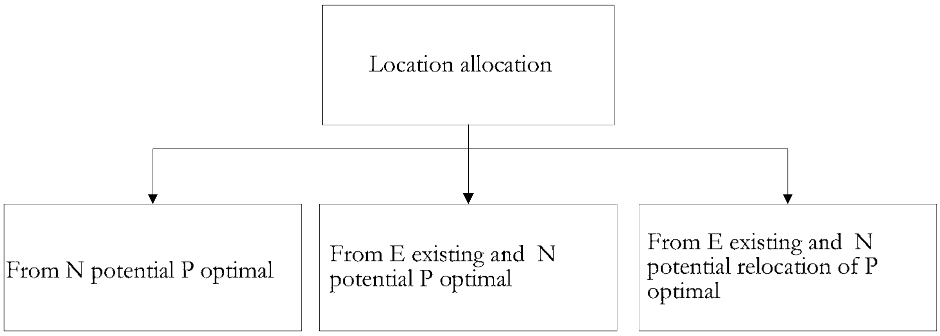


Figure: 4.8 Output of the location allocation model

How each problem of figure 4.8 will be solved, will be discussed in the next sub-sections elaborately.

4.9.1. P optimal from N potential location

Selecting P optimal from N potential location based on fitness value is classical p median location allocation problem. It is solved in ARCGIS location analyst as problem of maximizing impedance [1] and in LOLA software Q median and N facility problem [47]. However, the lacking of ARCGIS and LOLA software in this problem is they can not generate a solution with random data. Both of these software only loads previously fixed points as demand and facility point but do not use random generation of facility or demand data within same area. In our prototype of model, we have successfully used random data which will help to understand the scenario for location allocation model.

Here we have loaded fixed locations of facility as potential location. The number of potential location from shape file is 39. So, N=39. Hence chromosome length is also 39. We set the number of optimal location 25(=P) to choose out of 39. We need to generate 1600 demand points randomly over each neighbourhood in Enschede. So, we set the demand point for Enschede 600. Like random generation of demand point, potential location can also be generated randomly. Then we need to set the parameter for genetic algorithm. We choose the crossover rate 0.93 which means there is 93% chance for chromosome cross-over. We set the mutation rate .07 that implies only 7% chance for mutation of chromosome.

Both demand and facility data (points) are incorporated according the method described in section 4.6.2. In the following figure we have shown the outcome from location allocation model through spider web display.

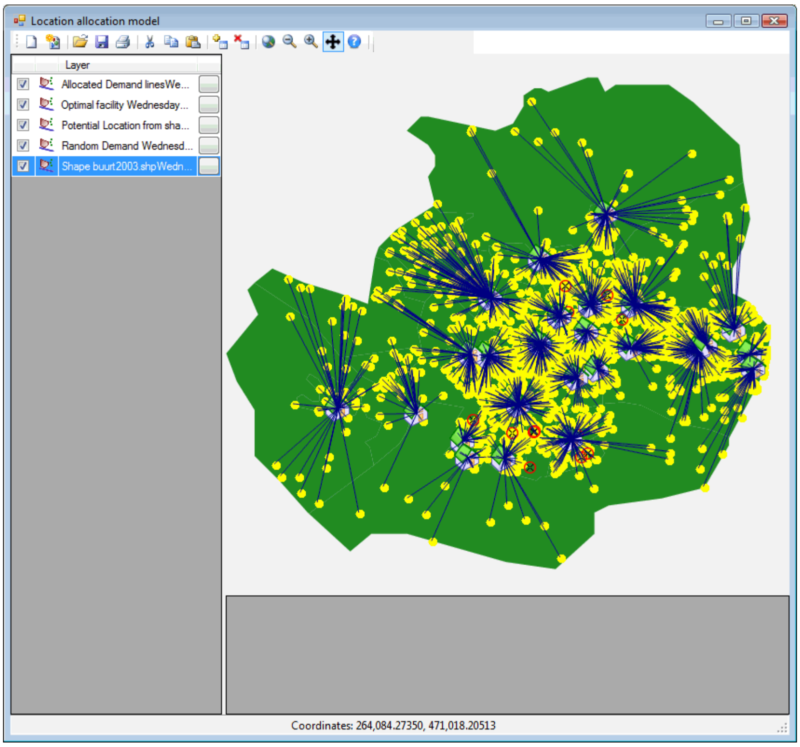


Figure: 4.9 Location allocation model with integration of GIS

In the above figure 4.9 the spatial distribution of potential location and demand is shown in red and yellow filled circle respectively. Optimal locations are shown with house symbol. Allocation of demand to the corresponding facility is shown with straight line. Total allocation creates the spider web display in the map.

The output graph for fitness value against population in genetic algorithm is taken as follows:

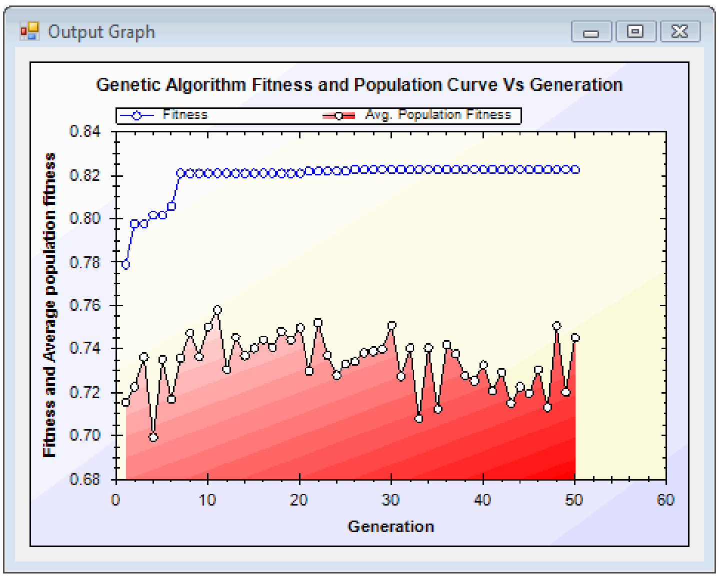


Figure: 4.10 Fitness versus generation

In the above figure 4.10 the fitness of population and best fitness in each generation is depicted. There is lots of zigzag in average population's fitness which implies the variety of individuals due to crossover and mutation. The best fitness in the population is always higher than the average fitness of the population. Local elitism is applied. A global search is applied among all generation's fitness value to get the highest fitness value. Fitness value is converging before there is no significant change for a number of generations.

Similarly different combination from table 4.2 can also be shown as output of our model. Due to reduce redundancy we shall not discuss all the combinations.

4.9.2. P optimal from E existing and N new potential location

In this type of problem, there are E number of existing facility and N number of new potential facility. Existing facility can be loaded from fixed point and new potential can be loaded from random point generation over area of Enschede (polygon type shape file). In total, there is E+N number potential facility from where we have to select P number of optimal facility.

Since all existing facilities need to be considered, we skipped the existing facility in the selection of optimal facility. As optimal facility existing facilities are already considered. Rather we selected P-E optimal facility from N number of optimal facility.

So, $P-E \leq N$

Now, selecting P-E optimal location from N location becomes identical with selecting P location out of N location. But when we need to allocate demand in existing facility and P-E newly selected facility. In this problem, capacity of each facility has to be allowed in time demand allocation.

We have to load fixed locations of facility as potential location. The number of new random location from shape file is 45. So, N=45. Hence chromosome length is also 45. We need to create 31 new locations in total there will be 70 optimal locations including existing and new. We need to generate 2500 demand points randomly over each neighbourhood in Enschede. So, we set 2500 demand points over Enschede. Like random generation of demand point, potential location can also be generated randomly. Then we need to set the parameter for genetic algorithm. We choose the crossover rate 0.90 which means there is 90% chance for chromosome cross-over. We set the mutation rate 0.20 that implies only 20% chance for mutation of chromosome.

In the following figure we can see the spider web of location allocation from existing and new location. The spatial distribution of fixed potential location in red, random potential facilities are in blue and demand is shown by yellow filled circle respectively. Optimal locations are shown with house symbol. Allocation of demand to the corresponding facility is shown with straight line. Total allocation creates the spider web display in the map.

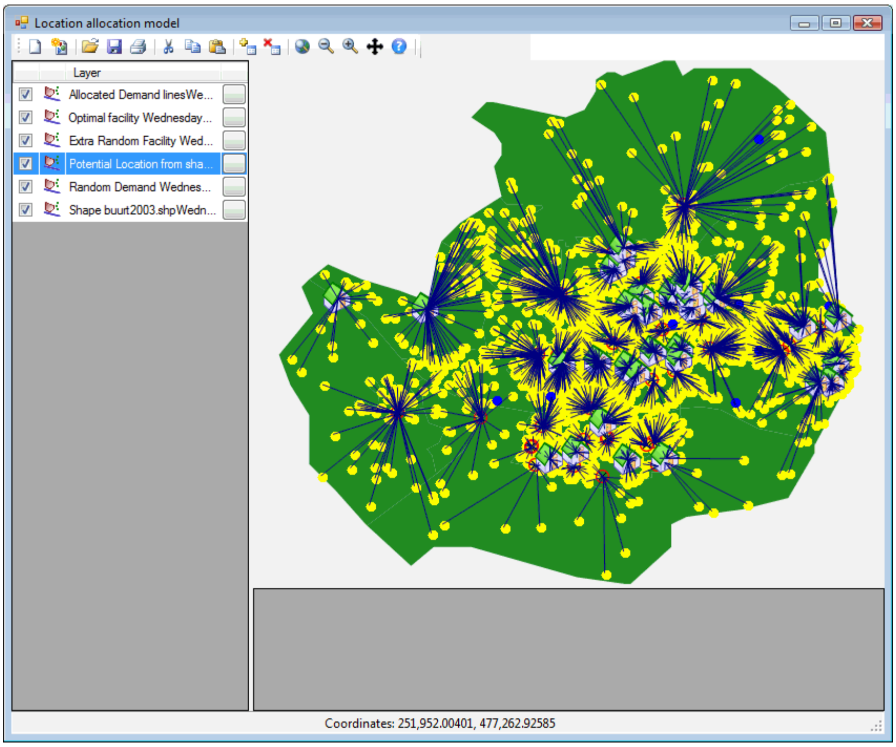


Figure: 4.11 Location allocation model with integration of GIS from existing and new location

In the following figure the fitness graph is shown against population. The fitness of population and best fitness in each generation is also depicted. The best fitness in the population is always higher than the average fitness of the population.

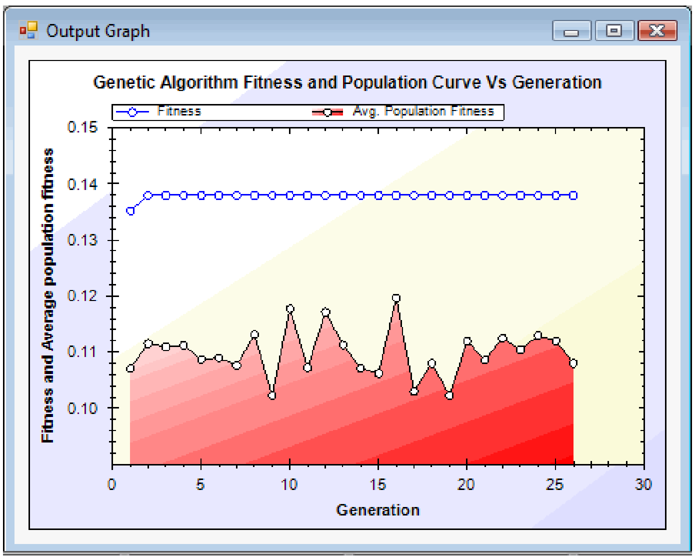


Figure: 4.12 Fitness versus generation

4.9.3. P optimal relocation from E existing and N new potential location

In this type of problem, some facilities need to be relocated from existing facility into new potential locations. There are E number of existing facility and N number of potential facility. We need to select P number of optimal facility. Unlike previous type of problem where all existing facilities were selected for allocation, some of the existing facilities will not be selected here. In stead of these nonselected existing facilities some other facilities will be selected from potential such that P optimal facilities are selected from both existing facilities and potential facilities.

So, $P \leq E+N$

Here, chromosome length is E+N. Unlike previous two problems, Chromosome of genetic algorithm is created from existing location and new potential location. In the following figure we shall show how to set the parameter to create random facility for new location and demand data and load fixed facility. Here we have loaded existing fixed locations of facility as some of potential locations which is 39. So, E =39. The number of new random location from shape file is 45 as the rest of the potential location. So, N=45. Hence chromosome length is 84(39+45). We need to create 42 optimal locations where there will be 84 potential locations including existing and new. Some of the existing facilities will be closed and relocated. We need to generate 2000 demand points randomly over each neighbourhood in Enschede. So, we set the demand point for Enschede 2500. Like random generation of demand point, potential location can also be generated randomly. Then we need to set the parameter for genetic algorithm. We choose the crossover rate 0.93 which means there is 93% chance for chromosome cross-over. We set the mutation rate 0.20 that implies only 20% chance for mutation of chromosome.

