

**KWAME NKRUMAH UNIVERSITY OF SCIENCE
AND TECHNOLOGY, KUMASI.**

**CONSTRUCTION HEURISTIC FOR THE INSPECTION OF ELECTRICITY
METERS
(CASE STUDY: KOFORIDUA MUNICIPALITY, EASTERN REGION, GHANA)**

BY

DAVID ASIOME KWEKU BEDU



**A thesis submitted in partial fulfilment of the requirements
For the degree of Master of Science in Industrial Mathematics
DEPARTMENT OF MATHEMATICS
SCHOOL OF GRADUATE STUDIES**

JUNE, 2012

DECLARATION

I hereby declare that this submission is my own work towards the award of Master of Science (MSc) in Industrial Mathematics and that to the best of my knowledge, it contains no material previously published by another person nor material which has been accepted for the award of any other degree of the University, except where due acknowledgement has been made in the text.

KNUST

DAVID ASIOME KWEKU BEDU

(PG: 4063910)

Name of Student and ID

Signature

Date

Certified by:

DR. S. K. AMPONSAH

Supervisor

Signature

Date

Certified by:

K. F. DARKWAH

Head of Department

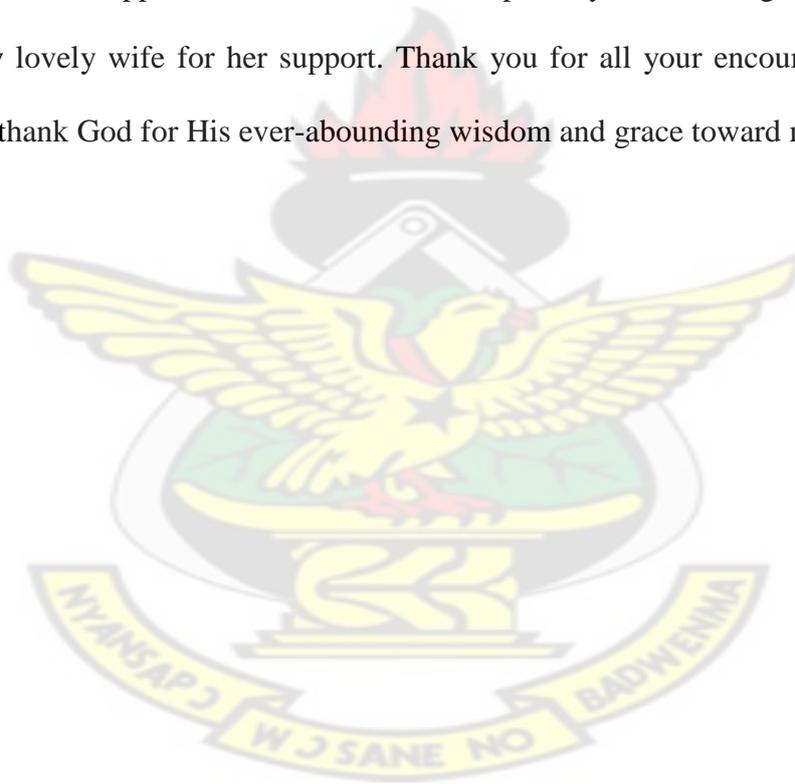
Signature

Date

ACKNOWLEDGEMENT

I would like to acknowledge the advice and guidance of Dr S. K. Amponsah for all the helpful discussions and advice during the preparation of this thesis. I would also like to thank the other lecturers in the Department of Mathematics who have taught me during my study for the degree of Master of Science at Kwame Nkrumah University of Science and Technology.

Also, I would like to express my gratitude to all my family members and my friends who have always been supportive of what I do. I especially acknowledge Charmain Yayra Ahiabor; my lovely wife for her support. Thank you for all your encouragement support. Above all, I thank God for His ever-abounding wisdom and grace toward me throughout my study.



