MINIMUM CONNECTION OF PIPES FOR WATER DISTRIBUTION NETWORK IN AGRIC NSIMA

BY

EBENEZER FRIMPONG OFOBI

A THESIS SUBMITTED TO THE DEPARTMENT OF MATHEMATICS

KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY

IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE DEGREE

OF MASTER OF PHILOSOPHY IN INDUSTRIAL MATHEMATICS

DEPARTMENT OF MATHEMATICS

FACULTY OF PHYSICAL SCIENCE

COLLEGE OF SCIENCE

JUNE 2012

DECLARATION

I Ebenezer Frimpong Ofobi, hereby declare that this thesis, "Minimum connection of pipes for water distribution network", consist entirely of my own work produced from research undertaken under supervision and that no part of it has been presented for another degree elsewhere, except for the permissible excerpts/references from other sources, which have been duly acknowledged.

Ebenezer Frimpong Ofobi	…………………….	………………………
PG4069210	Signature	Date

Certified by

Rev. W. Obeng –Denteh	……………………..	………………………
Supervisor	Signature	Date

Certified by

Prof. I. K. Dontwi	……………………….	………………………
Dean of IDL	Signature	Date

Certified by

Mr. K. F. Darkwah	……………………	………………………
Head of Department	Signature	Date

ABSTRACT

It is worthy to know that water is a constrained natural resource. Shortage of water therefore is a serious problem that requires urgent solution. In recent times, research on network has been on the increase. There is the need for improved network design, as demand for water from supply industries is high. Minimum connection of pipes for a network using shortest path algorithm means saving cost, thereby helping the government and GWCL to extend their services to meet the needs of the increasing population. The research showed that shortage of water for a long time at Nsima was due to broken pipes, rust pipes, the quantity of water pumped and the nature of network that existed at the time. A software Tora for Windows 2 analysis was used to analyze the data and assess cost of laying new pipes of network for Nsima township.

ACKNOWLEDGEMENT

I am grateful to great God of Wonders, for the provision of material, spiritual and intellectual resources during the entire period of the programme. I ask for God's bountiful blessings for Rev. W. Obeng-Denteh for his constructive criticism and immense support, without which this work would not have been possible. I wish to thank Mr. K. F. Darkwah, Dr. Frimpong and Mr. Charles Seibil for providing valuable suggestion and advice towards this study, I wish to express my profound gratitude to the entire staff of the mathematics Department, KNUST for their hospitality and encouragement.

Furthermore, my sincere thanks go to Mr. S. Gikunoo and Emmanuel Aduhene all of Ghana Water Company Limited and the good people of Agric Nsima for providing information data for my project.

I am also thankful to Mr. Frank Owusu, Prince N. Nketia and Mrs. Christiana Ofobi Frimpong for their numerous contribution towards this study.

Again I owed it a duty to thank my parents as well as my wife for their love and support.

DEDICATION

I dedicate this thesis to my father J. K. Asamoah, my dear wife Christiana, and my children Harriet, Eugene, De-Haan and Kwaku Frimpong.

TABLE OF CONTENT

Content **Pages**

CHAPTER THREE: METHODOLOGY

CHAPTER FOUR: DATA COLLECTION AND ANALYSIS

CHAPTER FIVE: SUMMARY, CONCLUSION AND RECOMMENDATION

LIST OF TABLES

Content **Page**

LIST OF FIGURES

Content **Page**

CHAPTER ONE

INTRODUCTION

1.1 Background

The saying is sure that water is life.

One could imagine how people would feel if they were without water for even a day. Two-thirds the weight of an average man is given by the content of water in him. In other words, the dry weight of an average man (that is when completely without water) is about thirteen kilogrammes. Man can hardly live without water. Human beings can survive longer without food than without water (Ayoade, 1975, 1988, NEST, 1991). Furthermore, the World Health Organisation has indicated that seventy five litres of water a day is necessary to protect people against household diseases and fifty litres a day is necessary for basic family sanitation. The 4th World water Forum (March 2006), reported that a person who lives in an urban area uses an average of 250 litres a day; but there is variation worldwide as far as human consumption is concerned (THD 2007)

On the field of Agriculture, plants and animals all need water for their survival. Most plants absorb nutrients dissolved in water from the soil. In addition to that, photosynthesis cannot take place without the presence of water. Animals also consume a lot of water in order to survive. It is interesting to know that most animals cannot do well on the desert as far as living conditions there are concerned.

In Ghana we most often than not depend on the rainfall pattern for our farming activities. Ghana has a tropical climate. The rainfall pattern varies and it is influenced by the southwest monsoon. Mean rainfall varies from 2000 mm in the southwest coastal area to about 850mm in the eastern coastal area and 1000mm in the north (Cornish and Lawrence 2001). As a matter of fact, water forms the largest part of most living matter. While on transportation water plays a great role in transporting and exporting goods (freight). This goes a long way to promote business. As a result it promotes the growth of the country's economy especially through trade. On the field of engineering, most machines use water to cool down their engines. This enables the machines to function well.

Water serves as a source of employment for people. For instance people are engaged as workers at Ghana water company limited, and people who dug wells for communities or families, and sachet water producers. etc.

Another point worth mentioning is that the World Health Organisation has indicated that seventy five litres of water a day is necessary for protection against household diseases and fifty litres a day as far as basic family sanitation is concerned.

River Volta is the biggest river in Ghana. The river has two hydroelectric dams. It also has a catchment area of 165700 km^2 within the country. Behind the Akosombo dam is Volta Lake. It is about 300 km long and transverses the centre of the country. However the demand for water exceeds its supply in Ghana. Urban communities obtain most of their supply of water from rivers at dams and diversion structures. Water from this source is usually treated before it is supplied. This helps to meet the health needs of the people. Sometimes especially in the dry season, access to pipe borne water is not easy. At times citizens experience acute water shortage for a number of days. In Kumasi, the main source of water is the pipe borne water supply from treated water sources (Barekese and Owabe dam).

The rural communities get their source of water mainly from ground (wells). These wells are either hand dug or drilled. The diameter of a hand-dug well, generally have a large diameter with depth ranging from 5m to 20m. Simple tools such as pick-axe, shovel etc, are used for its construction. A well of this nature is capable of serving at least two hundred people. On the contrary, the hand or machine drilled wells otherwise known as boreholes have small diameter about 100-200 cm as compared to the hand-dug ones and have a great depth of, up to 50m. Usually sophisticated equipments that are empowered by diesel or electric motors are used for this operation. It is important to know that the number of people the bore hole can serve depends to large extent, on the capacity of the hand pump. The ground water in most cases is considered to be good. There are however some exceptional cases where the water may not be safe for drinking particularly in locations where the sanitary conditions are poor. Unsafe water supply poses health threat to people. Waste water contamination serves as a source of bacteria viruses, and parasites that can cause gastrointestinal problems or transmit contagious diseases (Arms, 2000; UMES, 2008). In the dry season most wells dry up. Little or virtually no water is available from these wells for the residents. People may have to queue for the little water available.

It is important to know that there is inconsistency between access data from various sources. This may be due to the fact that the provision of access data is obtained by different institutions using different definitions. The joint monitoring program for water supply and sanitation from WHO and UNICEF access is as follows:

Table 1 .1 Multi – donor Africa MDG assessment

		Urban (46% of the population)	**Rural (54% of the population)**	**Total**
Water	Broad definition	88%	64%	75%
	House connection	37%	4%	19%
Sanitation	Broad definition	27%	11%	18%
	Sewerage	13%	2%	7%

Access to improved water sources, according to the Multi-donor Africa (MDG) is much lower (56%) while access to improved sanitation is higher (35%). The share of non-functional supply systems in Ghana is estimated as one-third, with many others operating substantially below designed capacity. Moreover, domestic water supply competes with a rising demand for water by the expanding industrial and agricultural sectors. Ghana aims at achieving 85% coverage for water supply and sanitation by 2015, which would exceed the Millennium Development Goals' target of 78%.
(Africa Economic Outlook, 2007- Ghana country Note).

The World water day is celebrated yearly on the 22nd of March, with the aim of addressing issues relating to water resources, their management and the supply of portable water. At the 2002 World Summit on Sustainable Development in Johannesburg, South Africa, great concern was expressed about the 1.1 billion people in the world who do not have access to safe drinking water and the 2.4 billion who live without sanitation (Cech, 2005):

In Ghana, the poor (defined by Living Standards Measurement Criteria) make up 47% of the total population in urban piped system areas (PURC, 2005). Within urban piped-system areas only 15% of the poor have access to piped water either directly or via yard taps [ibid].

The value of water is determined by two elements: supply – the cost of providing the resource in a certain quality, quantity and location which varies in different parts of the country and, Demand: - the utility to humans and their willingness to pay for that utility (Cech, 2005). It is important to know that laying of pipelines in the form of networks system is costly. Using or applying the appropriate method can help reduce cost.

1.2. Statement of the problem

Nsima community needs a new pipe layout. This has become necessary as a result of the following reasons:

- The old pipes have rusted. This happened because the pipes had been allowed to fall into disuse for almost ten years now.
- The increase in size of the town as well as the rapid population growth requires pipe layout which is suitable for the new status of the town.

1.3 Research Objectives

- To model the laying of pipe network as a minimum connector
- To use Prim's algorithm to obtain minimum cost of laying pipes.

1.4. Justification

Ghana's population is increasing at a faster rate hence the demands for water is high. By coming out with a model, for minimum connection of pipes, leads to saving cost. The government can therefore use the surplus for other cities and village as well. Because fewer pipes are used it will augur well for any small amount of water pumped to serve a lot of cities. G.W.C.L. would therefore maximize the operations of the company through this research by embracing suitable schemes to effectively supply to the consumers demand. The research findings are to intensify the existing store of knowledge on the topic. It will also serve as a spring board for further research.

1.5. Scope of the Study

The research work is limited to Agric Nsima township, Agric Nsima township is under GWCL water supply from Barekese Dam. The respondents will cover some employees at the company and residence of Agric Nsima Township. The area covered was the map of pipe layout from source to destinations within the town a suburb of Kumasi Metropolis. Findings of this study apply to the above mentioned town, which was selected for the study.

1.6 Methodology

The shortest path method which includes Floyds –Warshall algorithm, Prim's and Kruskal was used to minimize the pipe network system of Ghana Company Limited at Agric Nsima township. The researcher used Prim's algorithm which is good for smaller network system to minimize the pipe network systems of the Agric Nsima township. The map of the network system of Agric Nsima township and the distances they cover was provided by Geographical information service (GIS) of GWCL.

1.7. Limitations of the Study

The main aim of the study is to minimise cost of water supply by using shortest path model to investigate the impact of the arrangement of pipes. This will help GWCL to maximize profit. The constraint on time was a major setback as far as the conduct of the study was concerned. As a result enough time was not available for the researcher to use the total population. Furthermore, getting respondents for the answering of questionnaire was a problem. Funding was another setback since the research was financed solely by the researcher.