In the following figure we can see the spider web of location allocation model. The spatial distribution of fixed potential location in red, random potential facilities are in blue and demand is shown by yellow filled circle. Optimal locations are shown with house symbol. Allocation of demand to the corresponding facility is shown with straight line.

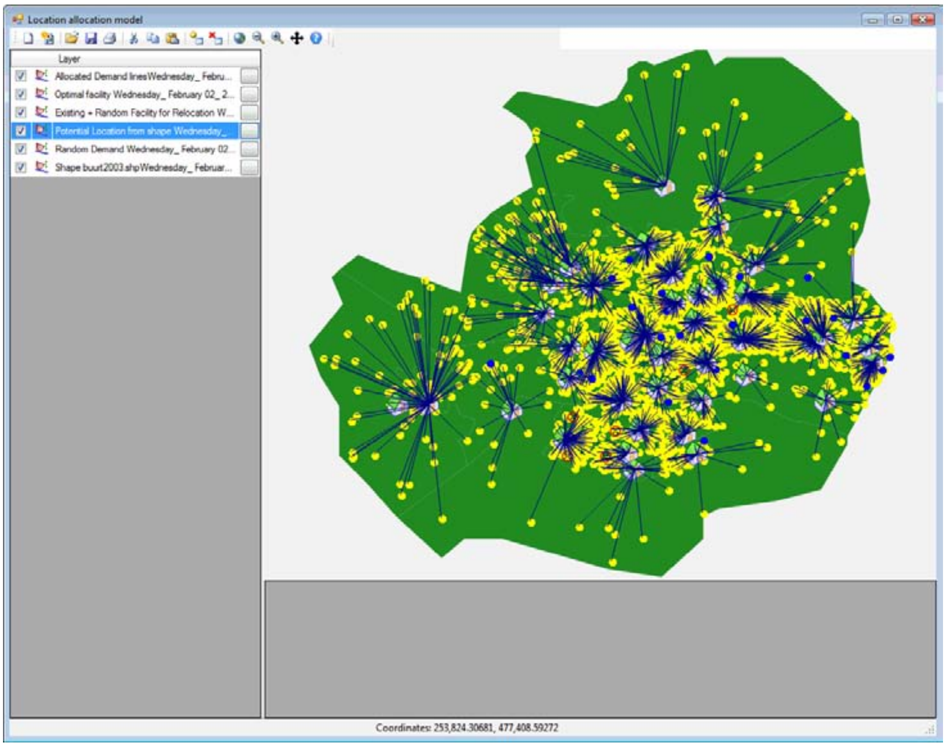


Figure: 4.13 Location allocation model with integration of GIS for relocation

In the following figure the fitness graph is shown against population.

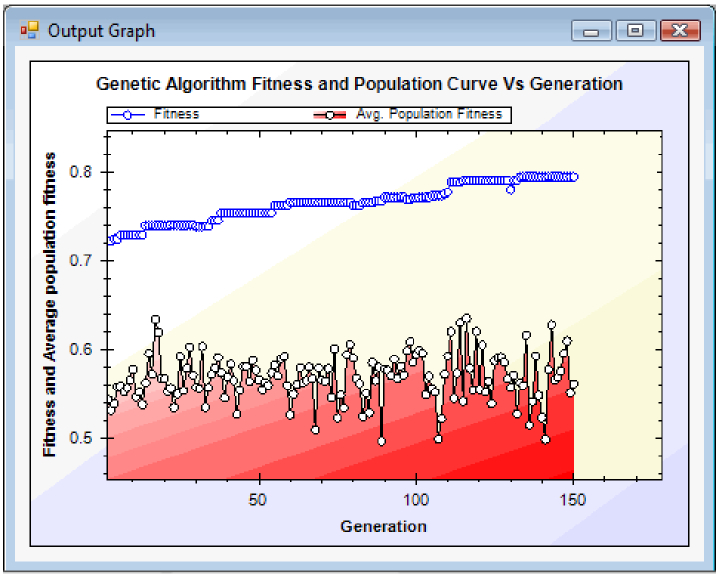


Figure: 4.14 Fitness versus generation for relocation

4.10. Summary

In this chapter we have discussed elaborately about methodology and implementation. Input types for model, different issues with genetic algorithm for model implementation and similarly simulated annealing for model implementation were discussed in good detail. Finally we have discussed about implementation of three different types model for location allocation.

5. CASE STUDY -RESULT & DISCUSSION

5.1. Introduction:

In this chapter we shall implement case study by location allocation problem for primary schools in Enschede. In one of our case studies, we shall try to allocate demand to nearest school considering capacity. In another case study we shall consider not only nearest distance but also user preference in school type to allocate demand. But first we shall adjust the relevant parameters for the model. After selecting suitable parameter we shall use these parameters into our case study. We shall also analyze and compare the results based on Euclidian and road network distance. A comparison between simulated annealing and genetic algorithm on fitness value will also be conducted. At the end of this chapter we shall make a discussion of the case study result.

5.2. Case study:

For this research, students are considered for primary school from an age from 4 to 12 in Enschede city. From the year 1996, the number of primary school students was increasing up to 2001. From 2001 to 2010, primary school students were between 15400 and 15600 range. This information is taken from the municipality website of Enschede [61]. The number of primary school students from January 1996 to October 2010 is shown in the following figure.

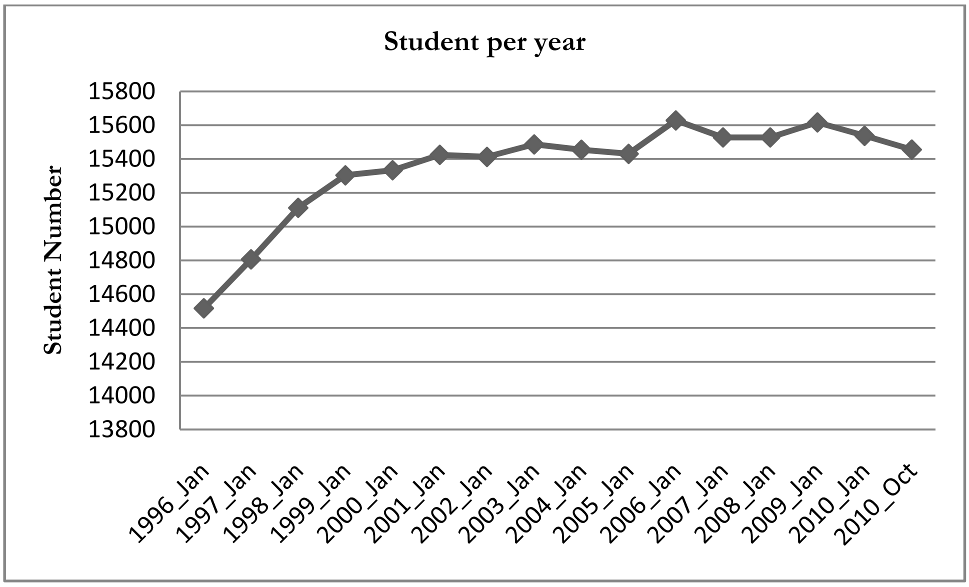


Figure: 5.1 Student per year from Enschede municipality website

From this situation we can think of three possibilities of primary school student number in the year 2012. First possibility, the number may remain stable like the years in last decade. Another possibility, the number may reduce due to huge economic collapse or any other catastrophe in Enschede or migration from Enschede. And third possibility is the increase of students due to migration into Enschede on account of huge economic development. The second and third possibilities may not have much probability but we shall consider all three possibilities in our model. This case study will be accomplished based on nearest distance and both nearest distance and user preference in section 5.4 to 5.5. But before case study we shall adjust the parameters of genetic algorithm and simulated annealing in the following section.

5.3. Parameter settings and selection of algorithm from test case scenario:

For getting good result from implementation we need to set up proper parameter for genetic algorithm and simulated annealing in our model. Genetic algorithm parameters such as crossover, mutation, population size need to be fine tuned. Their influences over solution need to be tested by a test case scenario. Similarly for simulated annealing initial temperature need to be fixed to achieve the best result from it. As a by product from this fine tuning we can also compare between genetic algorithm and simulated annealing. After comparing the result we shall select an algorithm for the implementation of the case studies.

Fine tuning of parameters will be done over 2003 demand data of Enschede with a simple test case scenario. In this test case scenario we are going to select best 39 school location out of 144 potential school locations. Potential school locations are created by 1 KM grid space in Enschede using ARCGIS which is shown in figure 5.2.

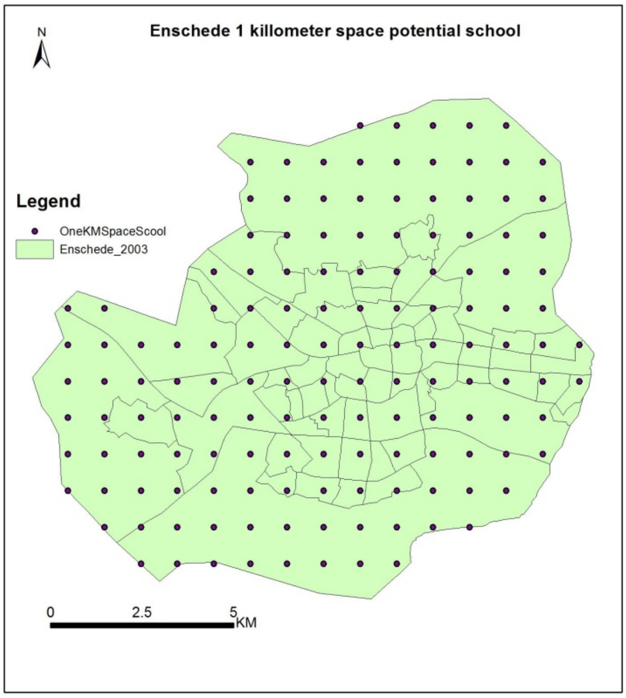


Figure: 5.2 One killomter potential school in Enschede

According to 2003 data, capacity is 6467 and demand is 15487. But this demand actually summation of total student from public school and private school. Since we are only dealing with students from public school, we shall use factor between capacity and demand such that capacity is greater than demand of public school students.

$capacity \geq factor \times total\ demand$ ……………………………….Equation 5-1

So, $factor \leq \frac{capacity}{total\ demand}$

$factor \approx .40$

We shall generate random demand of students which will be used as same demand points for all implementations in fine tuning parameters. 40% of 15487 create 6219 demand points. This test case scenario will also simulate capacitated P-median problem in our model. Figure 5.3 shows 6219 random demand points generated from our model. These random points are generated from the student number in each neighbourhood multiplied by the factor given found from equation 5-1.

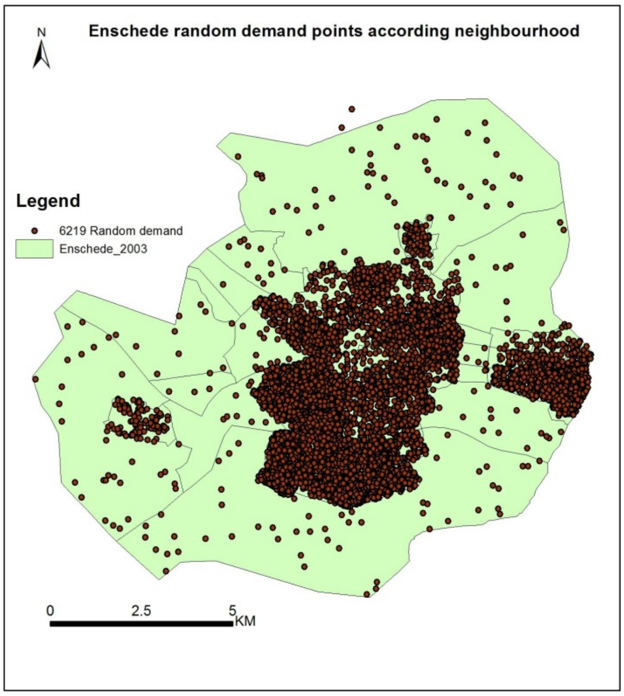


Figure: 5.3 Generating 6219 random demand points

5.3.1. Genetic algorithm parameter analysis

Jaramillo mentioned that mutation rate should be used with low value [52]. A crossover rate 0.98 and mutation rate 0.01 was used in [3]. So, we have used a crossover rate within .90 and .98 and a mutation rate within 0.05, 0.10 and 0.20. To reduce the number of combinations among all these types of parameter (crossover, mutation and population size) initially we have fixed population size. We have produced six results from the combination of these two types parameter using a fixed population size of 10. Table 5.1 shows that the best crossover and mutation rate among six combinations is 0.98 and 0.2. Here fitness value is computed according to the equation no 4.1 and 4.2 and Euclidian distance is used.