ABSTRACT

Chinese postman problems (CPP) heuristic algorithm is used for several real-world route inspection problems, such as street sweeping, mail delivery, solid waste collection, street watering among others. They can however, be modelled as CPP with some peculiar constraints. As a part of the preventive maintenance programme for the electricity service providers, this study employs the CPP heuristic algorithm to the inspection of electricity meters along the streets in Koforidua municipality as an undirected network. A formal definition of the CPP is presented. The heuristic procedure consists of cluster first, route second method. The Dijkstra's algorithm is initially implemented with Matlab programme to find the optimal distance to be covered in each route and the optimal route is found with the Fluery's algorithm. The construction heuristic algorithm is proposed which gives near optimal feasible solutions and applied to find the optimal inspection route for four (4) main inspection blocks within the municipality. The adoption of the proposed heuristic in Koforidua resulted in an optimal distance for the four inspection blocks as follows 6.17km in block-1, 9.015km in block-2, 9.955km in block-3 and 12.172km in block-4. The results revealed a good performance of the proposed heuristic method for any route inspection problem for Koforidua township.

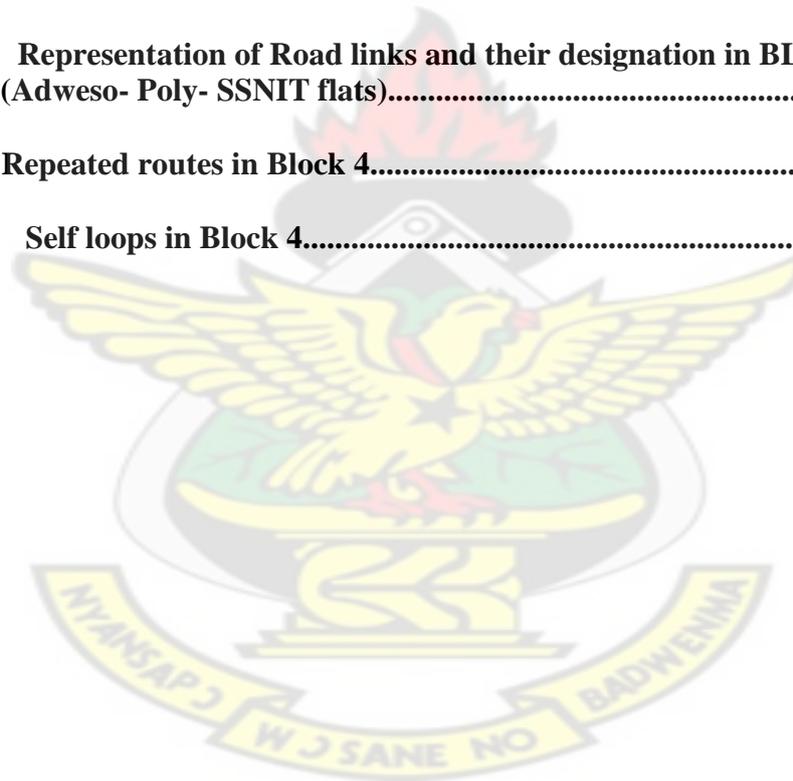
TABLE OF CONTENTS	PAGE
Declaration	ii
Acknowledgements	iii
Abstract	iv
List of Tables	viii
List of Figures	ix
CHAPTER 1.....	1
1.0 INTRODUCTION	1
1.1 Background of the Study.....	1
1.1.2 Background of study area: Koforidua Municipality	2
1.1.3 Types of Electricity Meters	2
1.2 Statement of the Problem.....	3
1.3 Objectives of the Study.....	4
1.4 Methodology.....	4
1.4.1 Synopsis of some relevant heuristic algorithms	5
1.4.2 Greedy heuristics	5
1.4.3 Local search.....	6
1.4.4 Simulated annealing	6
1.4.5 Tabu search.....	7
1.4.6 Genetic algorithm	7
1.4.7 The Travelling Salesman Problem algorithm	8
1.4.9 The Chinese Postman Problem algorithm.....	8
1.6 The Scope of the Study	10
1.7 Limitations of the study.....	10
1.8 Organization of the Study	10
1.9 Summary of the chapter.....	11
CHAPTER 2.....	12
2.0 LITERATURE REVIEW	12