1.8 History of Ghana Water Company Limited

The first piped water supply system was constructed at Cape Coast in the year 1928. The Water Supply Division of the Public Works Department was put in charge for the service provision in rural and urban areas of Ghana. The division was separated from the public Works Department and placed under the Ministry of Works and Housing. This was done after the independence of Ghana in 1957. The division was later changed to the Ghana

Water and Sewerage Corporation (GWSC) in 1965 under an act of parliament (Act 310) as a legal public entity. It was once again transformed to a limited liability company in 1999 with the name Ghana Water Company Limited under Act 461 as a statutory Corporation LI. 1648. The mission of Ghana Water Company Limited is to meet the increasing demand for better service delivery by efficiently and effectively managing their core business which is production, transmission, distribution of water and customer management.

1.9 Organization of the Study

Chapter one introduces the background study, problem statement, objectives, significance and justification of the study, scope of the study, limitations, and organisation of the study.

Chapter two is the literature review. This chapter is about the theoretical and empirical background literature.

Chapter three discusses the Methodology. Chapter four looks at the analysis and discussions of data. Chapter five talks about the summary, conclusion and recommendations of the study.

CHAPTER TWO

LITERATURE REVIEW

It is important to know that I am not the first to write on networking. As a result, I hope to review some literature in this field of networks and shortest path algorithm in particular. This chapter is grouped under sub-topics:

i. Network in general
ii. Shortest path algorithm
iii. Water distribution networks

2.1 NETWORK IN GENERAL

Albert-L Barab, author of 'Linked: the new science of networks' (Perseus Publishing 2002), writes; The diversity of networks in business and the economy is mind-boggling. There are policy networks, ownership networks, collaboration networks, organizational networks, network marketing – you name it. It would not be possible to integrate these diverse interactions into a single all-encompassing web. Yet no matter what organizational level we look at, the robust and universal laws that govern nature's web seem to greet us.' the possible impact that mathematics on network has on industries and businesses is overwhelming. Executives with this notion of possible impact of mathematics on their job have a superb analogy from financial innovation, the Nobel Prize-winning Black-Merton-Scholes - pricing equations. The mathematics was as much a machine tool for creating options as a diagnostic tool for analyzing them. Clever 'quants' could use equations to spot 'hidden options' in financial instruments and wring profits from them, or, alternatively, apply the equations to customers to customize innovative financial instruments for their clients. In recent times, most companies use real options as mathematical tools for pricing the risks associated with their own business investment.

Maini et al, (2005) proposed an evolving network model which exhibits community structure. The network model is based on the inter-community preferential attachment and mechanisms. The degree of distributions, of this network model, is analyzed based on a mean field method. Theoretical results and numerical simulations indicate that this network model has community structure and scale-free properties.

Many important applications cannot go on without multicommodity-flow problem. Of late steady advances have been made by researchers in solving extremely multicommodity-flow problem. This has been made possible partly due to algorithmic and to hardware advances. At the moment the primal simplex method using the basis-partitioning approach gives excellent solution times even on the modest hardware. These results imply that we can now efficiently solve the extremely large multicommodity-flow models that are found in the industries. The extreme point solution can also be quickly re-optimized to meet the additional requirement often imposed upon the continuous solution. Currently practitioners are using EMNET, a primal basis –partitioning algorithm, to solve extremely large logistics, a primal basis-partitioning algorithm, to solve extremely large logistics problems with more than 600000 constraints and 7000000 variables in the food industry cited. (Richard d. McBride, 1998)

(John W. Marner and Richard D. 2000) Proposed and tasted a new pricing procedure for solving large-scale structured linear programs. The procedure interactively solves a relaxed sub program to identify potential entering basic columns. The sub problem is chosen to exploit special structure rendering it easy to solve. The effect of the procedure is the reduction of the number of pivots needed to solve the problem. Their approach is motivated by the column-generation approach of Dantzig -Wolfe decomposition. They tested procedure on two sets of their multicommodity –flow problems. One group of test problems arises routing telecommunications traffic and the second group is a set of logistics problems. Which have been widely used to test multicommodity flow algorithms.

(Zilias kopoutos, K.A, 2000), cited Daganzo as having introduced the cell transmission model; a simple approach for modeling highway traffic flow consistent with the hydrodynamic model further used the cell transmission model to the single destination system Optimum Dynamic Traffic Assignment (SO DTA) problem as a Linear Program(LP) then demonstrated that the model can obtain insights into the DTA problem, and addressed various related issues, such issues, such as the concept of marginal travel time in a dynamic network and system optimum necessary and sufficient conditions. The model is limited to one destination and, although it can account for traffic realities as they are captured by the cell transmission model, it is not presented as an operational model for actual applications. The main objective of the presentation is to demonstrate that the DTA problem can be modeled as a LP, which allows the vast existing literature on LP to be used to better understanding and compute DTA. A numerical example Illustrate the simplicity and applicability of the proposed approach.

Georgiou, P and Papamichail D (2008) developed a non-linear programming optimization model with an integrated soil water balance, to determine the reservoir release policies, the irrigation allocation to multiple crops and the optimal cropping pattern in irrigated agriculture. Decision variables are the cultivated area and the water allocated to each crop. The objective function of the model maximizes the total form of income which is based on crop-water functions, production cost and crop prices. The proposed model is solved using the simulated annealing(SA) global optimization stochastic search algorithm combination with the stochastic gradient descent algorithm. The rainfall, evapotranspiration and inflow are considered to be stochastic and the model is run for expected values of the above parameters corresponding different probability levels of rainfall, evapotranspiration and inflow, four weather conditions are distinguished. The model takes into account an irrigation time internal in each growth stage and give the optimal distribution of area, the water to each crop and the total form of income. The outputs of this model were compared with the results obtained from the model in which the only decision variable are cultivated areas. The model was applied on data from a planned reservoir on the Havrias River in Northern Greece, is sufficiently general and has great potential to be applicable as a decision support tool for cropping patterns of an irrigated area an irrigation scheduling. (Yang W et al 2008) simulated a hydro-dynamic behaviour of flow in three different reactor clarifiers by three- dimensional, multiphrase flow model. The primary construction of reactor clarifier was based on the Baison Water Treatment Plant, Taiwan This is the traditional construction, and they call it Type A. The other two were designed in such a way as to make large well angle (Type B). Solid effluent flux can be calculated directly from this model. This simulation results showed that under the same daily throughput. The Type C construction of clarifier could decrease up flow fluid velocity in the clarifier and therefore reduce effluent water turbidity.

(Tabr, R. A. 2003) submitted a primal - dual path following interior –point method for solution of the optimal power flow dispatching (OPFD) problem. The underlying idea of most path following algorithms is relatively similar, starting from the Fiacco-Mc Cormic barrier function, define the central path and loosely follow it to the optimum solution. Several primal-dual method for OPF have been suggested, all of which are essentially direct extensions of primal-dual methods for linear programming. Nevertheless there are substantial variations in some crucial details which include the formulation of the non-linear problem, the associated linear system, the linear algebraic procedure to solve this system, the line search, strategies for adjusting the centering parameter, estimating higher order correction terms for the homology path and the treatment of indefiniteness. The presentation discusses some of the approaches that is undertaken in implementing a specific primal-dual method for OPFD. A comparison is carried out with previous research on interior-point methods for OPF. Numerical tests on IEEE systems and on a realistic network are very encouraging and show that the new algorithm converges while other algorithms fail.

2.2 SHORTEST PATH ALGORITHMS

In computer science, Prim's algorithm is an algorithm that finds a minimum spanning tree for a connected weighted undirected graph. This means it finds a subset of the edges that forms a tree that includes every vertex, where the total weight of all the edges in the tree is minimized. Prim's algorithm is an example of a greedy algorithm. The algorithm was developed in 1930 by Czech mathematician Vojtèch Jarnik and later independently by computer scientist Robert C. Prim in 1957 and rediscovered by Edsger Dijkstra in 1959. Therefore it is also sometimes called the DJP algorithm, the Jarnik algorithm, or the Prim-Jarnik algorithm.

Gonina et al, (2007) described parallel implementation of Prim's algorithm for finding a minimum spanning tree of a dense graph using MPI. Our algorithm uses a novel extension of adding multiple vertices per iteration to achieve significant performance improvements on large problems (up to 200,000 vertices). We describe several experimental results on large graphs illustrating the advantages of our approach on over a thousand processors.

Gloor et al, (1993) described a system for visualizing correctness proofs of graph algorithms. The system has been demonstrated for a greedy algorithm, Prim's algorithm for finding a minimum spanning tree of an undirected, weighted graph. We believe that our system is particularly appropriate for greedy algorithms, though much of what we discuss can guide visualization of proofs in other contexts. While an example is not a proof, our system provides concrete examples to illustrate the operation of the algorithm. These examples can be referred to by the user interactively and alternatively with the visualization of the proof where the general case is portrayed abstractly.

McCarthy et al, (2009) presented the application of two well known graph algorithms, Edmonds' algorithm and Prim's algorithm, to the problem of optimizing distributed SPARQL queries. In the context of this paper resolved by contacting any number of remote SPARQL endpoints. Two optimization approaches are described. In the first approach, a static query plan is computed in advance of query execution, using one of two standard graph algorithms for finding minimum spanning trees (Edmonds' algorithm and Prim's algorithm). In the second approach, the planning and execution of the query are interleaved, so that as each potential solution is expanded it is permitted to follow an independent query plan. Our optimization approach requires basic statistics regarding RDF predicates which must be obtained prior to the user's query, through automated querying of the remote SPARQL endpoints.

Martel et al. (2002) studied the expected performance of Prim's minimum spanning tree (MST) algorithm implemented using ordinary heaps. We show that this implementation runs in linear or almost linear expected time on a wide range of graphs. This helps to explain why Prim's algorithm often beats MST algorithms which have better worst-case run times. Specifically, we show that if we start with any n node m edge graph and randomly permute its edge weights, then Prim's algorithm runs in expected O(m + nlogn log(2m/n)) time. Note that O(m + nlogn log(2m/n)) = 0(m) when m = C2(n log n log log n).We extend this result to show that the same expected run times apply even when an adversary can select the weights of m/logn edges and the possible weights of the remaining edges (which are then randomly assigned). Chang et al, (2008) described the reasons about why it is beneficial to combine with graph theory and board game. Forbye, it also descants three graph theories: Dijkstra's, Prim's, and Kruskal's minimum spanning tree. Then it would describe the information about the board game we choose and how to combine the game with before-mentioned three graph theories. At last, we would account for the advantage of combining with these three graph theories and the game specifically.