Crossover rate	Mutation rate	Population size	Fitness
0.9	0.05	10	0.104
0.9	0.1	10	0.115
0.9	0.2	10	0.105
0.98	0.05	10	0.123
0.98	0.1	10	0.135
0.98	**0.2**	10	**0.138**

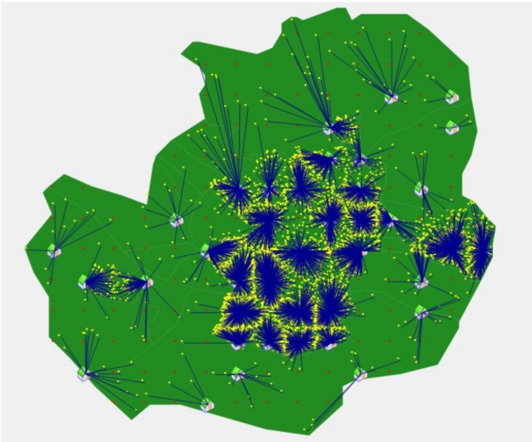
Table: 5.1 Selection of crossover and mutation rate in genetic algorithm using Euclidian distance

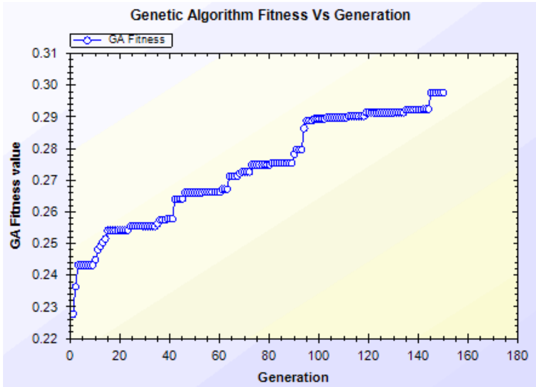
Li and Yeh suggested a population size 10 yields good result [3]. Gong *et al.* also used population size 10 [29]. But we have used 10, 30 and 50 as population size in our test case and let the solution run up to 150 generations. Only top fitness value is compared among three population size. We have taken the best parameter values of mutation and crossover from table 5.1 to select best population size which is shown in table 5.2.

Crossover rate	Mutation rate	Population size	Fitness
0.98	0.2	10	0.138
0.98	0.2	30	0.15
0.98	**0.2**	**50**	**0.235**

Table: 5.2 Selection of population size in genetic algorithm using Euclidian distance

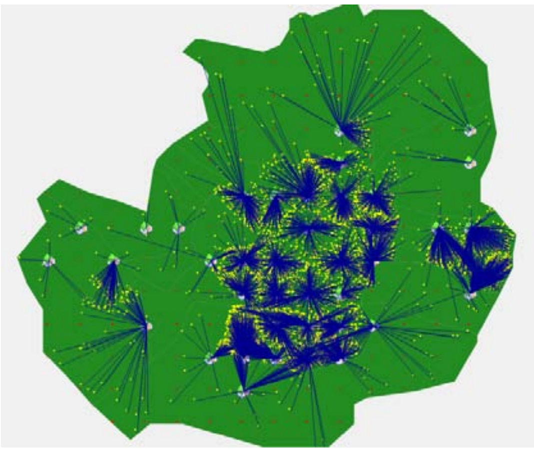
Figure 5.4 and 5.5 shows the output from best parameter settings from our model using Euclidian and road network respectively. In our model, red circle represents potential schools and yellow circle for demands. Green lines are drawn to show allocation and small house symbol shows optimal location. The output graph shows the convergence of fitness value. Fitness value is converging with the increase of generation. Since we have let the solution run up to 150 generation fitness curve shows top fitness values up to that generation. Figure 5.6 shows optimal location from our model in ARCGIS.

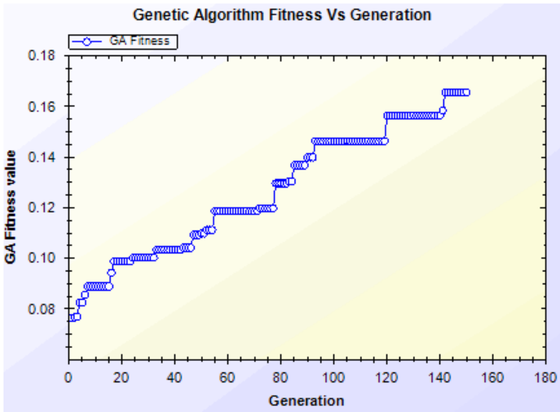




(a) **(b)**

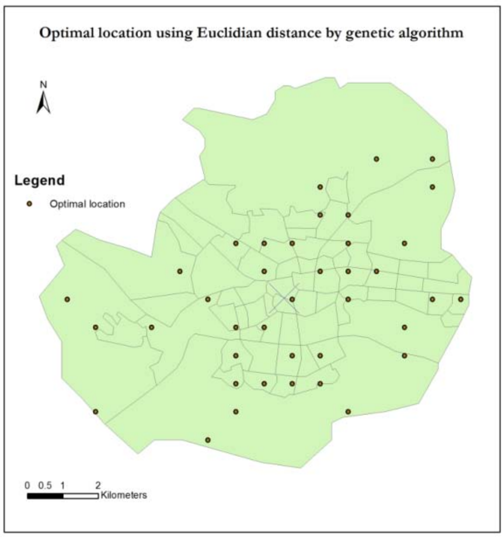
Figure: 5.4(a) Location allocation (b) Fitness output-using Euclidian distance by genetic algorithm

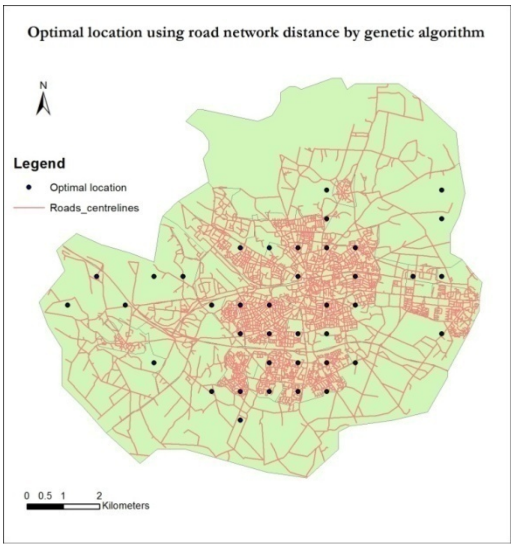




(a) **(b)**

Figure: 5.5(a) Location allocation (b) fitness output- using road network by genetic algorithm





(a) **(b)**

Figure: 5.6(a) Euclidian (b) Road network-Optimal location by genetic algorithm viewed in ARCGIS

From table 5.1 and 5.2 we have got the best parameter for crossover, mutation and population size. These values are determined for Euclidian distance. The top three fitness value's parameter from table 5.1 and 5.2 are used to select the best parameter setting for road network based distance. We have created origin destination cost matrix using gvSIG software [62]. Potential locations and demands are same as for Euclidian distance according to figure 5.2 and 5.3. Table 5.3 shows the fitness values using road network distance for the best three parameter settings from table 5.1 and 5.2.

Crossover rate	Mutation rate	Population size	Fitness
0.98	0.2	10	0.1
0.98	0.2	30	0.111
0.98	**0.2**	**50**	**0.139**

Table: 5.3 Selection of three parameters in genetic algorithm using road network distance

5.3.2. Simulated annealing parameter analysis

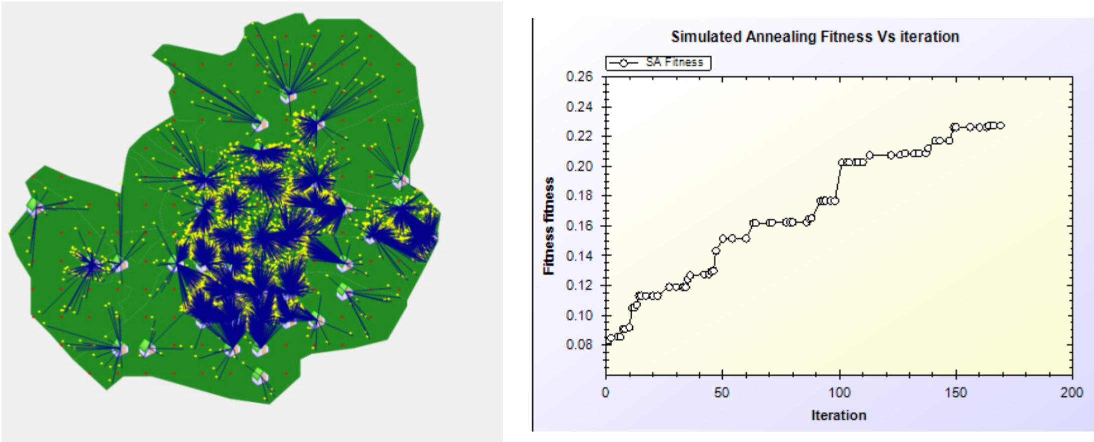
In different researches of simulated annealing different initial temperatures were used. Murray and Church used 40, 60 for different P-median problems [4]. In our model we shall check with initial parameter near to Murray and Church like 50 and also 100, 200 and 300. In these cases cooling or freezing rate will be 0.9 according to [3] and absolute temperature will be 0.001 to stop iteration. Table 5.4 shows best initial temperature according to fitness is 100 for Euclidian distance. The best fitness is 0.204. Table 5.5 shows 300 also as best initial temperature for road network distance. The best fitness for road network is 0.145.

Initial temperature	Fitness
50	0.193
100	**0.204**
200	0.196
300	0.201

Table: 5.4 Initial temperature for simulated annealing with fitness using Euclidian distance

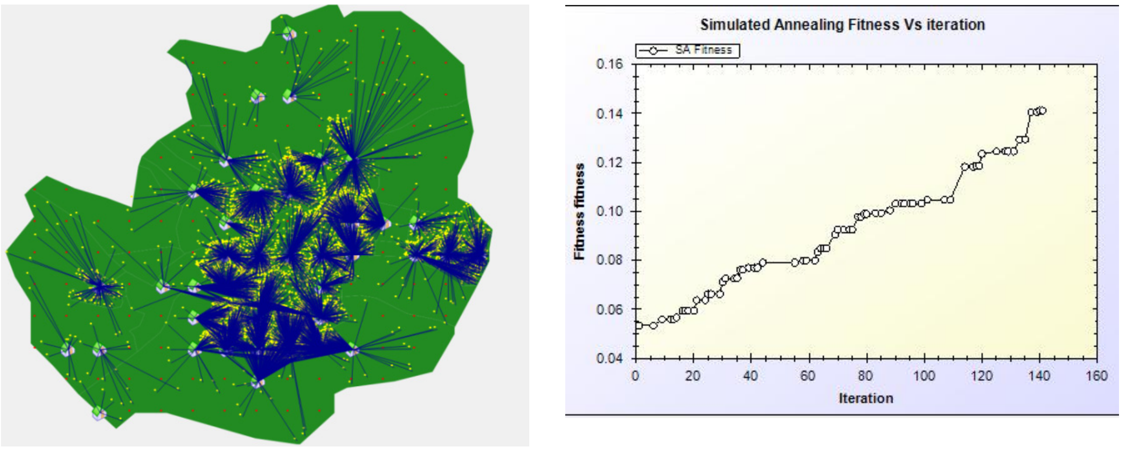
Initial temperature	Fitness
50	0.128
100	0.141
200	0.118
300	**0.145**

Table: 5.5 Initial temperature for simulated annealing with fitness using road network distance



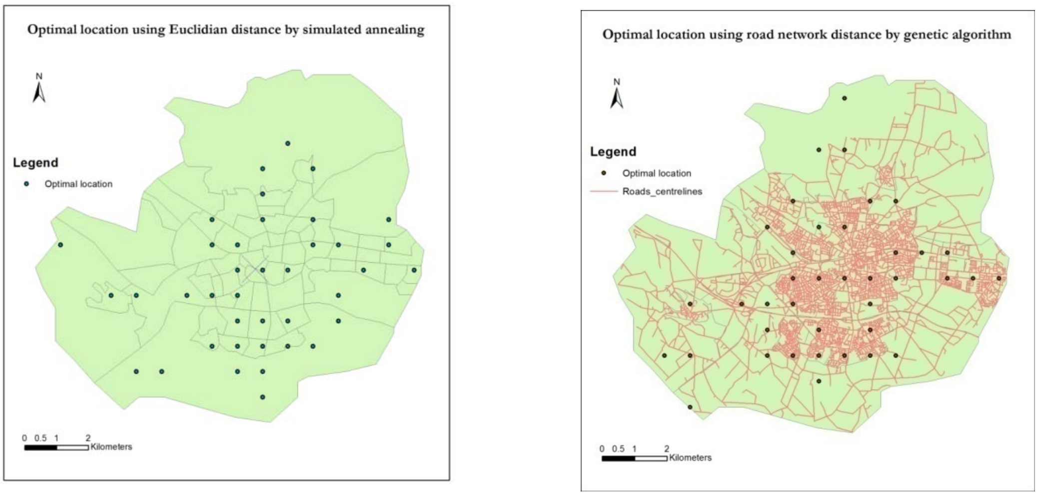
(a) **(b)**

Figure: 5.7(a) Location allocation (b) Fitness output- using Euclidian distance by simulated annealing



(a) **(b)**

Figure: 5.8(a) Location allocation (b) Fitness output - using road network distance by simulated annealing



(a) **(b)**

Figure: 5.9(a) Euclidian (b) Road network-Optimal location by simulated annealing viewed in ARCGIS

The figure 5.7 and 5.8 shows location allocation in spider web and output graph for Euclidian and road network distance respectively. Figure 5.9 delineate optimal location from our model in ARCGIS. Simulated annealing produces these results in around 150 iterations.