2.1 The Tabu Search method	12
2.2 Genetic Algorithms and Evolutionary Algorithms	13
2.3 Ant Colony Optimisation	15
2.4 Place of meta-heuristics in a classification of the optimization methods	16
2.5 The Travelling Salesman Problem	19
2.6 Capacitated Arc Routing Problems – (CARP).....	22
2.7 The Chinese Postman Problem.....	24
2.7.1 Min-max k-Chinese postman problem.....	27
CHAPTER 3.....	31
3.0 METHODOLOGY	31
3.1 Chinese Postman Problem Formulation and Notation.....	31
3.2 The graphic model	31
3.2.1 Description of the algorithm	32
3.2.2 Local Search Framework.....	33
3.2.3 Applying Local Search Moves	33
3.3 Mathematical formulations for the route inspection problem.....	34
3.3.1 Definitions:	34
3.3.2 Observations:	36
3.3.3 Theorems:.....	37
3.3.4 Route inspection heuristic ‘algorithm’ for optimal tour distance	37
3.4 Proposed mathematical heuristic model for electricity meter inspection.....	41
3.5 Model assumptions	42
3.6 Distinguishing feature of the proposed model:	43
3.6 General CPP Algorithm.	45
3.7 CPP Algorithm for Undirected Graph	45
3.7. 1 Obtaining the optimal tour route using Fleury’s Algorithm for a graph with only even vertices and one nontrivial component	46
3.7.2 Obtaining the optimal tour route using Fleury’s Algorithm for Graph with at most two odd vertices and one nontrivial component.....	46

3.8 Multi route Chinese Postman Problem.....	48
3.8.1 Solving the Constrained Chinese Postman - CCPP problem.....	49
3.8.2 The Constrained Chinese Postman Problem– (CCPP) Algorithm.....	49
CHAPTER 4.....	52
4.0 DATA COLLECTION AND ANALYSIS	52
4.1 Representation of the road layout in Koforidua township:	52
4.2 Formulation of the Constraints Chinese Postman’s Problem (CCCP).....	54
4.3 Analysis of the Multi Graph and Results.....	58
4.3.1 Analysis of the Multi graph and Results for BLOCK 1 (CBD)	58
4.3.2 Analysis of the Multi graph and Results for BLOCK 2	61
CHAPTER 5.....	69
DISCUSSIONS, CONCLUSION AND RECOMMENDATIONS	69
5.0 INTRODUCTION	69
5.1 DISCUSSION.....	69
5.2 CONCLUSION	70
5.3 RECOMMENDATIONS.....	72
REFERENCES	73
APPENDICES.....	77
APPENDIX A.....	77
Matlab code for Dijkstra’s Algorithm	77
APPENDIX B.....	78

LIST OF TABLES

Table	Page
Table 4.31 Representation of Road links and their designation for Block 1	60
Table 4.32 the optimal repeated paths for Block 1.....	61
Table 4.33 Representation of Road links and their designation for Block 2.....	63
Table 4.34: optimal repeated paths for Block 2.....	63
Table 4.35: Representation of Road links and their designation in BLOCK 3 (Galloway- Atekyem Suburbs).....	65
Table 3.36: Repeated route in the Block 3 (Atekyem - Galloway).....	66
Table 4.37: Self loops in Block 3.....	66
Table 4.38: Representation of Road links and their designation in BLOCK 4 (Adweso- Poly- SSNIT flats).....	68
Table 4.39: Repeated routes in Block 4.....	68
Table 4.40: Self loops in Block 4.....	69



LIST OF FIGURES

Figure	page
Figure 3.1: A typical representation of a multigraph	36
Figure 3.2 : Multi graph with only two odd vertices (Semi-Eulerian).....	38
Figure 3.3: Finding the path to be repeated in a multigraph of two odd vertices.....	49
Figure 3.4: Prototype example for multigraph with odd vertices.....	40
Figure. 3.5 Prototype example for finding the optimal route	47
Figure 3.6: Tentative partitioning of the single – CP tour into three approximate equal sub tour.....	50
Figure 3.7: Final partitioning of the single CP tour into three approximate subtour... 	51
Figure 4.1: Lay out of the Koforidua Municipality.....	53
Figure 4.2 Lay out of the road network divided into four subtours.....	54
Figure 4.3 line and node diagram for the central business district (CBD) – Koforidua..	59
Figure 4.4: line and node diagram for Old Estate – and Kenkey Factory.....	62
Figure 4.5: line and node diagram for Atekyem - Galloway Road Layout.....	64
Figure 4.6 line and node diagram for Adweso, Koforidua Polytechnic and SSNIT Flat	68
Figure 5.1: Map of Koforidua.....	78

CHAPTER 1

1.0 INTRODUCTION

1.1 Background of the Study

A prepaid meter is an integrating instrument used for electrical energy billing. It is installed by Electricity distribution companies to be used in measuring the amount of electrical energy consumed by a consumer over a period (wikipiedia, 2011).