Dijkstra's algorithm, conceived by Dutch computer scientist Edsger Dijkstra in 1959, is a graph search algorithm that solves the single-source shortest path problem for a graph with nonnegative edge path costs. producing a shortest path tree. This algorithm is often used in routing. An equivalent algorithm was developed by Moore in 1957. For a given source vertex (node) in the graph, the algorithm finds the path with lowest cost (i.e. the shortest path) between that vertex and every other vertex. It can also be used for finding costs of shortest paths from a single vertex to a single destination vertex by stopping the algorithm once the shortest path to the destination vertex has been determined. For example, if the vertices of the graph represent cities and edge path costs represent driving distances between pairs of cities connected by a direct road, Dijkstra's algorithm can be used to find the shortest route between one city and all other cities. As a result, the shortest path first is widely used in network routing protocols, most notably IS-IS and OSPF (epen Shortest Path First), The literature makes is abundantly clear that the procedure commonly known today as Dijkstra's Algorithm was discovered in the late 1950s, apparently independently, by a number of analysts. There are strong indications that the algorithm was known in certain circles before the publication of Dijkstra's famous paper. It is therefore somewhat surprising that this fact is not manifested today in the "official" title of the algorithm.

Dijkstra, (1959) submitted his short paper for publication in Numerische Mathematik June 1959. In November 1959, Pollack and Wiebenson [1960] submitted a paper entitled Solutions of the shortest-route problem - a review to the journal Operations Research. This review briefly discusses and compares seven methods for solving the shortest path problem.

However, the review presents a 'highly efficient' method, attributed - as a dateless private communication - to Minty. The procedure is defined as follows (Pollack and Wieberson [1960, p. 225]: The objective is to find the shortest path from city A to city B).

Sniedovich. (2006) described Dijkstra's Algorithm as one of the most popular algorithms in computer science. It is also popular in operations research. It is generally viewed and presented as a greedy algorithm. In this paper we attempt to change this perception by providing a dynamic programming perspective on the algorithm. In particular, we are reminded that this famous algorithm is strongly inspired by Bellman's Principle of Optimality and that both conceptually and technically it constitutes a dynamic programming successive approximation procedure par excellence. One of the immediate implications of this perspective is that this popular algorithm can be incorporated in the dynamic programming syllabus and in turn dynamic programming should be (at least) alluded to in a proper exposition/teaching of the algorithm.

Asano et al. (2009) presented two space-efficient algorithms. First, they showed how to report a simple path between MO arbitrary nodes in a given tree. Using a technique called "computing instead of storing", we can design a naive quadratic-time algorithm for the problem using only constant work space, i.e., O (log n) bits in total for the work space, where n is the number of nodes in the tree. Then, another technique "controlled recursion" improves the time bound to 0(n1+") for any positive constant".

Second, we describe how to compute a shortest path between two points in a simple n-gon. Although the shortest path problem in general graphs is NL-complete, this constrained problem can be solved in quadratic time using only constant work space.

Buriol et al, (2008) described dynamic shortest-path algorithms update the shortest paths taking into account a change in an arc weight. This paper describes a new generic technique that allows the reduction of heap sizes used by several dynamic single-destination shortest-path algorithms. For unit weight changes, the updates can be done without heaps. These reductions almost, always reduce the computational times for these algorithms. In computational testing, several dynamic shortest-path algorithms with and without the heap-reduction technique are compared. Speedups of up to a factor of

1.8 were observed using the heap-reduction technique on random weight changes and of over a factor of five on unit weight changes. We compare as well with Dijkstra's algorithm, which recomputes the paths from scratch. With respect to Dijkstra's algorithm, speedups of up to five orders of magnitude are observed.

Park, (2002) stated that a possible evolution for current telecommunication networks is towards an architecture which is increasingly connection oriented or based on virtual overlays, e.g., MPLS, as a means to support multiple service types and/or customers. At the same time such networks would continue support large amounts of traffic based on best effort service paradigm currently used in TCP/IP networks. This thesis investigates several aspects in the design and operation of such networks.

As technologies continue increasing the capacity of network resources to meet increasing traffic loads the number of users contending for a given congested resource is likely to grow. When such sharing is mediated by a best effort paradigm based on TCP, one might ask whether the effectiveness of the underlying mechanisms will scale. In this thesis we begin by assessing the scalability of active queue management combined with flow control mechanisms for higher capacity links and larger numbers of ongoing flows.

We propose and evaluate a simple algorithm for congestion avoidance which doesn't require that the number of connections be known (or estimated) but exhibits robust performance even when the number of contending flows varies over a wide range.

This provides a promising avenue for achieving good performance over a wide range of systems. The ability to perform a flexible routing of traffic loads across network resources is a key to enable providers to support a wide range of typically time varying loads. Nevertheless due to possible instabilities and unpredictability, operators have for the most part been unwilling to use dynamic routing mechanisms that adapt to changing loads on the fly. Instead they have opted to assign fixed administrative weights to links, and have traffic routed based on simple shortest path routing algorithms.

In his thesis he considered how to set those weights given a priori information on the traffic loads. This corresponds to solving an inverse problem: determine a set of fixed weights for network links such that shortest path routing of traffic gives a good flow distribution across the network. When the traffic being routed includes connection oriented services, a good distribution of load over the network, must consider both bandwidth and connection processing and maintenance resources. Indeed connection oriented services typically require network resources to process signaling messages for setup and tear down, as well as periodic signaling for maintenance purposes.

He proposed and evaluated an approach to solving this inverse problem when the objective is to balance loads across both bandwidth and connection signaling resources in the network. One way providers can deal with scalability issues associated with supporting multiple service classes or customer types is to setup overlay networks over a common infrastructure.

The last question he addressed in his thesis is associated with routing and provisioning virtual private networks (VPNs) when only a limited characterization of the traffic demands is available or desirable. Specifically we focus on the 'hose model' where only aggregate characterization/constraints on traffic entering and leaving a collection of endpoints is available. To this end we devise an approach to routing and provisioning VPNs so as to exploit spatio-temporal multiplexing gains to minimize provisioning cost. We study the same questions when the provider's objective is to maximize the lifetime of the routing decisions and/or capacity to carry new VPN requests rather than minimizing per VPN costs.

Ishikawa et al, (2007) proposed a parallel shortest path searching algorithm and implements it on a newly structured parallel reconfigurable processor, DAPDNA-2 (IPFlex Inc). Routing determines the shortest paths from the source to the ultimate destination through intermediate nodes.

In Open Shortest Path First (OSPF), Dijkstra's shortest path algorithm, which is the conventional one, finds the shortest paths from the source on a program counter-based processor. The calculation time for Dijkstra's algorithm is O (N2) when the number of nodes is N. When the network scale is large, calculation time required by Dijkstra's algorithm increases rapidly. It's very difficult to compute Dijkstra's algorithm in parallel because of the need for previous calculation results, so Dijkstra's algorithm is unsuitable for parallel processors.

Our proposed scheme finds the shortest paths using a simultaneous multi-path search method. In contrast with Dijkstra's algorithm, several nodes can be determined at one time. Moreover, we partition the network into different groups (network groups) and find the all-node pair's shortest path in each group using a pipeline operation. Networks can be abstracted, and the shortest paths in very large networks can be found easily. The proposed scheme can decrease calculation time from O (N2) to 0(N) using a pipeline operation on DAPDNA–2 . Our simulations show that the proposed algorithm uses 99.6% less calculation time than Dijkstra's algorithm. The proposed algorithm can be applied to the very large Internet network designs of the future.

Goldberg et al, (2006) proposed shortest path algorithms that use a search in combination with a new graph-theoretic lower-bounding technique based on landmarks and the triangle inequality. Our algorithms compute optimal shortest paths and work on any directed graph. We give experimental results showing that the most efficient of our new algorithms outperforms previous algorithms, in particular an search with Euclidean bounds, by a wide margin on road networks and on some synthetic problem families.

Pangilinan et al, (2007) presented an overview of the multi-objective shortest path problem (MSPP) and a review of essential and recent issues regarding the methods to its solution. The paper further explored a multi-objective evolutionary algorithm as applied to the MSPP and describes its behavior in terms of diversity of solutions, computational complexity, and optimality of solutions. Results show that the evolutionary algorithm can find diverse solutions to the MSPP in polynomial time (based on several network instances) and can be an alternative when other methods are trapped by the tractability problem. As in the case of the single-objective shortest path problem, the multi-objective shortest path problem has been studied extensively by various researchers in the fields of optimization, route planning for traffic, transport design. information and communications network design. The MSPP is an extension of the traditional shortest path problem and is concerned with finding a set of efficient paths with respect to two or more objectives that are usually in conflict. For example, the problem of finding optimal routes in communications networks involves minimizing delay while maximizing throughput or finding efficient routes in transportation planning that simultaneously minimize travel cost, path length, and travel time. The concept of optimization in the MSPP or in a multi-objective problem in general is different from the single-objective optimization problem wherein the task is to find a solution that optimizes a single objective function. The task in a multi-objective problem is not to find an optimal solution for each objective function but to find an optimal solution that simultaneously optimizes all objectives. And in most cases, no single optimal solution exists, only a set of efficient or non-dominated solutions.

A variety of algorithms and methods such as dynamic programming, label selecting, label correcting, interactive methods, and approximation algorithms have been implemented and investigated with respect to the MSPP. The problem is known to be NP-complete. It has been shown that a set of problems exist wherein the number of Pareto-optimal solutions is exponential which implies that any deterministic algorithm that attempts to solve it is also exponential in terms of runtime complexity at least in the worst case. But some labeling algorithm studies, dispute this exponential behavior. They show that the number of efficient paths is not exponential in practice. Other authors avoid the complexity problem by developing new methods that run in polynomial time. For instance, Hansen and Warburton separately develop fully polynomial time approximation schemes (FPTAS) for finding paths that are approximately Pareto-optimal. Interactive procedures similarly avoid the problem of generating the complete set of efficient paths by providing a user-interface that assists the decision-maker to focus only on promising paths and identify better solutions according to preference.

Given the wealth of literature in multi-objective algorithms for the MSPP, there still seems to be a lack of reported review in evolutionary algorithm (EA) applications in relation to the MSPP. Several of the most recent alternative methods focus mostly on execution speed comparisons of different MSPP algorithms but analysis of the salient issues in multi-objective performance analysis such as runtime complexity, diversity, and optimality of non-dominated solutions are almost omitted. In order to demonstrate a clearer picture of the advantages and disadvantages of EAs in optimization, this paper attempts to investigate a multi-objective evolutionary algorithm as applied to the MSPP.

Divoky, (1990) presented a framework for solving the shortest-path, cost-flow problem with positive edge weights can be implemented by itself or as a subordinate process in a solution procedure for bigger problems. The efficiency of such shortest-path frameworks depends on the technique employed to use the structure of the network from the solution of the bigger problem, as well as on the efficiency of the frameworks. The topology of the networks is characterized by arcs with positive weights and of sub networks called zero-weight components. The techniques for using the topology include: identifying the zero-weight edges and adding them to the graph; identifying and adding the basic sub-trees to the shortest-path tree; and interrupting the shortest-path frameworks and scanning or fixing the labels of all the nodes in the sub-tree.