5.4. Discussion about distance and selection of algorithm:

5.4.1. Road network and Euclidian

We have seen from the section 5.3 that genetic algorithm can be used using Euclidian and road network distance. But both types of distance do not bring similar result. The fitness value of Euclidian distance is 0.235. The fitness value of road network is 0.139. So the fitness value using Euclidian distance is higher than that of road network. According to the equation 4.1 if the fitness value is higher then optimal distance is minimum. So, for road network we get minimum optimal distance than road network which is also realistic. In reality when student will go along road network, in most cases it will not be straight line.

Similarly simulated annealing also solves location allocation using Euclidian and road network. The fitness value of simulated annealing using Euclidian also found higher than using road network according to table 5.4 and 5.5.

So, in Enschede, student's route to school using straight line distance will be smaller than the route using road network. Our result validates this fact.

5.4.2. Genetic algorithm and simulated annealing

Fitness value for genetic algorithm is higher than simulated annealing using Euclidian distance according table 5.2 and 5.4. Using road network distance the fitness value between both algorithms is almost same. They differ with 0.006. Since simulated annealing makes random solution first and then check whether it is better or not so, within limited iteration it may not be able to produce good fitness value. For the initial temperature the highest fitness was 100 in table 5.4 while in table 5.5 the highest fitness value was 300. This happens due to the dependency for the randomness of the solution and without having any driving factor of solution towards optimal. On the other hand genetic algorithm uses elite individual to drive the solution towards optimal from the first generation. So in limited generation it produces good fitness value. This also validates the result by Arostegui [6].

5.5. Case study 1, nearest distance

From the previous section, we have found that genetic algorithm performs better to converge quickly towards optima value. Since we shall use Euclidian distance for all case studies we shall use genetic algorithm as solution method. As we discussed in section 5.2 about two case studies of using nearest distance and using both nearest distance and user preference, in the first case study we shall consider nearest distance.

5.5.1. First possibility: stable number of primary school students:

Public primary schools in Enschede had a capacity of 6467 in 2003. Figure 5.1 shows that in October, 2010 in Enschede there was almost same number of student of 2003. Since primary school student number is stable we can think of relocating some schools so that nearest distance becomes more optimal with consideration to 2003 allocation. To do so, there will be closing of some schools and opening some new schools.

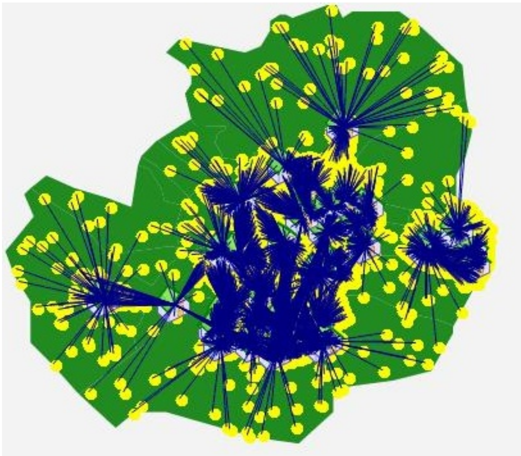
According to 2003 data, capacity is 6467 and demand is 15487. But this demand actually summation of total student from public school and private school. Since we are only dealing with students from public school, we shall use factor between capacity and demand such that capacity is greater than demand of public school students. So according to equation 5-1 factor becomes as follows:

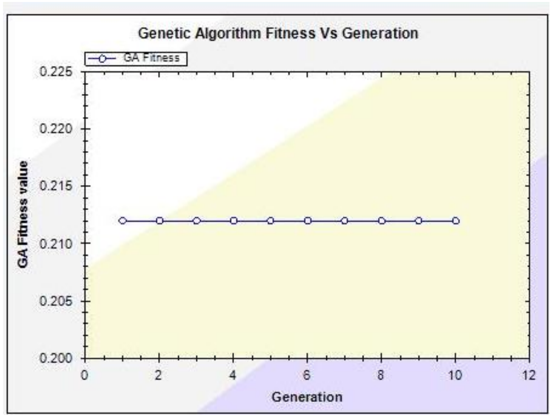
$$factor \leq \frac{capacity}{total\ demand}$$

$$factor \approx .40$$

Firstly we have found the average distance of each allocated demand according to school. With addition to this, we have also found total distance of allocated demand to each school using this factor for data 2003. This information is shown in appendix 2. Average distance per school for allocated student will also be found in that table. Total optimal distance for all allocated students is 4716.9 kilometres. Total student number is 6219 using factor 0.4. Average distance for any student is 758 meter.

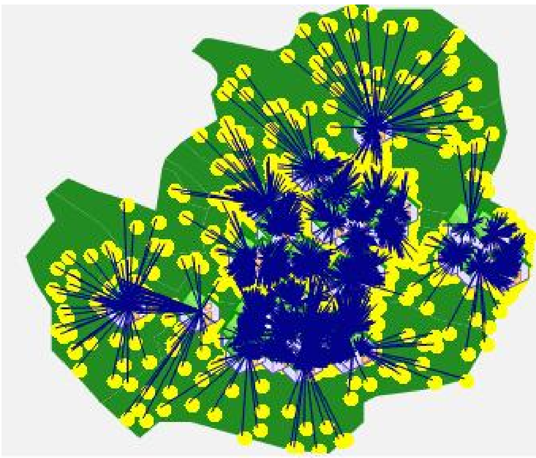
Now we shall allow maximum 5 school to relocate and in total we shall keep 39 school such that student can get more optimal distance. 5 schools and all demands will be generated randomly over Enschede neighbourhood. Demands will be generated with factor proportion of data taken from municipality website. Demand and school locations have been avoided from water body in Enschede using area exclusion option. Chromosome crossover and mutation rate has been fixed according to previous result which are 0.98 and 0.20 respectively. It will run up to 150 generation. The model allocates all students in their optimal location using spider web allocation over GIS data. Spider web for location allocation and the output graph is shown in the figure 5.10 (a) and (b) respectively.

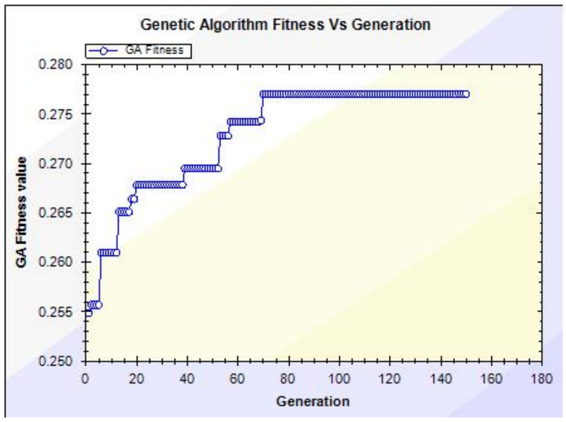




(a) **(b)**

Figure: 5.10(a) Location allocation (b) Fitness output - in the model before relocation



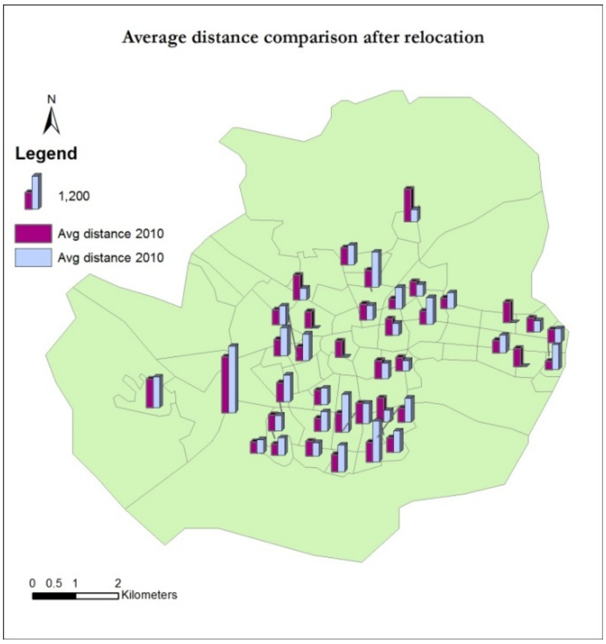


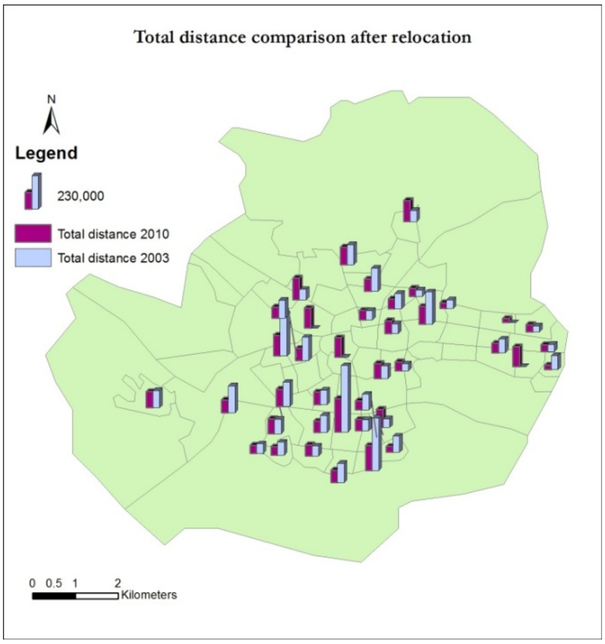
(a) **(b)**

Figure: 5.11(a) Location allocation (b) Fitness output - in the model after relocation

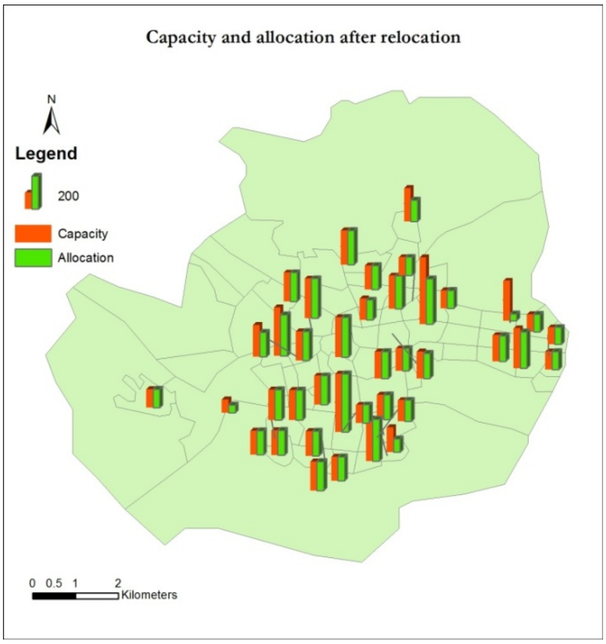
After relocation total optimal distance for all allocated student is 3609634 metres. So, optimal total distance is reduced around **23%**. Total student number is 6219 with factor 0.4. Again, after relocation average distance is 580 meter. So, average distance for any student is also reduced around **23%.** All data after relocation is also shown in appendix 3. Spider web for location allocation after relocation and output graph is shown in figure 5.11 (a) and (b) respecctively.

Since after relocation in many schools average distance per students are reduced, a comparison map of average distance before and after relocation is also shown in figure 5.12 (a). The figure 5.12 (b) shows that most of the schools' total distance have decreased. For some schools though total distance is low but average distance per student is very high. It has been understood from both figure (a) and (b). Figure 5.12 shows that after rellocation most of the schools are saturated with capacity although total distance per school is reduced.