By their nature, electrical values cannot be measured by direct observation. Therefore, some property of electricity must be used to produce a physical force that can be observed and measured. To ensure uniformity and accuracy, electric meters need regular inspection and calibration as part of the elements of preventive maintenance engineering, according to the accepted standards of measurement for the given electrical unit such as volt, ampere, ohm, and watt, (Microsoft Encarta 2009)

The sole electricity distribution firm in southern Ghana is the Electricity Company of Ghana (ECG) while in the northern part of Ghana; it is Northern Electrification Development (NED). These companies are the only ones mandated to install and maintain electricity meters.

In Ghana, these meters are mostly used in the urban areas to monitor electricity consumption. According to ECG prepaid department of Koforidua approximately 40,000 as at August 2011 prepaid meters are installed in Koforidua, households, offices, markets and small businesses; for both commercial and domestic energy consumption billing.

1.1.2 Background of study area: Koforidua Municipality

Koforidua is the administrative capital of the eastern region of Ghana, it has a total population of 170,000 (Regional Statistical Services, 2011). The businesses in the municipality are mostly small and medium scale enterprises (SMEs). The municipality has 20 second cycle institutions, 35 basic schools, 1 college of education, 2 nursing training schools and 1 polytechnic and 1 university. All of these institutions make use of electricity and have either the prepaid or credit meters. According to ECG- Koforidua, it is estimated that nearly 40,000 prepaid meters have been installed as at December 2011 and all government establishments still use the credit meters.

It is estimated that the municipality has 2% flats, 5% Semi-detached, 13% detached and about 70% compound houses in the smaller towns (Town and Country Planning – Koforidua - 2011)

1.1.3 Types of Electricity Meters

Formerly, these meters were predominantly electromagnetic. The electromagnetic meters were robust and could span about 20years. They needed little or no repairs but they were read monthly for billing purposes.

Beginning from the year 2004, the ECG as well as NED resolved to change credit meters to electronic prepaid meters. This came as a result of the difficulty in getting customers to pay their electricity bills promptly and to also control consumption. The installation started in metropolitan/municipal/district capitals and the change could continue to other towns. However, due to the fact that the new meters are electronic-based (contains solid state devices that help to read and record consumption pattern for the customer and as well as disconnect the customer

automatically when the credit get exhausted) it needs regular inspection. Unlike the credit meters whose reading and billing was the sole responsibility of the service providers, the prepaid meter now need no reading for billing. But after the installation, the meters need to read, operate and record correctly.

There is therefore the need to monitor the condition, maintain the good operation and inspect to check against illegal connection of unsuspecting customers.

1.2 Statement of the Problem

Unlike the credit meters whose reading and billing were the sole responsibility of the service providers, the prepaid meters now need no reading for billing. But after the installation, the meters need to read, run and record correctly for customer satisfaction.

There is an increasing level of report of faults and wrong operation of these meters to the ECG. But due to the increasing number of prepaid customers the companies are unable to carry out regular and effective inspection on the installed meters hence the need for a tour heuristic to aid the operation.

The inspections of the prepaid meters have financial implication and time constraints. The thesis therefore seeks to propose and construct a tour heuristic that will enable technical personnel of the ECG undertake the inspection of the all prepaid meters installed for customers in the Koforidua municipality; covering the optimum distance thereby taking the least possible time to carry out the exercise.

1.3 Objectives of the Study

Intuitively, the primary thrust of this research is aimed at proposing a mathematical model to find the minimum distance to cover in order to complete the task of inspecting all domestic and commercial electricity meters and a route that accomplishes this goal; or simply to find the optimal tour route of technical inspection personnel for every street/location.