Pettie et al, (2002) evaluated the practical efficiency of a new shortest path algorithm for undirected graphs which was developed by the rest two authors. This algorithm works on the fundamental comparison-addition model. Theoretically, this new algorithm out-performs Dijkstra's algorithm on sparse graphs for the all-pairs shortest path problem, and more generally, for the problem of computing single-source shortest paths from different sources. Our extensive experimental analysis demonstrates that this is also the case in practice. The authors presented results which showed the new algorithm to run faster than Dijkstra's on a variety of sparse graphs when the number of vertices ranges from a few thousand to a few million, and when computing single-source shortest paths from as few as three different sources.
Jesper et al. (2002) presented a simple parallel algorithm for the single-source shortest path problem in planar digraphs with nonnegative real edge weights. The algorithm runs on the EREW PRAM model of parallel computation in $0((n2\ \Sigma+nl-\Sigma) \log n)$ time, performing $0(n1+\Sigma \log n)$ work for any $0< \Sigma <1/2$. The strength of the algorithm is its simplicity,

making it easy to implement and presumable quite efficient in practice. The algorithm improves upon the work of all previous parallel algorithms. Our algorithm is based on a region decomposition of the input graph and uses a well-known parallel implementation of Dijkstra's algorithm. The logarithmic factor in both the work and the time can be eliminated by plugging in a less practical, sequential planar shortest path algorithm together with an improved parallel implementation of Dijkstra's algorithm.

Wang et al, (2005) said multiple pairs shortest path problem (MPSP) arises in many applications where the shortest paths and distances between only some specific pairs of origin-destination (OD) nodes in a network are desired. The traditional repeated single-source shortest path (SSSP) and all pairs shortest paths (APSP) algorithms often do unnecessary computation to solve the MPSP problem. We propose a new shortest path algorithm to save computational work when solving the MPSP problem. Our method is especially suitable for applications with fixed network topology but changeable arc lengths and desired OD pairs. Preliminary computational experiments demonstrate our algorithm's superiority on airline network problems over other APSP and SSSP algorithms.

Chang, (2009) described the shortest distance between two points as a straight line. But in the real world, if those two points are located at opposite ends of the country, or even in different neighborhoods, it is unlikely you will find a route that enables you to travel from origin to destination via one straight road. You might pull out a map to determine the fastest way to drive somewhere, but these days, you are just as likely to use a Web-based service or a handheld device to help with driving directions. The popularity of mapping applications for mainstream consumer use once again has brought new challenges to the research problem known as the "shortest-path problem".

The shortest-path problem, one of the fundamental quandaries in computing and graph theory, is intuitive to understand and simple to describe. In mapping terms, it is the problem of finding the quickest way to get from one location to another. Expressed more formally, in a graph in which vertices are joined by edges and in which each edge has a value, or cost, it is the problem of finding the lowest-cost path between two vertices. There are already several graph-search algorithms that solve this basic challenge and its variations, so why is shortest path perennially fascinating to computer scientists?

Goldberg, (2001), principal researcher at Microsoft Research Silicon Valley, said there are many reasons why researchers keep studying the shortest-path problem. "Shortest path is an optimization problem that's relevant to a wide range of applications, such as network routing, gaming, circuit design, and mapping," Goldberg says. "The industry comes up with new applications all the time, creating different parameters for the problem. Technology with more speed and capacity allows us to solve bigger problems, so the scope of the shortest-path problem itself has become more ambitious. And now there are Web-based services, where computing time must be minimized so that we can respond to queries in real time".

Venkataraman et al, (2003) proposed a blocked version of Floyd's all-pairs shortest-paths algorithm. The blocked algorithm makes better utilization of cache than does Floyd's original algorithm. Experiments indicate that the blocked algorithm delivers a speedup (relative to the unblocked Floyd's algorithm) between 1.6 and 1.9 on a Sun Ultra Enterprise 4000/5000 for graphs that have between 480 and 3200 vertices. The measured speedup on an SGI 02 for graphs with between 240 and 1200 vertices is between 1.6 and 2.

Karp et al, (1980) proposed let G = (V, E) be a digraph with n vertices including a special vertex s. Let E' C E be a designated subset of edges. For each e E E there is an associated real number fl (e). Furthermore, let 1 if e E E' f2 (e): 0 if e E-E' The length of edge e is flpe)-Af2 (e), where X is a parameter that takes on real values. Thus the length varies additively in X for each edge of E'. We shall present two algorithms for computing the shortest path from s to each vertex v E V parametrically in the parameter X, with respective running times 0(n3) and (nlE llogn).

For dense digraphs the running time of the former algorithm is comparable to the fastest (non-parametric) shortest path algorithm known. This work generalizes the results of Karp [2] concerning the minimum cycle mean of a digraph, which reduces to the case that E' = E. Furthermore, the second parametric algorithm may be used in conjunction with a transformation given by Bartholdi, Orlin, and Ratliff to give an 0(n21 logn) algorithm for the cyclic staffing problem.

Arulselvan et al, (2008) considered the problem of maximizing the total connectivity for a set of wireless agents in a mobile ad hoc network. That is, given a set of wireless units each having a start point and a destination point, our goal is to determine a set of routes for the units which maximizes the overall connection time between them. Known as the COOPERATIVE COMMUNICATION PROBLEM IN MOBILE AD HOC NETWORKS (CCPM), this problem has several military applications including coordination of rescue groups, path planning for unmanned air vehicles, and geographical exploration and target recognition. The CCPM is NP-hard; therefore heuristic development has been the major focus of research. In this work, we survey the CCPM examining first some early combinatorial formulations and solution techniques. Then we introduce new continuous formulations and compare the results of several case studies. By removing the underlying graph structure, we are able to create a more realistic model of the problem as supported by the numerical evidence.

Razzaque et al, (2009) said a fast algorithm is proposed to calculate kth power of an n x n Boolean matrix that requires addition operations, where p is the probability that an entry of the matrix is 1. The algorithm generates a single set of inference rules at the beginning. It then selects entries (specified by the same inference rule) from any matrix (Ak-1) and adds them up for calculating corresponding entries of Ak. No multiplication operation is required. A modification of the proposed algorithml can compute the diameter of any graph and for a massive random graph, it requires only 0(n2 (1-p)E[q]) operations, where q is the number of attempts required to find the first occurrence of 1 in a column in a linear search. The performance comparisons say that the proposed algorithms outperform the existing ones.

The Floyd-Warshall algorithm, also variously known as Floyd's algorithm, the Roy Floyd algorithm, the Roy-Warshall algorithm, or the WFI algorithm, is an algorithm for efficiently and simultaneously finding the shortest paths (i.e., graph geodesics) between every pair of vertices in a weighted and potentially directed graph. The Floyd algorithm is essentially equivalent to the transitive closure algorithm independently discovered by Roy (1959) and Warshall (1962) (Pemmaraju and Skiena (2003), which is the reason it is associated with all three authors.

Hougardy, (2010) the Floyd–Warshall algorithm is a simple and widely used algorithm to compute shortest paths between all pairs of vertices in an edge weighted directed graph. It can also be used to detect the presence of negative cycles. Hougardy, (2010) show that for this task many existing implementations of the Floyd–Warshall algorithm will fail because exponentially large numbers can appear during its execution.

Misra et al, (2006) presented a new solution to the Dynamic All-Pairs Shortest Path Routing Problem, using a linear reinforcement learning scheme. The particular instance of the problem that we have investigated concerns finding the all-pairs shortest paths in a stochastic graph, where there are continuous probabilistically-based updates in edge-weights. We present the details of the algorithm with an illustrative example. The algorithm can be used to find the all-pairs shortest paths for the "statistical" average graph, and the solution converges irrespective of whether there are new changes in edge-weights or not. On the other hand, the existing algorithms will fail to exhibit such a behavior and would recalculate the affected shortest paths after each edge-weight update. There are two important contributions of the proposed algorithm. The first contribution is that not all the edges in a stochastic graph are probed and, even if they are, they are not all probed equally often. Indeed, the algorithm attempts to almost always probe only those edges that will be included in the final list involving all pairs of nodes in the graph, while probing the other edges minimally. This increases the performance of the proposed algorithm.

The second contribution is designing a data-structure, the elements of which represent the probability that a particular edge in the graph lies in the shortest path between a pair of nodes in the graph. All the algorithms were tested in environments where edge-weights change stochastically and where the graph topologies undergo multiple simultaneous edge-weight updates. Its superiority in terms of the average number of processed nodes, scanned edges, and the time per update operation, when compared with the existing algorithms, was experimentally established. Chen et al, (2007) focused on the optimization problems about complicated network, this paper presents an algorithm KSPA to solve the K-shortest paths problem in complicated network based on algorithm, in which the time cost is taken as target function and the establishment of the target function model is given. Experimental results show the proposed KSPA maintains an excellent efficiency on certain public traffic data. It can be used to solve the K-shortest paths problems in multi-graph.

Lysgaard, (2000) presented a new algorithm for finding the shortest path from a source to a single sink in a network, in which the location in the plane of each node is known. The algorithm consists of two phases. In the first phase a heuristic solution to the shortest path problem is found. In the second phase the upper bound provided by the heuristic solution is utilized in a modification of a standard shortest path algorithm. Estimates based on computational tests show that on average the computation time of the presented algorithm is on the order of 40-60% of the computation time required if the information on node locations is not utilized.

Rabbani et al, (2008) presented a distribution network design problem in a multi-product supply chain system that involves locating production plants and distribution warehouses as well as determining the best strategy for distributing the product from plants to warehouses and from the warehouses to customers. The goal is to select the optimum numbers, locations and capacities of plants and warehouses to open, so that all customer demands of all product types are satisfied at minimum total costs of the distribution network. Unlike most of the previous researches, our study considers a multi-product supply chain system. We develop a mixed-integer mathematical programming model for designing a supply chain distribution network. Finally, this paper presents a real-case study to investigate designing a pharmaceutical supply chain distribution network. A possible extension is also offered in the conclusion. Multiple pairs shortest path problem (MPSP) that arises in many applications where the shortest paths and distances between only some specific pairs of origin-destination (OD) nodes in a network are desired. The traditional repeated single-source shortest path (SSSP) and all pairs shortest paths (APSP) algorithms often do unnecessary computation to solve the MPSP problem. Sokol et al, (2005) proposed a new shortest path algorithm to save computational work when solving the MPSP problem. Our method is especially suitable for applications with fixed network topology but changeable arc lengths and desired OD pairs. Preliminary computational experiments demonstrate our algorithm's superiority on airline network problems over other APSP and SSSP algorithms.

Hsieh et al, (2004) designed shortest path routing algorithms is in general more difficult than designing simple routing algorithms. In this paper, we derive a shortest path routing algorithm for pyramid networks. The proposed algorithm takes (1) time to determine a shortest path between any two nodes in a pyramid network. We also design a distributed routing algorithm such that an intermediate node takes (1) time to confirm the next node along the shortest path without any centralized controller.