(a) (b)



(c)

Figure: 5.12 Relocation analysis on (a) Average distance (b) Total distance (c) Capacity-allocation

5.5.2. Second possibility: primary school students decrement

Primary student population may decrease due to migration. Migration may happen for higher opportunities in other cities, economic collapse and catastrophe. In this case one or two of the existing schools can be closed to reduce the cost. As a scenario we can assume that five schools need to be closed due to migration of 1200 students. In this case 34 school locations need to be selected out of 39. After reduction total student number will be 5019. Now if we consider that students have uniformly migrated from Enschede then we have to consider a new factor according to equation 5-1.

$$factor \leq \frac{5019}{15487} \approx .30$$

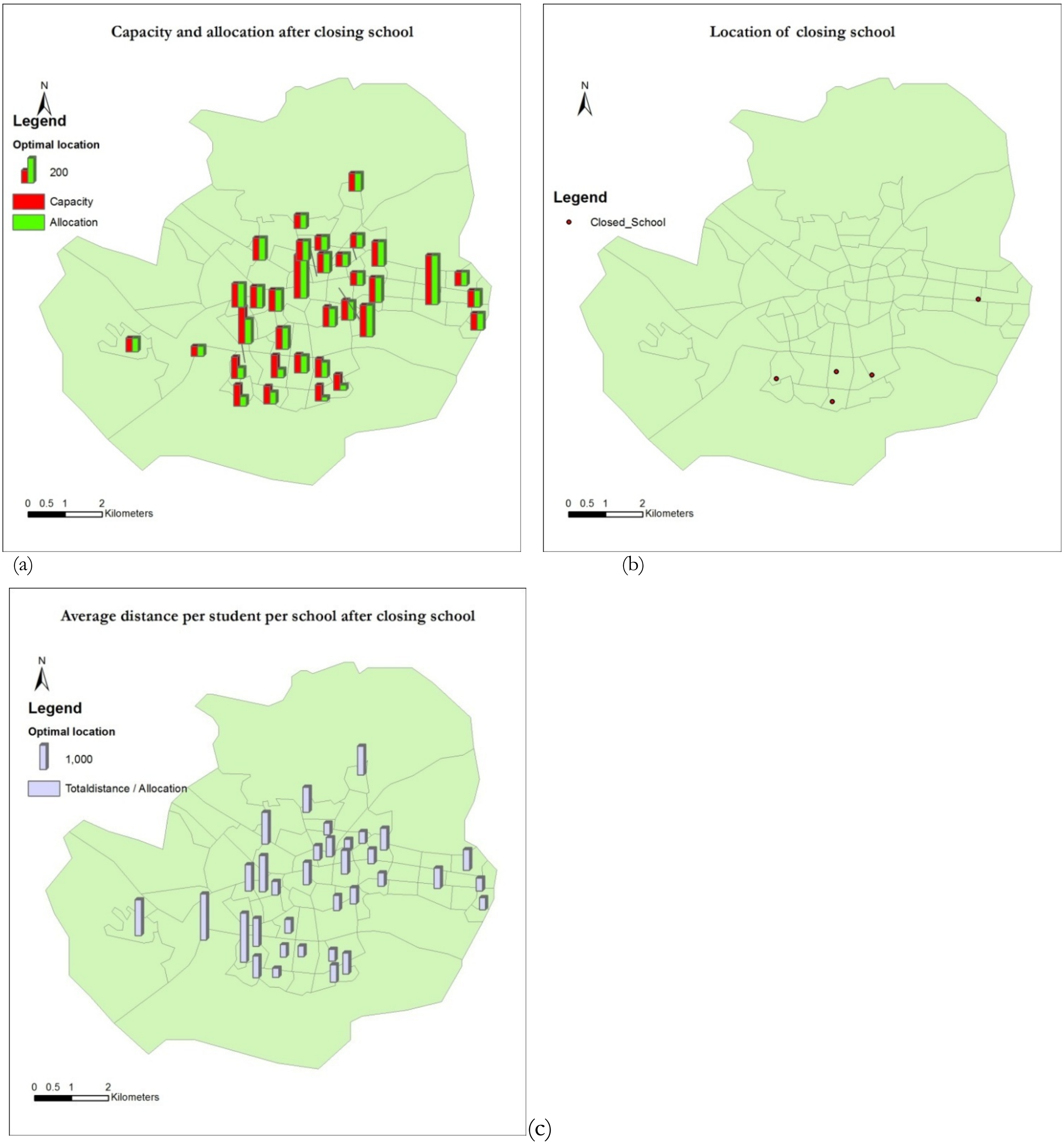


Figure: 5.13 (a) Capacity-allocation (b) Closed school location(c) Average distance per student per school –Closing school scenario

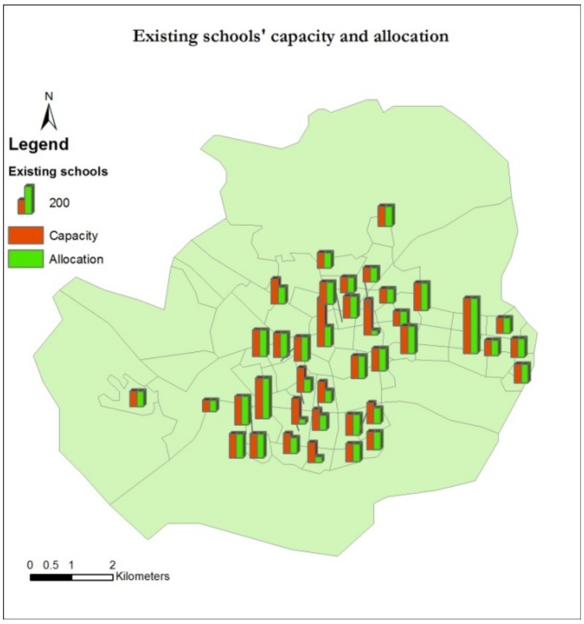
5019 demand points were generated randomly according to factor proportion to each neighbourhood data. A generated spider web for location allocation and fitness graph along its data is shown in appendix 4. Total distance for all 5019 student is 4300411 meter. Average distance for any student is around 856. Although 5 schools are closed, average distance has been increased compared to 2003 allocation. This occurs due to distribution of demand over whole Enschede remained almost same. Figure 5.13(a) shows capacity and allocation after optimal location selection from model and figure 5.13 (b) shows closed school after optimal school location selection. Figure 5.13 (c) shows average distance per student per school after closing of 5 schools. In appendix 5 location allocation model output is shown with fitness graph.

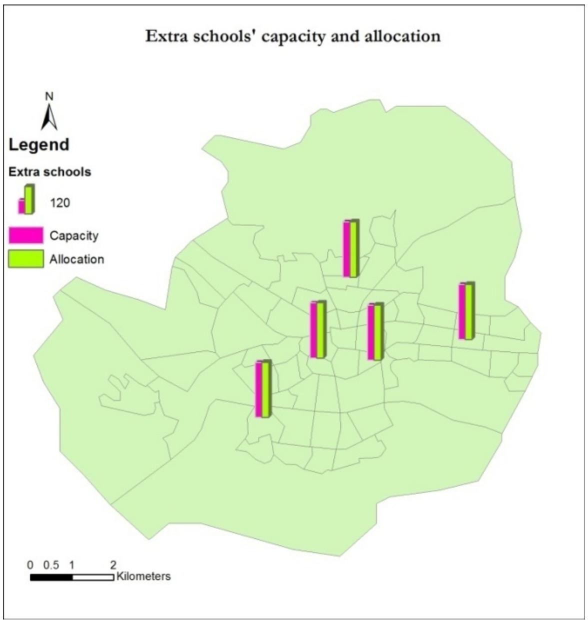
5.5.3. Third possibility-increment of primary school students

If the primary school students number increases then new schools need to be established to meet the extra demand. Here assumption is that the new school's allocation which will best for all students will also be best for new students. In this case, we have considered a scenario where 5 schools need to be set up for up to 1200 new students with an average capacity of 240.

$$factor \leq \frac{6467 + 1200}{15487 + 1200} \approx .45$$

With a factor of 0.40 less than 0.45, the number of new students will be 6675. 60 random potential locations will be used to select 5 extra facilities. For allocation of all student existing schools will also be considered. 6675 students as demand is created randomly over neighbourhoods of Enschede and saved as shape file.





(a) (b)

Figure: 5.14 (a) Students allocation in existing schools (b) Students allocation in extra schools

Figure 5.14 (a) and (b) shows students' allocation in both existing and extra schools. Model output along fitness graph is shown in the appendix 6.

5.6. Case-study 2, nearest distance with demand distribution by preference:

In case study at section 5.5 we have only considered nearest distance for all students. Student preference according to facility type was not considered. Decision makers may be flexible to some number of students by allowing them to choose their preferred school. In this case since we have no real life data we have assumed three types of school namely A, B, C. All schools are classified randomly among these three schools. For example, 20% schools are taken as 'A' type, 60% schools are 'B' type and 20% schools are 'C' type as assumption. Similarly all demands are classified. For example, 10% demands are taken as 'A' type, 80% demands are 'B' type and 10% demands are taken 'C' type. Both demand and facility type distributions are assigned randomly on demand and facility points respectively. We shall use the same genetic algorithm parameter of casestudy-1 for all three possibilities. But above stated user preference will be used with an addition to casestudy-1 parameters.

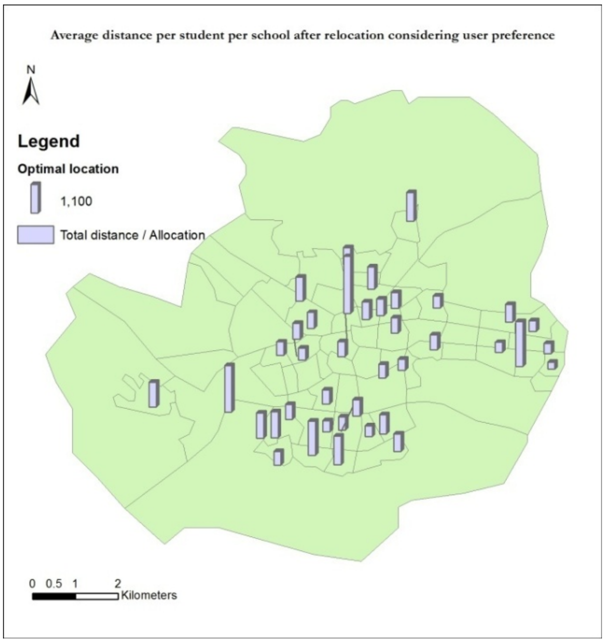
5.6.1. First possibility: stable number of primary school students (user preference):

The 4 new chosen facilities from the first possibility of case study 1 were added with 39 school locations according to data 2003. In total there are 43 potential facilities. This scenario can relocate maximum 4 schools. The same demand points of the first possibility of case study 1 were used so that we can compare relocation using nearest distance between with and without user preference. According to model output total distance is 4732765 meter. The number of students is 6219. Average distance of any student is around 761 meter which is higher than the average distance of the relocation of case study 1. Average distance per student per school is also high over 1000 meter for some schools which have been shown in figure 5.15 (a). Data is shown in appendix 7.

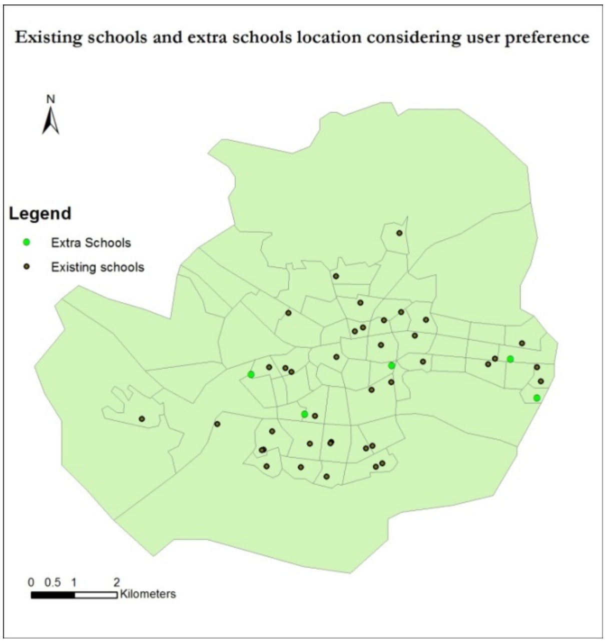
5.6.2. Second possibility: primary school students decrement (user preference)

For comparing 2nd possibility of case study 1, we have used exactly same data. We are going to close 5 schools and select 34 schools out of 39 locations. Same demand points are used of second possibility in case study 1. The number of demand points is 5019. Total distance is 5339304 meter. Average distance for any student is around 1064 meter. Data is shown in appendix 8. Figure 5.15 (c) shows the closed schools' locations according to location allocation model output.