Specifically, the study is set to be carried out with the following aims;

- (i) To demarcate the municipality into smaller geographic areas (clusters or inspection blocks) according to the suburbs in the municipality.
- (ii) To find the estimated distance of roads or streets leading to all junctions/nodes (geographic areas).
- (iii) To come out with a tour construction heuristic that will enable the ECG prepaid metering sectional personnel to inspect these meters while covering an optimum distance.

1.4 Methodology

In this research, realistic mathematical models for the inspection of prepaid meters are formulated and solved using the Chinese postman's algorithm. It goes beyond developing Meta-heuristic to solve simple strategies to optimize the inspection tour.

The idea is to divide the Koforidua Township into inspection blocks which could cover one or more suburbs. With the estimated road distances, a multi graph is created and approximated into a line and node diagram which is used to solve the problem.

The sources of data for the thesis are the internet and libraries for relevant literature, Electricity Company of Ghana (ECG) on current information on electricity meters

and New Juabeng Municipal Assembly is consulted for information on the demarcation as well as distances of the network routes between suburbs and towns within the municipality.

1.4.1 Synopsis of some relevant heuristic algorithms

These are adhoc, trial-and-error methods which do not guarantee to find the optimal solution but are designed to find near-optimal solutions in a fraction of the time required by optimal methods. A heuristic is typically a simple intuitively designed procedure that exploits the problem structure and does not guarantee an optimal solution. Because most of practical problems and many interesting theoretical problems are *NP*-hard, heuristics and approximation algorithms play an important role in solving high level optimization problems.

Such algorithms are used to find suboptimal solutions when the time or cost required to find an optimal solution to the problem would be very large. A meta-heuristic (“meta” means “beyond”) is a general high-level procedure that coordinates simple heuristics and rules to find good approximate (or even optimal) solutions to computationally difficult combinatorial optimization problems. A meta-heuristic does not automatically terminate once a locally optimal solution is found.

(wikipedia.org/wiki/heuristic_algorithm)

1.4.2 Greedy heuristics

These are simple iterative heuristics specifically designed for a particular problem structure. A greedy heuristic starts with either a partial or infeasible solution and then constructs a feasible solution step by step based on some measure of local effectiveness of the solutions. In each iteration, one or more variables are assigned new values by making greedy choices. The procedure stops when a feasible solution

is generated. As an extension of greedy heuristics, a large number of local search approaches have been developed to improve given feasible solutions. (wikipedia.org/wiki/Greedy_algorithm)

1.4.3 Local search

This is a family of methods that iteratively search through the set of solutions. Starting from an initial feasible solution, a local search procedure moves from one solution optimal within a neighboring set of solutions; this is in contrast to a global optimum, which is the optimal solution in the whole solution space solution to a neighboring solution with a better objective function until a local optimum is found or some stopping criteria are met. The next two algorithms, simulated annealing and tabu search, enhance local search mechanisms with techniques for escaping local optima. (wikipedia.org/wiki/local_search)

1.4.4 Simulated annealing

This is a probabilistic meta-heuristic derived from statistical mechanics. This iterative algorithm simulates the physical process of annealing, in which a substance is cooled gradually to reach a minimum-energy state. The algorithm generates a sequence of solutions and the best among them becomes the output. The method operates using the neighborhood principle, i.e., a new solution is generated by modifying a part of the current one and evaluated by the objective function (Corresponding to a lower energy level in physical annealing). The new solution is accepted if it has a better objective function value. The algorithm also allows occasional non-improving moves with some probability that decreases over time, and depends on an algorithm parameter and the amount of worsening. A non-improving move means to go from one solution to another with a worse objective

function value. This type of move helps to avoid getting stuck in local optimum. It has been proved that with a sufficiently large number of iterations and a sufficiently small final temperature, the simulated algorithm converges to a global optimum with a probability close to one. However, with these requirements, the convergence rate of the algorithm is very low. Therefore, in practice it is more common to accelerate the algorithm performance to obtain fast solution approximations. (Russell, 1997)

1.4.5 Tabu search

This is a meta-heuristic technique that operates using the following neighborhood principle. To produce a neighborhood of candidate solutions in each iteration, a solution is perturbed a number of times by rules describing a move. The best solution in the neighborhood replaces the current solution. To prevent cycling and to provide a mechanism for escaping locally optimal solutions, some moves at one iteration may be classified as tabu if the solutions or their parts, or attributes, are in the tabu list (the short-term memory of the algorithm), or the total number of iterations with certain attributes exceeds a given maximum (long-term memory). There are also aspiration criteria which override the tabu moves if particular circumstances apply. (Wikipedia.org/wiki/Tabu_search)