Algorithms for finding the shortest path from one point to another have been researched for years. Applications abound, but let's keep things simple by saying we want to find the shortest path from point A to point B in a city with just a few streets and intersections. There are quite a few different algorithms that have been developed to solve such problems, all with different benefits and drawbacks. Before we delve into them though, let's consider how long a naive algorithm - one that tries every conceivable option - would take to run. If the algorithm considered every possible path from A to B (that didn't go in circles), it would not finish in our lifetimes, even if A and B were both in a small town. The runtime of this algorithm is exponential in the size of the input, meaning that it is O (C^N) for some C. Even for small values of C, CN becomes astronomical when N gets even moderately large. One of the fastest algorithms for solving this problem has a runtime of O (E*V*Log (V)), where E is the number of road segments, and V is the number of intersections. To put this in perspective, the algorithm would take about 2 seconds to find the shortest path in a city with 10,000 intersections, and 20,000 road segments (there are usually about 2 road segments per intersection).

The algorithm, known as Djikstra's Algorithm, is fairly complex, and requires the use of a data structure known as a priority queue. In some applications, however, even this runtime is too slow (consider finding the shortest path from New York City to San Francisco - there are millions of intersections in the US), and programmers try to do better by using what are known as heuristics. A heuristic is an approximation of something that is relevant to the problem, and is often computed by an algorithm of its own. In the shortest path problem, for example, it is useful to know approximately how far a point is from the destination. Knowing this allows for the development of faster algorithms (such as A*, an algorithm that can sometimes run significantly faster than Djikstra's algorithm) and so programmers come up with heuristics to approximate this value. Doing so does not always improve the runtime of the algorithm in the worst case, but it does make the algorithm faster in most real-world applications.

Traffic information systems are among the most prominent real-world applications of Dijkstra's algorithm for shortest paths. Schulz et al, (2000) considered the scenario of a central information server in the realm of public railroad transport on wide-area networks. Such a system has to process a large number of on-line queries for optimal travel connections in real time. In practice, this problem is usually solved by heuristic variations of Dijkstra's algorithm, which do not guarantee an optimal result. We report results from a pilot study, in which we focused on the travel time as the only optimization criterion. In this study, various speedup techniques for Dijkstra's algorithm were analyzed empirically. This analysis was based on the timetable data of all German trains and on a "snapshot" of half a million customer queries. In graph theory, the shortest path problem is the problem of finding a path between two vertices (or nodes) such that the sum of the weights of its constituent edges is minimized. An example is finding the quickest way to get from one location to another on a road map; in this case, the vertices represent locations and the edges represent segments of road and are weighted by the time needed to travel that segment.

Formally, given a weighted graph (that is, a set V of vertices, a set E of edges, and a real-valued weight function f :p ER), and one element v of V, find a path P from v to v' of V so that;

from a source vertex v to all other vertices in the graph.

- The single-destination shortest path problem, in which we have to find shortest paths from all vertices in the graph to a single destination vertex v. This can be reduced to the single-source shortest path problem by reversing the edges in the graph.
- The all-pairs shortest path problem, in which we have to find shortest paths between every pair of vertices v, v' in the graph.

These generalizations have significantly more efficient algorithms than the simplistic approach of running a single-pair shortest path algorithm on all relevant pairs of vertices.

2.3. WATER DISTRIBUTION NETWORK

(Samani, H. M. V and Mottaghi A, 2006) opined that the optimum design of municipal water distribution networks for a single loading condition is determined by the branch and bound integer linear programming technique. The hydraulic and optimization analyses are linked through an iterative procedure. This procedure enables them to design a water distribution system that satisfies all required constraints with a minimum total cost. The constraints include pipe sizes which are limited to the commercially available sizes, reservoir levels, pipe flow velocities, and nodal pressures. The accuracy of the developed model has been assessed using a network with limited solution alternatives, the optimal solution of which can be determined without employing optimization techniques. The proposed model has also been applied to a network solved by others. Comparison of the results indicates that the accuracy and convergence of the proposed method is quite satisfactory.

(Biscos, C. et al 2003) presented an approach for the operational optimization of potable water distribution networks. The maximization of the use of low-cost power (e.g overnight pumping) and the maintenance of a target chlorine concentration at final delivery points were defined as important optimization objectives. The first objective is constrained by the maintenance of minimum emergency volumes in all reservoirs, while the second objective would include the minimization of chlorine dosage and re-dosage requirements. The routing makes this a challenging predictive control and constrained optimization problem, which is being solved by MINILP (Mixed Integer Non-linear Programming). Initial experimental results show the performance of this algorithm and its ability to control the water distribution process.

(Ahmed, E. E. et al 2010), present a mathematical simulation of canal operation for optimizing water allocation within a multiple crop rotation canal. Zero-one linear programming algorithm was made for optimal flow regulation and to optimal flow regulation and to optimize irrigation water allocation and sequencing of different outlets in the irrigation canal under the constraints of fixed canal capacity and irrigation interval. The model was coded in a personal computer using Excel – visual basic application language (VBA) and designed to serve as a decision-making tool for operating an existing canal or for appropriate sizing of canal capacity (flow rate and cross- sectional area) of a new one. The model is verified statistically in comparison with published models. The allocation model is applied to the real cases of wet and dry regions of Gezira Scheme-Sudan for sequencing canal outlets for the cases of early and peak season at Sunni Minor canal and Ugud Minor canal. The model sensitivity to changes in outlet inflow rate (working time per outlet) was made for the said two cases. This indicated the model capabilities to effectively provide a constant flow rate into the canal during the operating period, and consequently minimize the need to frequently adjust the settings of canal head regulator. The model capabilities to optimize water scheduling and allocation process can be used to save irrigated schemes in the Sudan.

2.4. Summary

The chapter deliberated on existing theoretical and empirical literature on the topic under discussion. A set of precedence-constrained procedures has to be scheduled in order to minimize a given objective as far as shortest path is concerned. In resource-constrained project, the jobs additionally compete for scarce resources. Shortest path algorithms can be applied in many areas including transportation, communication, networking, and project management because of its universality. It is one of the most intractable problems in operations research, consequently it has become a popular playground for the latest optimization techniques, including virtually all local search paradigms. It demonstrates that a seemingly classical mathematical programming approach leads to both competitive feasible solutions and strong lower bounds, within reasonable computation times. Resources and methodology used in developing the mathematical model (shortest path model) used in solving the problem are considered in the next chapter.

CHAPTER THREE

METHODOLOGY

Shortest path methods, particularly Prim's Algorithm, were used to minimize the pipe network system at Nsima.

3.1.0 DEFINTION OF TERMS:

- **Graph** = (V,E); V is a finite set of points called vertices and E is a finite set of edges
- **Undirected graph**— edge e Є E

  – unordered pair uv where u,v Є V

A **tree** is a cycle- free network which comprises of a subset of all the nodes.

A **spanning tree** of an undirected graph of n nodes is a set of n-1 edges that connects all nodes.

Minimum spanning tree is defined as follows: Given an undirected graph G = (V, E) with a

weight mapping w:E → R, find a connected sub graph $T=(V,E^1 c\ E)$ with |E'| = |V| -1 that minimizes the objective function $\Sigma_{ecE}{}^{1}\ w(e)$

Mathematical network: a network is a digraph with weighted edges

Algorithm: a set of instructions that are followed in a fixed order and used for solving a mathematical problem and making a computer programme The aim of **minimum connector** problems is to find a spanning tree of minimum weight.

3.2.0 A Minimum Connector Problem

A company is installing a system of cables to connect all towns in a region. The lower case variables in the network show distances in kilometers. The question is what is the least amount of cable needed?

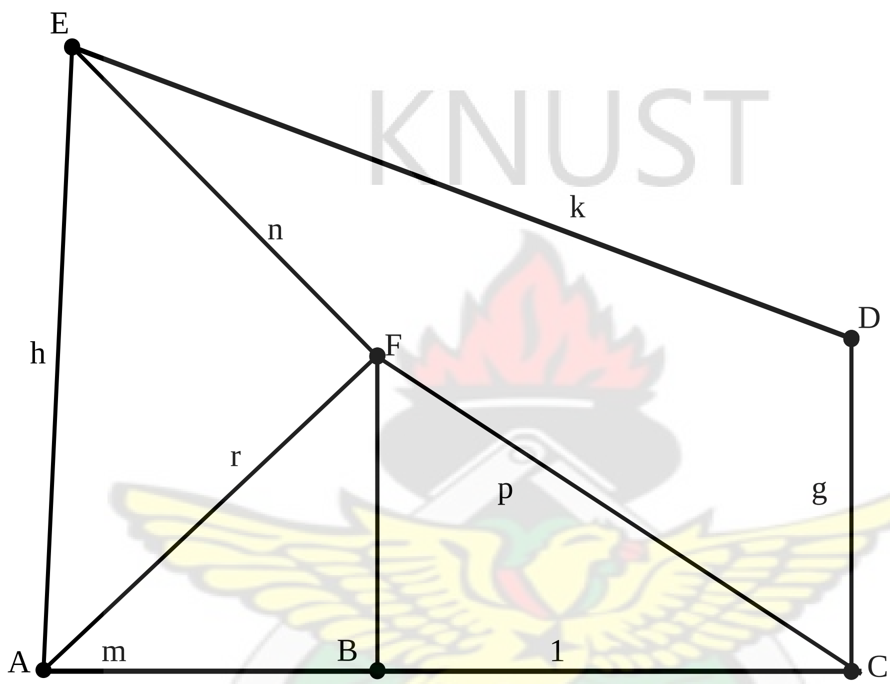


Fig. 3.1: Network of Towns

3.2.1. MINIMUM CONNECTOR FORMULATION

The Prim's algorithm for minimum connector problems, which finds minimum spanning tree, was used for an undirected graph *G= [V, E]*, where *V* is the set of vertices (nodes) and *E* is the set of edges. Each edge is defined by a pair of vertices *(u, v)*. Each edge has a nonnegative length. We start at a given node and sequentially add an edge to the tree at each iteration. The length (weight) of the graph is the total distance covered in town

3.3.1 Prim's Algorithm

Prim's algorithm is an algorithm in graph theory that finds a minimum spanning tree for a connected weighted graph. This means it finds a subset of the edges that forms a tree that includes every vertex, where the total weight of all the edges in the tree is minimized. If the graph is not connected, then it will only find a minimum spanning tree for one of the connected components. The algorithm was discovered in 1930 by mathematician Vojtech Jarnik and later independently by computer scientist Robert Prim in 1957 and rediscovered by Dijkstra in 1959. Therefore it is sometimes called DJP algorithm or Jarnik algorithm.