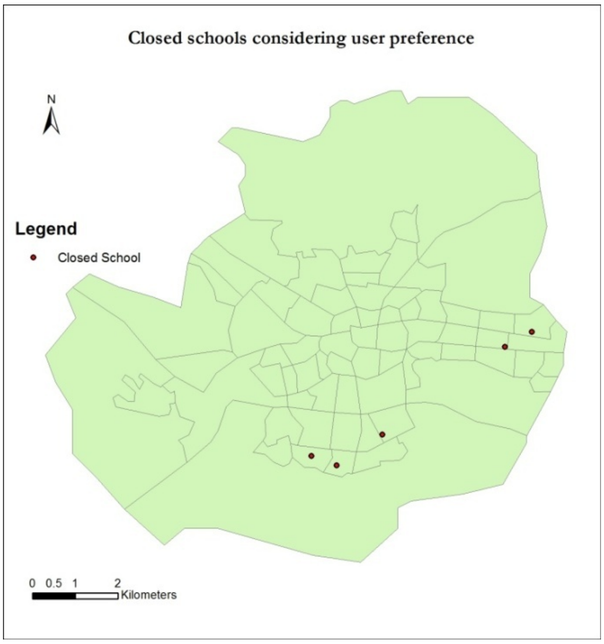
5.15 (d) indicate average distance per student per school. For some schools this distance is high and over 1400 meter.



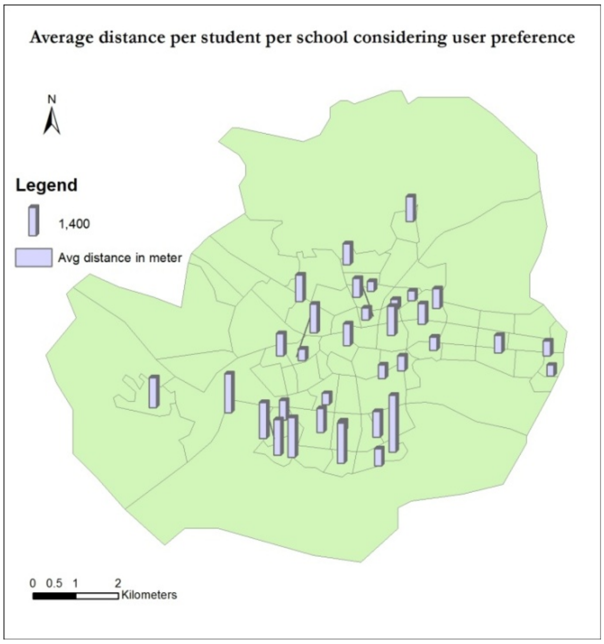
(a)



(b)



(c)



(d)

Figure: 5.15 (a) Average distance per student per student per school after relocation, (b) optimal location of existing and extra school (c) closed school location (d) average distance per student per school after school closing. (Considering user preference)

5.6.3. Third possibility-increment of primary school students using user preference:

In this third possibility, since we have to produce random potential facility, we only consider the scenario. We shall not compare this with third possibility of case study 1 because potential random facilities are not same in both cases.

The number of new students will be 6675. These demand points are also different from case study 1. 60 random potential locations will be used to select 5 extra facilities. For allocation of all student existing schools will also be considered. 6675 students as demand are created randomly over neighbourhoods of Enschede. Figure 5.15 (b) shows optimal location of extra facilities with existing facilities which are generated from our model and viewed in ARCGIS.

5.7. Discussion from both case studies:

From the first case study it has been clearly seen that relocation of some schools reduces total distance and hence average distance per student in Enschede. So if there is a forecast of remaining stable of public schools' students' amount, some schools can be relocated to reduce average distance of student. If there is a reduction of number of students, closing of some schools may increase the average distance of student per school. So it may be better to keep schools open even if there is reduction of student for certain period. If there is a forecast of the increment of the students in Enschede, establishing some extra school according forecasted demand will allocate all student considering certain average distance.

From the second case study, we can perceive that allocating student considering their preference is possible in Enschede. But this will increase average distance per student. This distance is considered as Euclidian. In Enschede, total optimal distance along road network is higher than total optimal distance using Euclidian distance according to section 5.4.1. So, considering user preference, average distance along road network will also be much higher than along Euclidian. So, decision makers should think about the threshold in the number of user preference which can create extra traffic in urban area.

5.8. Summary:

In this chapter, we have focused in optimizing algorithms and implementing case studies based on Enschede schools. In these two case studies we have tried to deal scenarios which can cope with increased, decreased and stable demand in long term. These have made the model complex.

Location allocation problem has become also complex using user preference with nearest distance.

6. CONCLUSION & RECOMMENDATION:

6.1. Conclusion:

The objective of this research is to solve location allocation problem using metaheuristic solutions. In order achieve this objective we have comprehensively studied previous literature to understand different types and classifications of location allocation problem and their solution. In order to build a model of location allocation problem's solution we had some research questions. The location allocation model was finally achieved following the proposed research methodology and having the answers of research questions accordingly

Question1. How to prepare and process GIS data in the model for metaheuristic solutions to find optimal location?

Genetic algorithm and simulated annealing are two different metaheuristic methodologies. Preparing GIS data for genetic algorithm and simulated annealing in the same model is a real challenge. We had to apply a technique that is supported by both algorithms. Instead of directly using geographic points into both genetic algorithm and simulated annealing, we have applied index of the points. That brings two different algorithms in the common platform. Another challenge was using most common GIS data type format like shape file as GIS data container. Some existing open source tools like SharpMap [64] and NetTopologySuite [66] are used in this concern. These tools are also used in delineating GIS information, performing basic GIS operation and saving information in shape file. Thus GIS data was handled in the model.

Questions2. What will be the objective function with applying capacitated facility and user preference in order to get optimal location?

The objective function is important for metaheuristic solutions. Our objective function is distance based single objective function. In the objective function capacity is used as constrained in allocating demand. Since we had two types of single objective like selecting facility with nearest distance and both nearest distance and user preference, we have used two types of objective function. Considering user preference, demand is distributed in the objective function.

Question3. When can genetic algorithm and simulated annealing be optimized?

With very long iteration both algorithms can give good result since these algorithms can diversify their search in the long run. But in the short iteration reaching global optima or near global optima is a challenge for both. By tuning parameter both genetic algorithm and simulated annealing can give optimize result in short iteration.

Question4. What are the strengths or weaknesses of genetic algorithm in this problem context with compare to simulated annealing?

We have seen genetic performance is better than simulated annealing. Since the fittest individual drives the search space towards a direction of good fitness value in genetic algorithm, it helps to achieve the convergence of the algorithm. On the other hand, simple simulated annealing depends only random search without having a driving factor towards a good fitness value. Simulated annealing is simple to implement and has less parameter to adjust than genetic algorithm.

With answering these questions, this research also has directed location allocation problem's solution into following achievements.

1. Capacitated location allocation problem's solution has been successfully integrated with GIS and solved by metaheristic solutions.
2. All used tools are open source.
3. It has been coped with increased, decreased and stable number of demands
4. It can deal not only Euclidian distance but also road network distance.
5. It has data interoperability with other GIS software, so it is loosely coupled with those for pre and post processing.

6.2. Reccomendation:

Some future works in this area can be as follows:

6.2.1. Multi-objective

In this research, we have solved single objective location allocation problem. But the solution of multi-objective was near to our current model. We haven't driven our solution towards multi-objective due to time constraint. We have left it as our future work.

6.2.2. Origin destination matrix from road network

We have used origin destination matrix generated from an open source software gvSIG [62]. This origin destination matrix is an input to our model. Future work can also be done to integrate with the model so that the model itself can create origin destination matrix.

6.2.3. Solution in vector –area process in raster

Our model has solution only for vector data. But after having solution in vector data it can be geo-referenced with raster data to have an area as solution rather than point. As one of the multiple objectives, the concept of using contiguity of cell (area) was applied with GA by Brookes in [33, 67]. But the concept is applied in land allocation not in facility location. This concept can be absorbed in the facility location.

LIST OF REFERENCES

[1] ESRI Inc. (2010, 22-8-2010). *Location-allocation analysis (In ArcGIS Desktop 10 Help ed.)* [Web page from website]. Available: http://help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/Location_allocation_analysis/004700000050000000/

[2] M. A P Taylor and L. Pitaksringkarn, "Grouping genetic algorithm in GIS: a facility location modelling," *Journal of the Eastern Asia Society for Tranportation Studies,* vol. 6, pp. 2908-2920, 2005.

[3] X. Li and A. Gar-On Yeh, "Integration of genetic algorithms and GIS for optimal location search," *International Journal of Geographical Information Science,* vol. 19, pp. 581-601, 2005.

[4] A. T. Murray and R. L. Church, "Applying simulated annealing to location-planning models," *Journal of Heuristics,* vol. 2, pp. 31-53, 1996.

[5] C. M. Hosage and M. F. Goodchild, "Discrete space location-allocation solutions from genetic algorithms," *Annals of Operations Research,* vol. 6, pp. 35-46, 1986.

[6] M. A. Arostegui Jr., S. N. Kadipasaoglub, and B. M. Khumawal, "An empirical comparison of Tabu Search, Simulated Annealing, and Genetic Algorithms for facilities location problems," *International Journal of Production Economics,* vol. 103, pp. 742-754, 2006.

[7] S. N. Sivanandam and S. N. Deepa, *Introduction to Genetic Algorithms*: Springer Berlin Heidelberg New York, 2008.

[8] M. Jamshidi, "Median Location Problem," in *Facility Location*, R. Zanjirani Farahani and M. Hekmatfar, Eds., ed: Physica-Verlag HD, 2009, pp. 177-191.

[9] Z. Azarmand and E. Neishabouri, "Location Allocation Problem," in *Facility Location*, R. Zanjirani Farahani and M. Hekmatfar, Eds., ed: Physica-Verlag HD, 2009, pp. 93-109.

[10] M. P. Scaparra and M. G. Scutellà, "Facilities, Locations, Customers: Building Blocks of Location Models. A survey.," Technical report TR-01-182001.

[11] E. Suomalainen, "Multicriteria Analysis and Visualization of Location- Allocation Problems," Masters thesis, Department of Engineering Physics and Mathematics, Helsinki University of Technology, 2006.

[12] J. Redondo, J. Fernández, I. García, and P. Ortigosa, "Sensitivity analysis of a continuous multifacility competitive location and design problem," *TOP,* vol. 17, pp. 347-365, 2009.

[13] A. J. Comber, S. Sasaki, H. Suzuki, and C. Brunsdon, "A modified grouping genetic algorithm to select ambulance site locations," *International Journal of Geographical Information Science,* 2010.

[14] S. Sasaki, A. Comber, H. Suzuki, and C. Brunsdon, "Using genetic algorithms to optimise current and future health planning - the example of ambulance locations," *International Journal of Health Geographics,* vol. 9, p. 4, 2010.

[15] M. N. Neema and A. Ohgai, "Multi-objective location modeling of urban parks and open spaces: Continuous optimization," *Computers, Environment and Urban Systems,* vol. 34, pp. 359-376, 2010.

[16] E. S. Correa, M. T. A. Steiner, A. A. Freitas, and C. Carnieri, "A Genetic Algorithm for Solving a Capacitated P-Median Problem," *Numerical Algorithms,* vol. 35, pp. 373-388, 2004.

[17] K. Ghoseiri and S. F. Ghannadpour, "Solving capacitated P-median problem using genetic algorithm," in *IEEE International Conference on Industrial Engineering and Engineering Management*, 2007, pp. 885-889.

[18] D. Gong, M. Gen, G. Yamazaki, and W. Xu, "Hybrid evolutionary method for capacitated location-allocation problem," *Computers & Industrial Engineering,* vol. 33, pp. 577-580, 1997.

[19] X. Li, Z. Liu, and X. Zhang, "Applying genetic algorithm and Hilbert curve to capacitated location allocation of facilities," in *2009 International Conference on Artificial Intelligence and Computational Intelligence, AICI 2009, November 7, 2009 - November 8, 2009*, Shanghai, China, 2009, pp. 378-383.

[20] A. Murray, "Advances in location modeling: GIS linkages and contributions," *Journal of Geographical Systems,* vol. 12, pp. 335-354, 2010.

[21] A. N. Sadigh and H. Fallah, "Demand Point Aggregation Analaysis for Location Models," in *Facility Location*, R. Zanjirani Farahani and M. Hekmatfar, Eds., ed: Physica-Verlag HD, 2009, pp. 523-534.

[22] M. Neema, K. Maniruzzaman, and A. Ohgai, "New Genetic Algorithms Based Approaches to Continuous p-Median Problem," *Networks and Spatial Economics,* pp. 1-17, 2008.

[23] S. Salhi and M. D. H. Gamal, "A Genetic Algorithm Based Approach for the Uncapacitated Continuous Location–Allocation Problem," *Annals of Operations Research,* vol. 123, pp. 203-222, 2003.

[24] S. L. Straitiff and R. G. Cromley, "Using GIS and K = 3 Central Place Lattices for Efficient Solutions to the Location Set-Covering Problem in a Bounded Plane," *Transactions in GIS,* vol. 14, pp. 331-349, 2010.

[25] H. Fallah, A. N. Sadigh, and M. Aslanzadeh, "Covering Problem," in *Facility Location*, R. Zanjirani Farahani and M. Hekmatfar, Eds., ed: Physica-Verlag HD, 2009, pp. 145-176.

[26] M. L. Brandeau and S. S. Chiu, "An Overview of Representative Problems in Location Research," *MANAGEMENT SCIENCE,* vol. 35, pp. 645-674, 1989.

[27] L. Cooper, "Location-Allocation Problems," *Operations Research,* vol. 11, pp. 331-343, 1963.

[28] S. L. Hakimi, "Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph," *Operations Research,* vol. 12, pp. 450-459, 1964.

[29] D. Gong, M. Gen, W. Xu, and G. Yamazaki, "Hybrid evolutionary method for obstacle location-allocation," *Computers & Industrial Engineering,* vol. 29, pp. 525-530, 1995.

[30] T. Uno, S. Hanaoka, and M. Sakawa, "An application of genetic algorithm for multi-dimensional competitive facility location problem," in *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, 2005, pp. 3276-3280 Vol. 4.

[31] A. L. Medaglia, J. G. Villegas, and D. M. Rodríguez-Coca, "Hybrid biobjective evolutionary algorithms for the design of a hospital waste management network," *Journal of Heuristics,* vol. 15, pp. 153-176, 2009.

[32] J. Yang, "Flow Capturing Location-Allocation Problem with Stochastic Demand under Hurwicz Rule," in *Proceedings of the 2008 Fourth International Conference on Natural Computation*, 2008, pp. 169-173.

[33] C. J. Brookes, "A genetic algorithm for designing optimal patch configurations in GIS," *International Journal of Geographical Information Science,* vol. 15, pp. 539 - 559, 2001.

[34] H. Min, V. Jayaraman, and R. Srivastava, "Combined location-routing problems: A synthesis and future research directions," *European Journal of Operational Research,* vol. 108, pp. 1-15, 1998.

[35] Flowmap-Lab. (27-1-2011). *Flowmap Home page* [Web page from website]. Available: http://flowmap.geog.uu.nl/

[36] J. C. Teixeira and A. P. Antunes, "A hierarchical location model for public facility planning," *European Journal of Operational Research,* vol. 185, pp. 92-104, 2008.

[37] F. J. F. Silva and D. S. de la Figuera, "A capacitated facility location problem with constrained backlogging probabilities," *International Journal of Production Research,* vol. 45, pp. 5117 - 5134, 2007.

[38] M. Jabalameli and A. Ghaderi, "Hybrid algorithms for the uncapacitated continuous location-allocation problem," *The International Journal of Advanced Manufacturing Technology,* vol. 37, pp. 202-209, 2008.

[39] W.-s. Liu, J.-f. Zhuang, and M.-q. Li, "A genetic algorithm approach to location-allocation problem in Urban Garbage Logistics System," in *IT in Medicine and Education, 2008. ITME 2008. IEEE International Symposium on*, 2008, pp. 71-75.

[40] D. Zhao, Y. Zhao, Z. Li, and J. Chen, "Multi-objective Emergency Facility Location Problem Based on Genetic Algorithm," *Computational Intelligence and Intelligent Systems,* vol. 51, pp. 97-103, 2009.

[41] O. Kariv and S. L. Hakimi, "An Algorithmic Approach to Network Location Problems. II: The p-Medians," *SIAM Journal on Applied Mathematics,* vol. 37, pp. 539-560, 1979.

[42] O. Kariv and S. L. Hakimi, "An Algorithmic Approach to Network Location Problems. I: The p-Centers," *SIAM Journal on Applied Mathematics,* vol. 37, pp. 513-538, 1979.

[43] M. Avazbeigi, "An Overview of Complexity Theory," in *Facility Location*, R. Zanjirani Farahani and M. Hekmatfar, Eds., ed: Physica-Verlag HD, 2009, pp. 19-36.

[44] Z. Khoban and S. Ghadimi, "Metaheuristic Methods," in *Facility Location*, R. Zanjirani Farahani and M. Hekmatfar, Eds., ed: Physica-Verlag HD, 2009, pp. 505-521.

[45] L. Cooper, "Heuristic Methods for Location-Allocation Problems," *SIAM Review,* vol. 6, pp. 37-53, 1964.

[46] R. L. Church, "Geographical information systems and location science," *Computers & Operations Research,* vol. 29, pp. 541-562, 2002.

[47] H. W. Hamacher, S. Nickel, and P. Dominguez-Marin. (2010, 22-1-2011). *LoLa* [Web page from website]. Available: http://www.mathematik.uni-kl.de/~lola/docu_prov.html

[48] S. Tafazzoli and M. Mozafari, "Classification of Location Models and Location Softwares," in *Facility Location*, R. Zanjirani Farahani and M. Hekmatfar, Eds., ed: Physica-Verlag HD, 2009, pp. 505-521.

[49] P. Domínguez-Marín, S. Nickel, P. Hansen, and N. Mladenović, "Heuristic Procedures for Solving the Discrete Ordered Median Problem," *Annals of Operations Research,* vol. 136, pp. 145-173, 2005.

[50] S. Samir. (2006, 19-1-2011). *TSP(Travelling Salesman Problem) using Genetic Algorithm* [Web page from website]. Available: http://www.codeproject.com/KB/cs/TSP_Solved_using_GA.aspx

[51] X. Qin, "A Social Vulnerability-Based Genetic Algorithm to Locate-Allocate Transit Bus Stops for Disaster Evacuation in New Orleans, Louisiana," Master of Science, The Department of Geography and Anthropology, Beijing Normal University, Beijing, China, 2009.

[52] J. H. Jaramillo, J. Bhadury, and R. Batta, "On the use of genetic algorithms to solve location problems," *Computers & Operations Research,* vol. 29, pp. 761-779, 2002.

[53] J. Dias, M. Captivo, and J. Clímaco, "A memetic algorithm for multi-objective dynamic location problems," *Journal of Global Optimization,* vol. 42, pp. 221-253, 2008.

[54] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science,* vol. 220, pp. 671-680, May 13, 1983 1983.

[55] A. Al-khedhairi, "Simulated Annealing Metaheuristic for Solving P-Median Problem," *International Journal of contemporary mathematical science,* vol. 3, pp. 1357-1365, 2008.

[56] S. Z. Selim and K. Alsultan, "A simulated annealing algorithm for the clustering problem," *Pattern Recognition,* vol. 24, pp. 1003-1008, 1991.

[57] R. Kincaid, "Good solutions to discrete noxious location problems via metaheuristics," *Annals of Operations Research,* vol. 40, pp. 265-281, 1992.

[58] J. E. Beasley, "OR-Library: distributing test problems by electronic mail," *Journal of the Operational Research Society,* vol. 41, pp. 1069-1072, 1990.

[59] F. Chiyoshi and R. Galvão, "A statistical analysis of simulated annealing applied to the p-median problem," *Annals of Operations Research,* vol. 96, pp. 61-74, 2000.

[60] T. V. Levanova and M. A. Loresh, "Algorithms of Ant System and Simulated Annealing for the p-median Problem," *Automation and Remote Control,* vol. 65, pp. 431-438, 2004.

[61] E. Municipality. (2010, 11-01-2011). *Buurtmonitor Enschede* [Web page from website]. Available: http://enschede.buurtmonitor.nl/

[62] gvSIG. (2009, 14-2-2011). *Portal gvSIG* [Web page from website]. Available: http://www.gvsig.org/web/?id=gvsig&L=2

[63] C. S. ReVelle and R. W. Swain, "Central Facilities Location," *Geographical Analysis,* vol. 2, pp. 30-42, 1970.

[64] SharpMap. (27-1-2011). *SharpMap Geosptation Application Framework for the CLR* [Web page from website]. Available: http://sharpmap.codeplex.com/

[65] A. Hamdar. (2008, 19-1-2011). *Simulated Annealing - Solving the Travelling Salesman Problem (TSP)* [Web page from website]. Available: http://www.codeproject.com/KB/recipes/simulatedAnnealingTSP.aspx

[66] NetTopologySuite. (27-1-2011). *NetTopologySuite is a C#/.NET port of JTS Topology Suite, a Java library for GIS operations, (OpenGIS compliant).* [Web page from website]. Available: http://code.google.com/p/nettopologysuite/

[67] C. J. Brookes, "A genetic algorithm for locating optimal sites on raster suitability maps," *Transactions in GIS,* vol. 2, pp. 201-212, 1997.

APPENDIX

Appendix 1 2003 year data for age group 4-12 downloaded from municipality website [61]

Serial	City_ID	CityID_NAME	Total4-12
1	00	00 City	41
2	01	01 Lasonder/'t Zeggelt	113
3	02	02 Laares	224
4	03	03 De Bothoven	184
5	04	04 Hogeland-Noord	225
6	05	05 't Getfert	229
7	06	06 Veldkamp, Getfert-West	176
8	07	07 Horstlanden/Stadsweide	83
9	08	08 Boddenkamp	28
10	10	10 Velve/Lindenhof	488
11	11	11 Wooldrik	119
12	12	12 Hogeland-Zuid	290
13	13	13 Varvik/Diekman	423
14	14	14 Sleutelkamp	12
15	15	15 't Weldink	0
16	16	16 De Leuriks	5
17	20	20 Cromhoffsbleek/Kotman	141
18	21	21 Boswinkel/de Braker	362
19	22	22 Pathmos	210
20	23	23 Stevenfenne	459
21	24	24 Stadsveld-Zuid	124
22	25	25 Elferink/Heuwkamp	280
23	26	26 Stadsveld-Noord/Bruggert	236
24	27	27 't Zwering	392
25	28	28 Ruwenbos	247
26	30	30 Tubantia/Toekomst	481
27	31	31 Twekkelerveld	299
28	40	40 Walhof/Roessingh	216
29	41	41 Bolhaar	149
30	42	42 Roombeek/Roomveldje	175
31	43	43 Mekkelholt	148
32	44	44 Deppenbroek	487
33	45	45 Voortman/Amelink	111
34	46	46 Drienerveld/UT	26
35	50	50 Schreurserve	233
36	51	51 't Ribbelt/Ribbelerbrink	180
37	52	52 Park Stokhorst	397
38	53	53 't Stokhorst	108
39	60	60 Stroinkslanden NO	502

40	61	61 Stroinkslanden Zuid	669
41	62	62 Stroinkslanden NW	275
42	63	63 Wesselerbrink NO	445
43	64	64 Wesselerbrink ZO	515
44	65	65 Wesselerbrink ZW	265
45	66	66 Wesselerbrink NW	730
46	67	67 Helmerhoek-Noord	721
47	68	68 Helmerhoek-Zuid	602
48	70	70 Industrie- en havengebied	32
49	71	71 Marssteden	10
50	72	72 Koekoeksbeekhoek	0
51	73	73 de Broeierd	1
52	80	80 Glanerveld	119
53	81	81 Bentveld/Bultserve	374
54	82	82 Schipholt/Glanermaten	388
55	83	83 de Eekmaat	250
56	84	84 Oikos	149
57	86	86 de Slank	27
58	87	87 Dolphia	99
59	88	88 Eekmaat west	352
60	90	90 Dorp Lonneker	170
61	91	91 Dorp Boekelo	244
62	92	92 Lonneker-West	125
63	93	93 Noord-Esmarke	33
64	94	94 Zuid-Esmarke	38
65	95	95 Broekheurne	131
66	96	96 Usselo	22
67	97	97 Goorseveld	75
68	98	98 Twekkelo	23

Appendix 2 Optimal allocation of public school students from data 2003

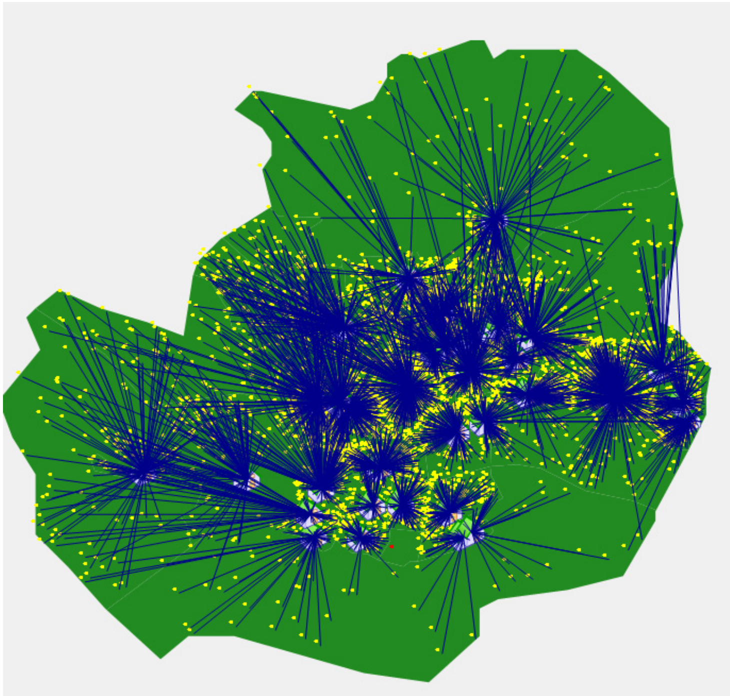
SchooNum	Total Distance for school	Capacity	Allocation	Avg Distance for user
1	75331	200	182	413.91
2	49609	103	103	481.64
3	159950	144	132	1211.74
4	49951	102	102	489.72
5	97226	110	110	883.87
6	48195	135	135	357
7	55560	135	135	411.56
8	60602	110	110	550.93
9	44164	110	110	401.49
10	221329	400	241	918.38
11	107259	129	129	831.47
12	67879	162	162	419.01
13	106262	200	142	748.32
14	40936	105	105	389.87
15	362294	255	255	1420.76
16	87760	163	163	538.4
17	65192	129	129	505.36
18	111752	150	150	745.01
19	60724	150	150	404.83
20	180856	158	158	1144.66
21	457579	350	350	1307.37
22	90240	150	150	601.6
23	69706	150	150	464.71
24	118633	175	175	677.9
25	98711	182	182	542.37
26	71803	174	174	412.66
27	66271	143	143	463.43
28	98517	174	174	566.19
29	138221	205	205	674.25
30	134308	144	144	932.69
31	289132	292	292	990.18
32	100093	174	174	575.25
33	159534	174	174	916.86
34	124595	191	191	652.33
35	96799	157	156	620.51
36	77745	110	110	706.77
37	167222	180	180	929.01
38	185068	80	80	2313.35
39	119861	112	112	1070.19

Appendix 3 After relocation, optimal allocation of public school students from data 2010.