It works as follows:

- create a tree containing a single vertex, chosen arbitrarily from the graph
- create a set containing all the edges in the graph
- loop until every edge in the set connects two vertices in the tree

  *remove from the set an edge with minimum weight that connects a vertex in the tree with a vertex not in the tree

  * add that edge to the tree

Using a simple binary heap data structure, Prim's algorithm can be shown to run in time which is 0((m + n) log n) where m is the number of edges and n is the number of vertices. Using a more sophisticated Fibonacci heap, this can be brought down to 0(m +

nlog n), which is significantly faster when the graph is dense enough that m is

EXAMPLE

Given a network

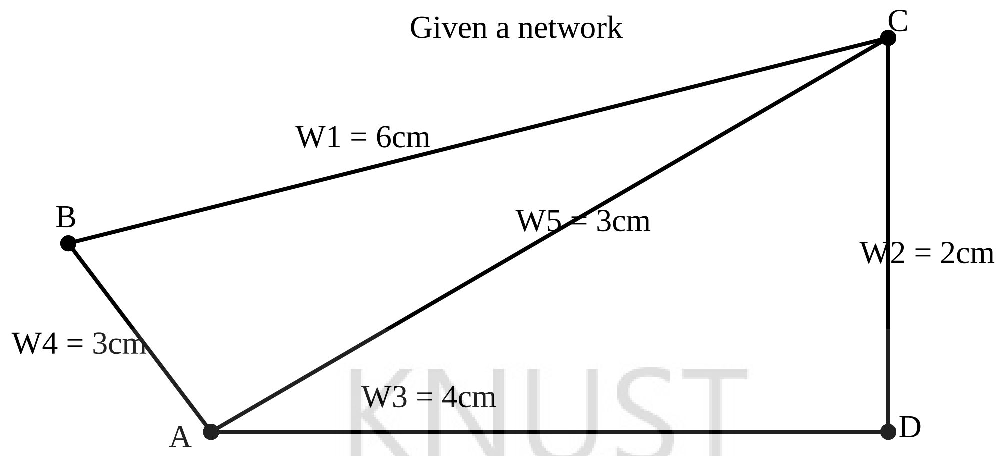


Fig 2: Network

Table 2: Matrix method:

Form a table for the above network as shown below:

	A	B	C	D
A	∞	3	3	4
B	3	∞	6	∞
C	3	6	∞	2
D	4	∞	2	∞

Choose a starting vertex say C. delete the row C. look for smallest entry in column

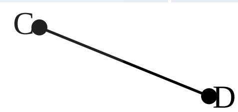
↓

	A	B	C	D
A	∞	3	3	4
B	3	∞	6	∞
~~C~~	~~3~~	~~6~~	~~∞~~	~~2~~
D	4	∞	2	∞

- **C**

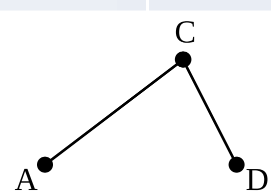
CD the smallest edge joining C is put into solution. Row D is deleted

	A	B	C	D
A	∞	3	3	4
B	3	∞	6	∞
~~C~~	~~3~~	~~6~~	~~∞~~	~~2~~
~~D~~	~~4~~	~~∞~~	~~2~~	~~∞~~



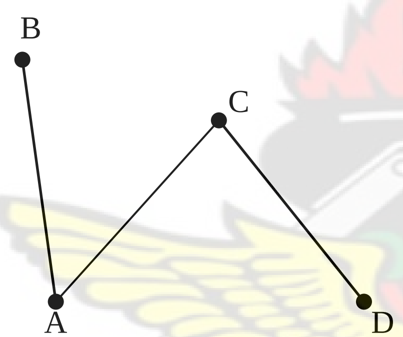
Smallest entry for C and D corresponds to A. B, C and D are put into solution. Row B is deleted

	A	B	C	D
~~A~~	~~∞~~	~~3~~	~~3~~	~~4~~
B	3	∞	6	∞
~~C~~	~~3~~	~~6~~	~~∞~~	~~2~~
~~D~~	~~4~~	~~∞~~	~~2~~	~~∞~~



The smallest entry for B, C and D corresponds to A. Row A is deleted. A, B, C and D enter into solution.

	A	B	C	D
A	∞	3	3	4
B	3	∞	6	∞
C	3	6	∞	2
D	4	∞	2	∞



3.4.0 Single-Source Paths, Non-Negative Weights (Dijkstra's Algorithm)

Dijkstra's algorithm, conceived by Dutch computer scientist Edsger Dijkstra in 1959, is a graph search algorithm that solves the single-source shortest path problem for a graph with nonnegative edge path costs, producing a shortest path tree. This algorithm is often used in routing. An equivalent algorithm was developed by Edward F. Moore in 1957.

For a given source vertex (node) in the graph, the algorithm finds the path with lowest cost (i.e. the shortest path) between that vertex and every other vertex. It can also be used for finding costs of shortest paths from a single vertex to a single destination vertex by stopping the algorithm once the shortest path to the destination vertex has been determined. For example, if the vertices of the graph represent cities and edge path costs represent driving distances between pairs of cities connected by a direct road, Dijkstra's algorithm can be used to find the shortest route between one city and all other cities. As a result, the shortest path first is widely used in network routing protocols, most notably IS-IS and OSPF (Open Shortest Path First).

3.4.1 Algorithm

Let the node we are starting be called an initial node. Let a distance of a node Y be the distance from the initial node to it. Dijkstra's algorithm will assign some initial distance values and will try to improve them step-by-step. Assign to every node a distance value. Set it to zero for our initial node and to infinity for all other nodes.

i. Assign to every node a distance value. Set it to zero for our initial node and to infinity for all other nodes.
ii. Mark all nodes as unvisited. Set initial node as current.
iii. For current node, consider all its unvisited neighbours and calculate their distance (from the initial node). For example, if current node (A) has distance of 6, and an edge connecting it with another node (B) is 2, the distance to B through A will be 6+2=8. If this distance is less than the previously recorded distance (infinity in the beginning, zero for the initial node), overwrite the distance.
iv. When we are done considering all neighbours of the current node, mark it

as visited. A visited node will not be checked ever again; its distance recorded now is final and minimal.

v. Set the unvisited node with the smallest distance (from the initial node) as the next "current node" and continue from step 3.

Dijkstra's algorithm finds the shortest paths from a source node S to all other nodes in a network with non-negative arc lengths. Dijkstra's algorithm maintains a distance label d(i) with each node i, which is an upper bound on the shortest path length from the source node to each node i. At any intermediate step, the algorithm divides the nodes of the network under consideration into two groups: those which it designates as permanently labeled (or permanent) and those which from it designates as temporarily labeled (or temporal). The distance label to any permanent node represents the shortest distance from the source node to that node.

The basic idea of the algorithm is to find out from the source node S and permanently labeled nodes in the order of their distances from the node S. Initially, node S is assigned a permanent label of zero, and each other; node j a temporary label equal to infinity. At each iteration, the label of a node I is its shortest distance from the source node along a path whose internal nodes (i.e. nodes other than S or the node I itself) are permanently labeled. The algorithm selects a node i with minimum temporary label (breaking ties arbitrarily), makes it permanent, and reaches out from node- that is, scans all the edges/arcs emanating from node I to update the distance labels of adjacent nodes. The algorithm terminates when it has designated all nodes permanent.

We can now express Dijkstra's algorithm as a set of steps.

Step 1: Assign the permanent label 0 to the starting vertex.

Step 2: Assign temporary labels to all the vertices that are connected directly to the most recently permanent labeled vertex.

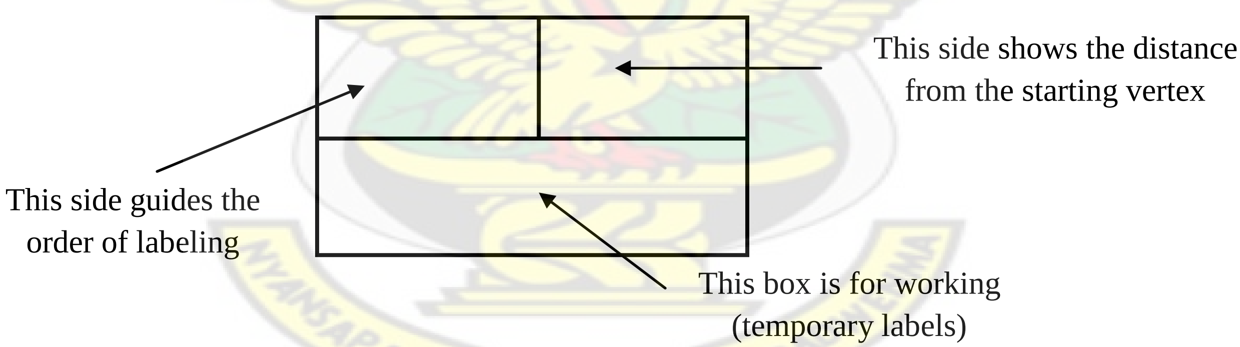
Step 3: Choose the vertex with the smallest temporary label and assign a permanent label to that vertex.

Step 4: Repeat steps 2 and 3 until all vertices have permanent labels.

Step 5: Find the shortest path by tracing back through the network.

Note

Recording the order in which we assign permanent labels to the vertices is an essential part of the algorithm.



The algorithm gradually changes all temporary labels into permanent ones.

3.4.2 All-Pairs Shortest Path Problem

The shortest path between two nodes might not be a direct edge between them, but instead involve a detour through other nodes. The all-pairs shortest path problem requires that we determine shortest path distances between every pair of nodes in the network.

3.4.3 Floyd-Warshall Algorithm

The Floyd-Warshall algorithm obtains a matrix of shortest path distances within 0 $\{n^3\}$ Computations. The algorithm is based on inductive arguments developed by an application of a dynamic programming technique.

Let $d^k(i, j)$ represent the length of the shortest path from node I to node j subject to the condition that this path uses the nodes 1,2,3,...,k-1 as internal nodes clearly, e^l (i, j) represents the actual shortest path distance from i and j. The algorithm first computes $d^{'}(i, j)$ for all node pairs i and j. Using $d^l(i, j)$, it then computes $d^2(i, j)$ for all pairs of nodes i and j. It repeats the process until it obtains $d^{n+1}(i, j)$ for all node pairs i and j when it terminates. Given $d^k(i, j)$, the algorithm computes $d^{k+1}(i, j) = \min\{ d^k(i, k) , d^k(i, j)$ 1. The Floyd-Warshall algorithm remains of interest because it handles negative weight edges correctly.

3.4.4 Data Collection

The information and Data needed for the analysis was gathered from Geographical Information Services of Ghana Water Company Limited.

3.4.5 Summary

In this chapter, we considered the modeling of the water supply problem into shortest path problem for easy computation. The next chapter will focus on analyzing the water supply path from the dam worksite to Nsima Township, Kumasi Metro using Prim's algorithm.

CHAPTER FOUR

DATA COLLECTION AND ANALYSIS

4.0 Introduction

Ghana Water Company Limited (GWCL), is a legal public utility company that is put in charge. of urban and rural water supply for public, domestic, and industrial purposes. In addition to that the company is responsible for the establishment, operation, and control of sewerage systems. Most communities lack potable water. This is because most of the areas do not have pipe network. Most of the pipe network systems have seen little or practically no maintenance since its establishment, a period of not less than fifteen years. Some of these pipes are broken while others are rusted, which is detrimental to human health. These pipes therefore need to be replaced by new ones. Considering the current economic state of the country and the rapid population growth, it is prudent that more efficient scientific method is put in place to solve these problems. This chapter focuses on computational procedure, data analysis and finding the optimal solution to this problem.

4. 1 Computational Procedure and Analysis of data from the company

The map of the Nsima Township shares boundary with Kokode, Nwamase, Kromoase and Denkyemuoso. The distances covered by the pipe network are indicated on the map in meters.