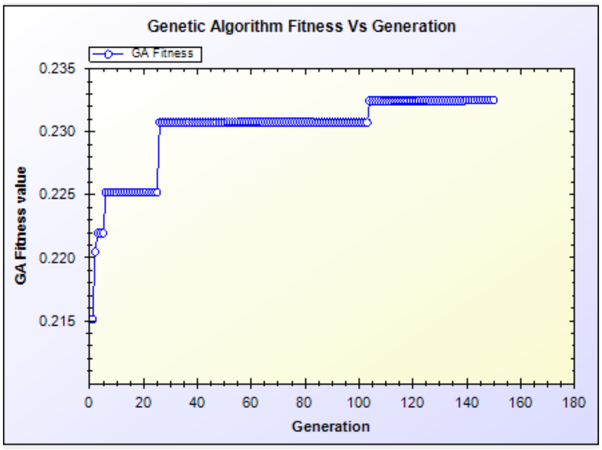
SchoolNUm	TotalDist	Capacity	Alloc	AvgDist	School_index
1	143505	200	128	1121.13	0
2	46714	103	103	453.53	1
3	84335	144	144	585.66	2
4	0	102	0	0	3
5	32874	110	110	298.85	4
6	61928	135	135	458.73	5
7	0	135	0	0	6
8	39621	110	110	360.19	7
9	55409	110	110	503.72	8
10	121965	400	272	448.4	9
11	61766	129	129	478.81	10
12	87588	162	151	580.05	11
13	67984	200	198	343.35	12
14	50759	105	105	483.42	13
15	172287	255	251	686.4	14
16	103091	163	163	632.46	15
17	65709	129	120	547.58	16
18	40819	150	81	503.94	17
19	124961	150	150	833.07	18
20	0	158	0	0	19
21	229794	350	350	656.55	20
22	57037	150	150	380.25	21
23	77199	150	150	514.66	22
24	78110	175	175	446.34	23
25	101570	182	182	558.08	24
26	150291	174	174	863.74	25
27	55906	143	143	390.95	26
28	0	174	0	0	27
29	119953	205	205	585.14	28
30	85678	144	144	594.99	29
31	141389	292	248	570.12	30
32	87525	174	174	503.02	31
33	83001	174	174	477.02	32
34	76459	191	148	516.61	33
35	67055	157	154	435.42	34
36	78703	110	110	715.48	35
37	119342	180	180	663.01	36
38	89258	80	46	1940.39	37
39	111623	112	112	996.63	38
40	138869	240	240	578.62	39
41	29103	240	41	709.83	40
42	136061	240	219	621.28	41
43	0	240	0	0	42
44	134393	240	240	559.97	43

Appendix 4 Optimal allocation of students after closing school along location allocation model & fitness graph

SchoolNum	Totaldistance	Capacity	Allocation
23	61169	150	139
32	98713	174	174
15	250441	255	255
21	325334	350	350
27	36654	143	95
10	334880	400	400
25	34364	182	67
16	91690	163	147
8	94236	110	110
6	68379	135	135
3	170080	144	144
12	100870	162	150
34	208739	191	191
33	260618	174	174
2	50235	103	103
39	166518	112	112
5	54052	110	110
26	71149	174	79
31	229390	292	197
19	60951	150	123
20	93839	158	158
28	177372	174	87
7	72128	135	135
38	153087	80	80
37	236029	180	180
14	63324	105	105
11	23246	129	32
4	47042	102	102
35	123816	157	157
36	113054	110	110
1	182558	200	200
13	109706	200	200
24	98919	175	175
17	37829	129	43

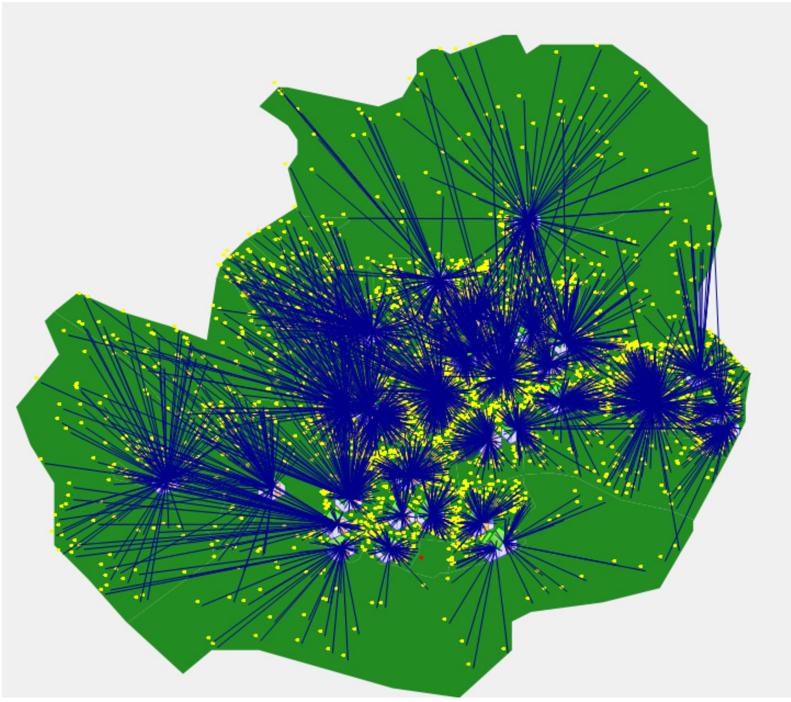


Location allocation model

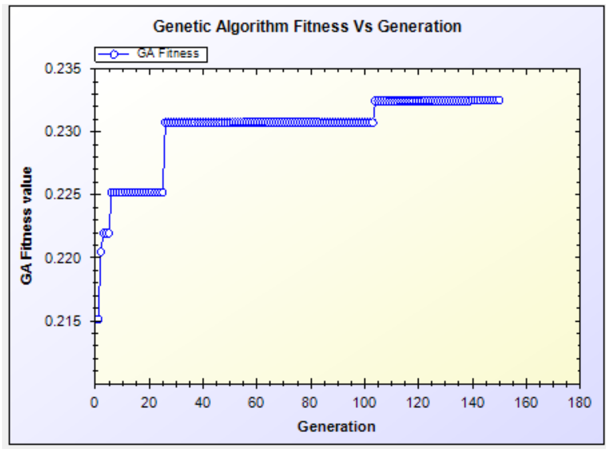


Fitness graph

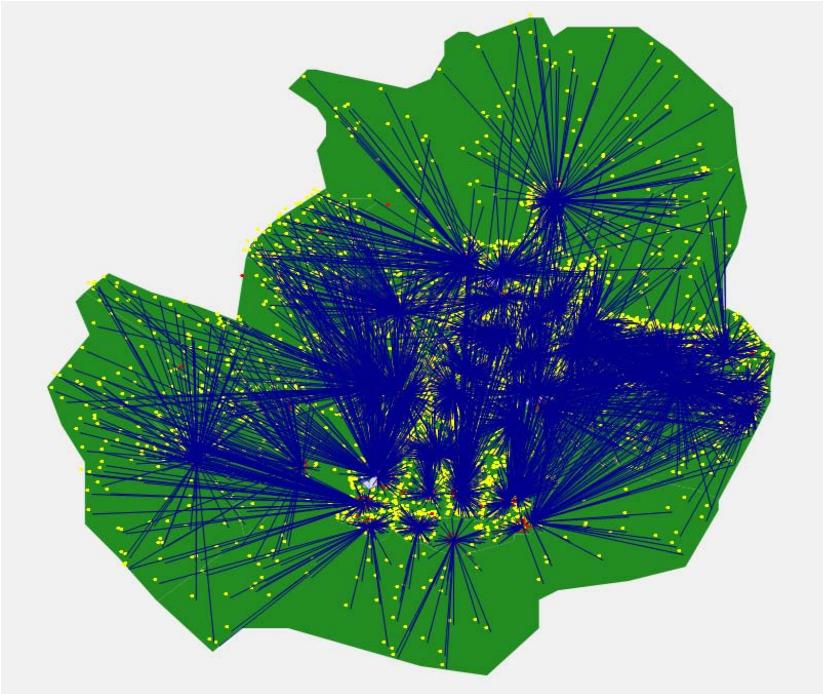
Appendix 5 Location allocation model and fitness graph for closing school.



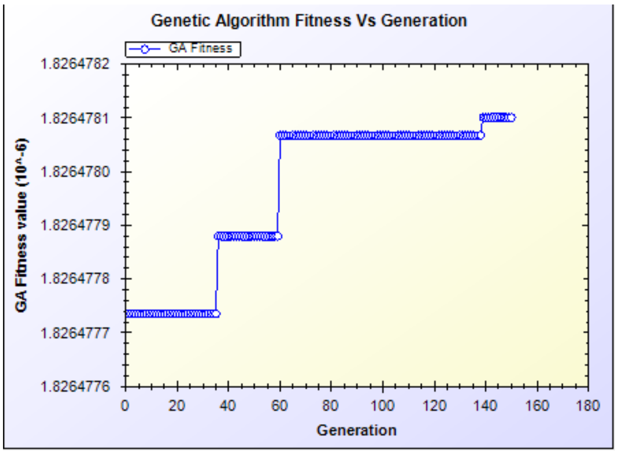
Location allocation model



Appendix 6 Extra school establishment for student increment in location allocation model



Location allocation model



Fitness graph

Appendix 7 Relocation data considering user preference

School number	Total distance	Capacity	Allocation
1	96378	200	200
3	153269	144	135
4	64948	102	102
5	96834	110	110
6	37489	135	135
7	54188	135	135
8	40715	110	95
10	59822	400	149
12	69890	162	162
13	120759	200	200
15	156495	255	255
16	88924	163	163
17	91183	129	129
18	115652	150	150
19	63412	150	150
20	85444	158	122
21	420361	350	185
22	79838	150	150
23	97029	150	150
24	98311	175	175
25	80542	182	182
26	94832	174	174
27	193716	143	143
28	180681	174	174
29	206987	205	205
30	135270	144	119
31	174631	292	292
32	83001	174	174
33	107428	174	174
34	103000	191	191
35	104133	157	157
36	73041	110	110
37	171758	180	180
38	113625	80	61
39	111623	112	112
40	127153	240	214
41	29103	240	41
42	398672	240	224
43	152628	240	240

Appendix 8 Closing school data considering user preference

School number	Total distance	Capacity	Allocation	Avg. distance
2	50235	103	103	487.72
39	166518	112	112	1486.77
23	280175	150	141	1987.06
37	236029	180	180	1311.27
14	104460	105	105	994.86
5	54052	110	110	491.38
24	98308	175	175	561.76
35	145236	157	157	925.07
33	247190	174	174	1420.63
15	367494	255	255	1441.15
11	25660	129	30	855.33
25	216504	182	182	1189.58
16	88704	163	131	677.13
12	95461	162	132	723.19
19	188972	150	150	1259.81
1	191415	200	196	976.61
29	84288	205	43	1960.19
17	320002	129	114	2807.04
36	113054	110	110	1027.76
3	175089	144	144	1215.9
4	47042	102	102	461.2
6	72274	135	135	535.36
32	98713	174	174	567.32
7	98275	135	135	727.96
34	208739	191	191	1092.87
28	77887	174	44	1770.16
22	35985	150	73	492.95
21	375531	350	350	1072.95
20	93839	158	158	593.92
26	266279	174	151	1763.44
38	153087	80	80	1913.59
13	121192	200	185	655.09
10	317248	400	370	857.43
31	124367	292	127	979.27