The map is sketched below;

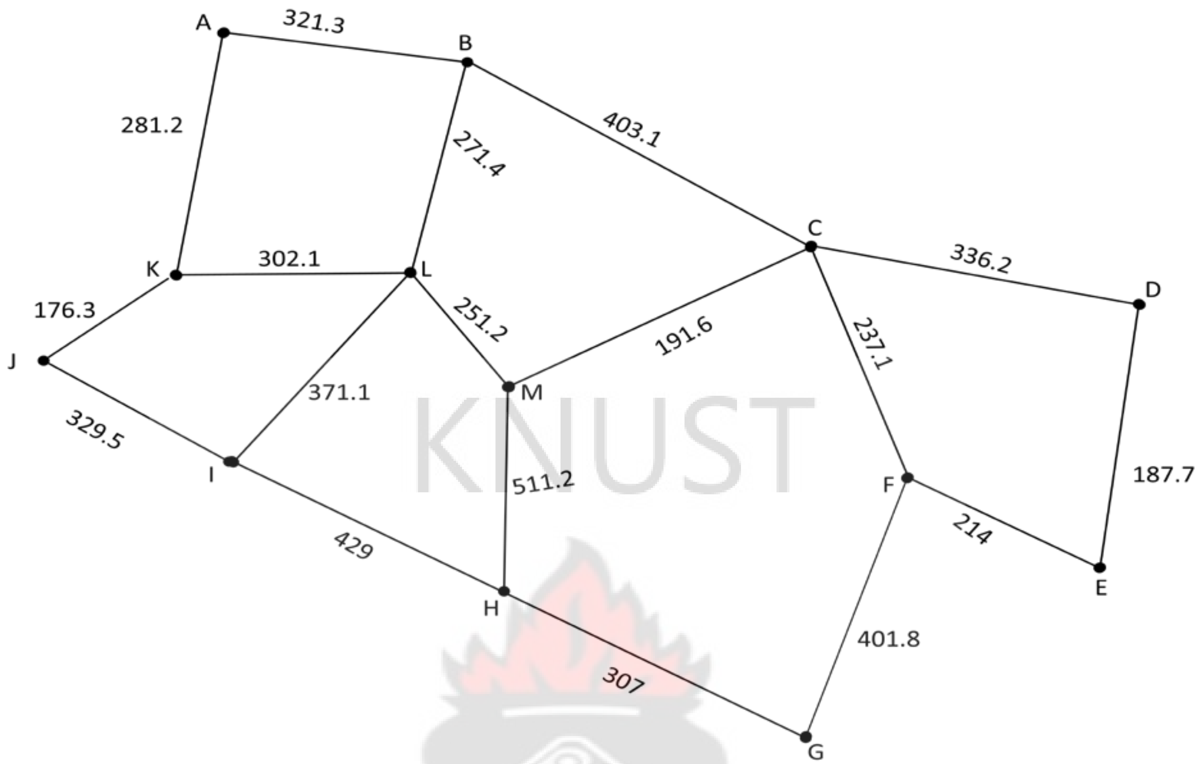


Fig. 4.2: Pipe network at Nsima in metres

Source: Ghana Water Company Limited, Geographical Information services (Not drawn to scale)

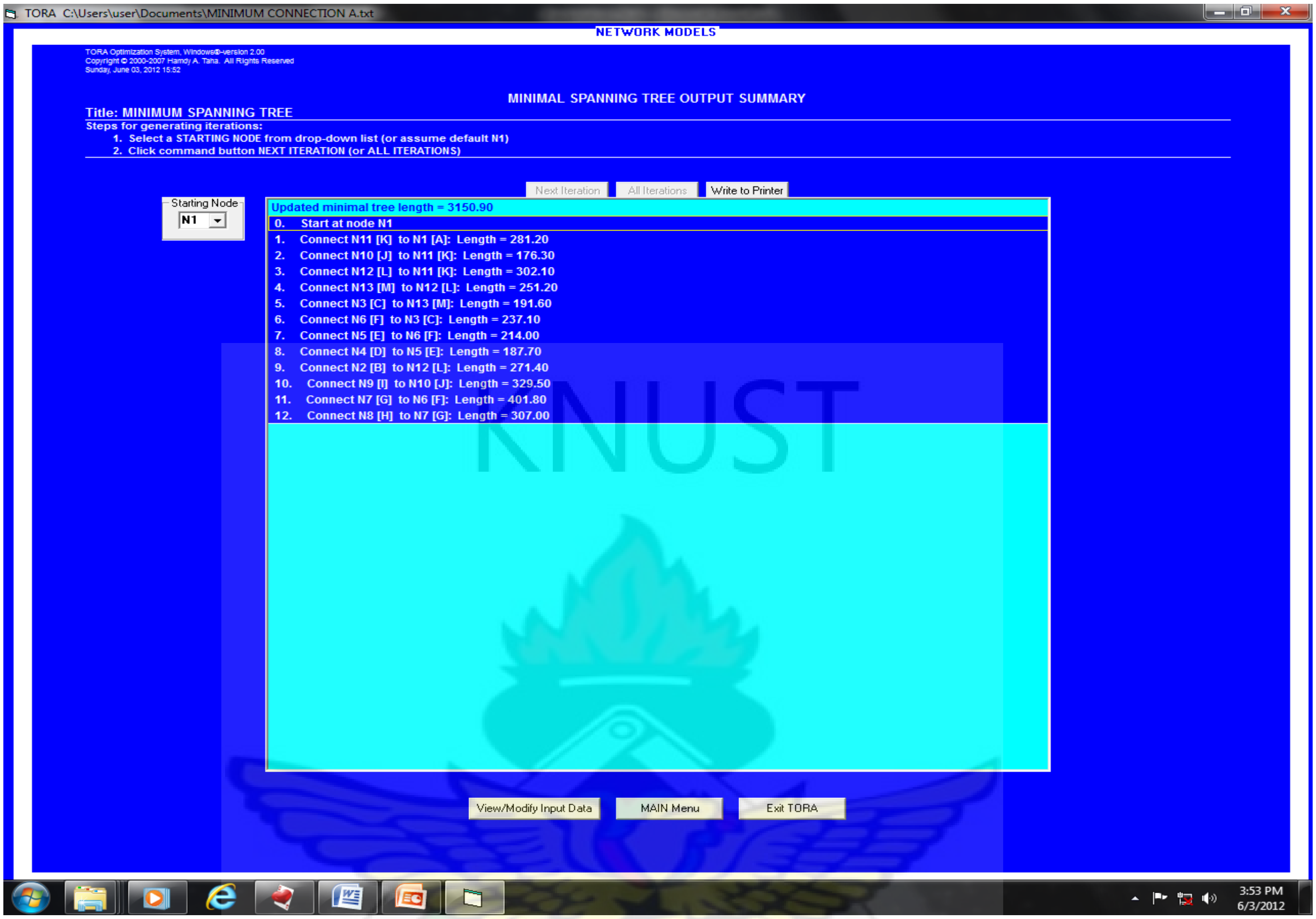
The pipes used to build these network systems varies in diameters and length (distances). Barekese dam is the main source of water earmarked to serve Nsima community. The main pipe that comes from the source (Barekese dam) has a diameter 24centimetres. Water is pumped through this to the company’s station at Suame. Afterwards water distribution starts to the various parts of the region using the 18 centimetre pipe. This pipe enters Nsima through Krodadaam, a surburb of Nsima., 15centimeters pipes are used to supply water to the main areas before smaller pipes are connected to the various homes, schools, churches, institutions and hospitals.

The map consists mainly of the 15 centimetre diameter pipes. The smaller pipes are not captured by the map and are not part of the research work. The researcher worked his way to minimize the distances covered by the pipe network of the Township.

Using the data on the maps above, we then formulate the shortest path problem to minimize the total distance covered by the pipe network system of the Nsima Township. The distances covered by the pipe network in meters are put in a table form first before the Prim's algorithm is applied. The table is shown in appendix i.
In addition its solution done manually is also in appendix ii

Results from a software (Tora)

- A software (tora) for windows 2 analysis was used to analyse the data for laying pipes of the network
- Intel (R) Celeron (R) M processor @ 1.40 GHz, RAM 760MB.
- OS: Windows XP computer was used.
- The surburbs (Nodes) of Nsima of Nsima were represented as follows:
- Roman (A), SDA (B) Foriwaa (C) , Kurodadaam (B), Agric School (E), Atwima (F), Nana Boadu (G), Asieye (H), Station (I), Methodist (J), Market (K), Adade(L), and Ahenfie (M)



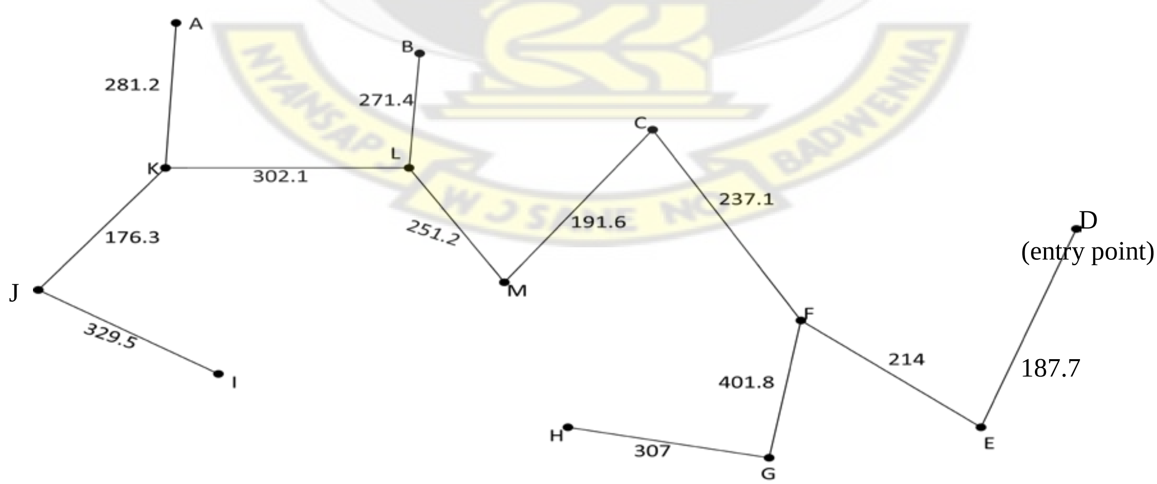


Fig.4.3. Solution layout using prim's Algorithm.

Total distance covered by the minimum spanning tree is 3150.9 metres

Table 4.3: Shows the distance between the pair of nodes starting from the entry node to the last node.

NODES	DISTANCE
D → E	187.7 Metres
E → F	214.0 Metres
F → G	401.8 Metres
G → H	307.0 Metres
F → C	237.1 Metres
C → M	191.6 Metres
M → L	251.2 Metres
L → B	271.4 Metres
L → K	302.1 Metres
K → A	281.2 Metres
K → J	176.3 Metres
J → I	329.5 Metres
TOTAL	3150.9 Metres

4. 2. Discussion

Table 4.1 in appendix 1 represents the pipe network of Ghana Water Company limited system at Nsima Township. The alphabets represent nodes at the main destinations of water supply in the township. From the table, distances between nodes are indicated by numerals. The infinity sign shows where there is no direct link between nodes. The total distance covered by the layout of the network is 3150.9 metres.

The pipes under consideration are the 15 centimeters diameter used to supply water from the main source 18 centimeters diameter pipe. The current price for the 15 centimeters diameter pipe is GH¢166.00 for 6 meter long pipe. The total price for the layout can be computed as follows;

$$\text{Total Cost} = \frac{\text{Total distance}}{6} \times \text{GH¢ } 166.00 = \text{Total Price.}$$

$$\therefore \text{ Total cost of layout} = \frac{3150.9}{6} \times \text{GH¢ } 166.00$$

$$= \text{GH¢ } 87174.90$$

4.3 Analysis

The layout which was developed as a result of using Prim's Algorithm gave a total distance of 3150.9 metres

15 centimeter diameter pipes cost GH¢166.00 for 6 metre long pipe.

Total cost of pipes for the layout is GH¢ 87174.90

The implication of the findings is that the cost of the layout as shown above is the lowest.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATION

5.0 Introduction

This chapter discusses the summary, the conclusion and recommendations of the study.

5.1 Summary

In view of the findings in chapter four, it is deduced that the shortest path problem especially Prim's algorithm can be used to reduce the network systems. The Nsima Township falls under the West district of the Ghana Water Company Limited in the Kumasi Metropolis. The township has water supply problems mainly due to the lack of pipe network. The Prim's algorithm which is good for both small and large network systems was used for the computation of data. Considering the original map, total distance covered by the pipe network system was the minimum, following the method that was applied. Minimizing the pipe network also means reduction in cost, labour and time as well.

5.2 Conclusion

The layout for the pipe network starts from Kurodadaam (D) where the main source of water supply enters the town. This layout passes through Agric School area (E) and is joined to Atwima (F). We have two outlets from Atwima. One of them leads to Nana Boadu’s area (G) and ends at Asieye area (H), while the other passes through Foriwaa (C),Ahenfie (M) to Adade’s area (L) respectively, where it branches. One of them leads to S. D. A (A) and the other to the market (K) where it branches. One of them leads to

Roman (B). The other one passes through Methodist area (J) and finally end at Station I. That is the pipe lines were modeled as a minimum connector. This was possible as a result of making use of prim's algorithm which minimized the number of pipelines. The amount ¢ 87174.90 being the cost of laying pipes for Nsima Township is the minimum due to the method applied.

5.3 Recommendations

- The services of an operational researcher must be employed in networking companies
- Network companies should embrace shortest path Algorithm to plan their network system.
- Network companies can reduce cost and maximize profit using efficient methods
- Network companies should make data available for researchers since they stand the chance of gaining from their work.

REFERENCES

1. Ahmed, E.E. et al (2010). Water allocation and optimization of minor canal operation, Agriculture and Biology Journal of North America. ISSN Print2151-7517(http://www.scihub.org/ABJNA)

2. Ahuja R. K,. Magnanti T. L., and Orlin, J. B. (1993). Network Flows:"Theory, Algorithms and Applications. Prentice Hall. ISBN 0-13-617549-X.

3. Arms, K. (2008). Environment: Problems and Solution, Second Revise Edition. S. Chand and company Ltd. New Delhi. ISBN: 81-219-1654-2

4. Ayoade, J.O. (1975). Water Resources and their Development in Nigeria Hydrological Sciences-Bulletin, 581-591.

5. Biscos, C et al (2003). Optimal operation of water distribution networks by predictive control using MINLP. Pollution Research Group, School of Chemical Engineering, University of Natal, Durban, 4041, South Africa.

6. Cech, T. V. (2005) Principles of Water Resources: History, development, Management, and Policy. 2nd Edition. John Wiley and Son, Inc, Hoboken. ISBN:

7. Chen, Danny Z. (1996). "Developing algorithms and software for geometric path planning problems". *ACM Computing Surveys* 28

8. Cherition D. and Tarjan R. E(1976): *Finding minimum spanning trees.* In: *SIAM Journal on Computing, 5* pp. 724-741.

9. Cherkassky B.V, Goldberg A.V, Radzik T (1996): *Shortest paths algorithms: theory and experimental evaluation,* Mathematical programming.

10. Cormen T. H., Leiserson C. E., Rivest R. L., and Stein C, ed (2001). "All-Pairs Shortest Paths". Introduction to Algorithms (2nd ed.). Cambridge, Massachusetts.

11. Dijkstra E.W and Scholten G. S. (1980) *Termination detection for diffusing computations.* Info. Proc. Lett., 11(1):1-4,

12. Divoky, James J., Hung, Ming S (1990). *Performance of shortest path algorithms in network flow problems.* Institute for Operations Research and the Management Sciences.

13. Doe, W. H (2007) (PDF), *Assessing the Challenges of Water Supply in. Urban Ghana: The case of North Teshie. (EESI Master Thesis),* Stockholm: Department of Land and Water Resources Engineering, Royal Institute of Technology (KTH).

14. Georgiou, P. and Papamichail, D. (2008). Optimization modelof an irrigation reservoir for water allocation and crop planning under various weather conditions. Irrigation Science, Volume 26, number 6, 487-504(18)

15. Komives, K.; Akanbang, B.;Thorsten, R.; Tuffuor, B.; Wasser. W; Larbi, E; Bakalian, A; Whittington, D. (2008). "Post-construction Support and the Sustainability of Rural Water Projects in Ghana". *Paper presented at the 33rd WEDC International Conference - Access to Sanitation and Safe Water: Global Partnerships and Local Actions* (Accra). p. 2-4.

16. Lang, Z. X., Horne, R. N., Stanford. U (1983). Optimum Production Scheduling Using Reservoir Simulators: a comparison of a linear Programming and Dynamic Programming Techniques. SPE Annual Technical Conference and Exhibition, San Francisco, California.

17. Li, C. and Maini, P. K. (2005). An evolving network model with community structure. Journal of Physics A: Mathematical and General, 38 (45). pp. 9741-9749.

18. Manner, J.W. and McBride R. D. (2000). Decomposition – Based Pricing Procedure for Large – Scale Linear Programs: An Application to the Linear Multicommodity Flow Problem Management Science vol. 46, No. 5,693 – 709.

19. Mcbride, R. D. (1998). Advanced in solving the multicomodity flow problem.(http://interfaces.journal.informs.org/content/28/2/32.abstract).

20. Nyarko, K. B.; Oduro-Kwarteng, S.; Adama, I. (2006). "Cost recovery of community-managed piped water systems in the Ashanti region, Ghana". *Water and Environmental Journal* (Chartered Institution of Water and Environmental Management (CIWEM)).

21. Pettie and Vijaya R. and Srinath S., 2002. Experimental Evaluation of a New Shortest Path Algorithm (Extended Abstract)

22. Prim R. C (1957): Shortest connection networks and some generalizations. In: Bell System Technical Journal, 36 , pp. 1389-1401

23. Public utilities Regulatory Commission (PURC 2005). Social Policy Strategy for regulation ,(www.purc.com.gh/social policy 21 01 05.pdf)

24. Rabbani M, Tavakkoli-Moghaddam R, Parsa H (2008) International Journal of Manufacturing Technology and Management. Vol. 15, No.1 pp. 1 – 11

25. Robert G. Gallager R. G. (1982) Distributed Minimum Hop Algorithms. Technical Report LIDS-P-1175, MIT Lab. for Information and Decision Systems.

26. Samani, H. M. V. and Mottaghi, A. (2006) Optimization of water distribution networks using Integer Linear Programming. J. Hydr. Engrg. Volume 132, Issue 5, 501-509.

27. Tongaat Hulett Developments, (THD, 2007). Sustainable Water. Retrieved February 5, 2009 from (http/www.thdev.co.za/content/view/56/2) cited Research Journal of Environmental and Earth Sciences 1(1):16.

28. University of Minnesota Extension Services (UMES, 2008). Ensuring a Safe Water Supply. Retrieved May 6, 2008 from (http://www.extension.umn.edu/distribution/naturalresources/components/DD694d.htm)

29. World Health Organization; UNICEF (2006). "Joint Monitoring Programme for Water Supply and Sanitation. Coverage Estimates Improved Drinking Water".

30. Yang, W. et al (2008). Two-phase flow simulation of reactor clarifiers. Journal of the Chinese Institute of Chemical Engineers, Volume 39, Issue 3, pp. 275-280.

31. Ziliaskopoulos, K. A.(2000). A Linear Programming Model for the Single Destination System Optimum Dynamic Traffic Assignment Problem. Transportation Science.Vol. 34, No.1,37-49.

APPENDIX 1

Table 4.1 Information on Fig. 4.1

	A	B	C	D	E	F	G	H	I	J	K	L	M
A		321.3									281.3		
B	321.3		403.1									271.4	
C		403.1		336.2		237.1				189.2			191.6
D			336.2		187.7								
E				187.7		214.0				365.2			
F			237.1		214.0		401.8						
G						401.8		307.0					
H							307.0		429.0				511.2
I								429.0		329.5		371.1	
J	439.5		189.2						329.5		176.3		
K	281.2									176.3		302.1	
L									371.1		302.1		251.2
M			191.6					511.2				251.2	

APPENDIX 2

Table 4.2 Information on Fig. 4.2

10 ↓ 7 ↓ 4 ↓ 1 ↓ 2 ↓ 3 ↓ 12 ↓ 13 ↓ 11 ↓ 9 ↓ 8 ↓ 6 ↓ 5 ↓

	A	B	C	D	E	F	G	H	I	J	K	L	M
A	∞	321.3	∞	∞	∞	∞	∞	∞	∞	281.2		∞	∞
B	321.3	∞	403.1	∞	∞	∞	∞	∞	∞	∞		271.4	∞
C	∞	403.1	∞	336.2	∞	237.1	∞	∞	189.2	∞		∞	191.6
D	∞	∞	336.2	∞	187.7	∞	∞	∞	∞	∞		∞	
E		∞	∞	187.7	∞ 214	∞	∞	∞	∞	36.52		∞	
F	∞	∞	237.1	∞	214	∞	401.8	∞	∞	∞		∞	
G	∞	∞	∞	∞	∞	401.8	∞	307	∞	∞		∞	
H	∞	∞	∞	∞	∞	∞	307	∞	42.9	∞	∞		511.2
I	∞	∞	∞	∞	∞	∞	429	∞	329.5		371.1		
J	439.5	∞	189.2	∞	∞	∞	∞	329.5	∞	176.3		∞	∞
K	281.7	∞	∞	∞	∞	∞	∞	∞	176.3	∞		302.1	
L	∞	271.4	∞	∞	∞	∞	∞	371.1	∞	302.1	∞	251.2	
M	∞	∞	191.6	∞	∞	∞	511.2	∞	∞	∞		251.2	∞