1.4.6 Genetic algorithm

These are probabilistic meta-heuristics that mimic some of the processes of evolution and natural selection by maintaining a population of candidate solutions, called individuals, which are represented by strings of binary genes. A genetic algorithm starts with an initial population of possible solutions and then repeatedly applies operations such as crossover, mutation, and selection to the set of candidate solutions. A crossover operator generates one or more solutions by combining two

or more candidate solutions, and a mutation operator generates a solution by slightly perturbing a candidate solution. Thus, the population of solutions evolves via processes which emulate biological processes. The basic concept is that the strong species tend to adapt and survive while the weak ones tend to die out.

(lancet.mit.edu/~mbwall/presentations/IntroToGAs)

1.4.7 The Travelling Salesman Problem algorithm

The Travelling Salesman Problem (TSP) is a problem in combinatorial optimization studied generally in some field of engineering, operations research and computer science. Given a number of nodes/location/ports/cities and their pair wise distances, the major task is to find the shortest possible tour that visits each node/location/port/city exactly once. The Travelling Salesman Problem (TSP) has been studied during the last five decades and many exact and heuristic algorithms have been proposed and used to solve problem which otherwise have no direct ways of having an optimal solutions. Notable among such algorithms used include construction algorithms, iterative improvement algorithms, branch-and-cut exact branch-and-bound and algorithms and many meta-heuristic algorithms, such as tabu search (TS), simulated annealing (SA), genetic algorithm (GA) and ant colony (AC) (Russell, 1997)

1.4.9 The Chinese Postman Problem algorithm

The Chinese Postman Problem (CPP) is a close heuristic to TSP. In this routing problem the traveller must traverse every arc (i.e. road link) in the network. The name comes from the fact that a Chinese mathematician, Mei-Ko Kwan in 1962, developed the first algorithm to solve this problem for a rural postman. It is an

extension to one of the earliest graph theory questions, the Königsberg Bridge Problem (KBP), which was studied by Euler in 1736.

In general, graph theorists are interested in understanding whether or not a circuit exists that does not require traversing the same arc twice. Operations researchers are interested in finding the shortest route in any type of network.

The Chinese Postman class of problems is relevant to a number of other services. Garbage collection, street sweeping, salting or gritting of icy roads, and snow plowing are some of the other services for which CPP algorithms have been applied. Meter readers also must travel up and down every street. Checking roads for potholes or serious deterioration or checking pipelines for weak spots also fall into this class of problems.

1.5 Significance of the Study

This study introduces a more proactive approach in the dealing with the preventive maintenance of prepaid meters. An algorithm that proposes a good monitoring of prepaid meters will help increase the response to faults and also customer satisfaction.

One major setback of the energy distribution sector is the issue of illegal connection in markets in the central business district. With this optimum tour route found with this construction heuristics, inspection could be more regular and this menace can easily be checked to save the company revenue.

1.6 The Scope of the Study

The study partly parallels the travel salesman problem but more closely, it mimics a type of the Chinese Postman Problem (CPP), the heuristic procedure consists of cluster first, route second method. It is carried out without time constraints with a case study covering the Koforidua Municipality in the Eastern region of Ghana.

1.7 Limitations of the study

- (i) Unplanned nature of the settlement in the town makes the determination of the distances very difficult.
- (ii) Testing alternative algorithms to select efficient ones, etc. The limited time at the researcher's disposal lead to not many items being reviewed as literature. The quality of data depends not only on the amount of time one spends in gathering them but partially on how much money one is prepared to spend in gathering them.

The researcher also encountered certain difficulties in connection with data collection. Other information, like the distance at a particular point with reference to a given geographical direction, as well as proper designation (naming) of streets or road links in the Koforidua municipality.

1.8 Organization of the Study

The study is organized in five chapters as follows. Chapter one provides general background issues of the study. It also provides the statement of problem and it sets out the objectives of the study, provides the significance of the study as well as the scope of the study.

Chapter two reviews pertinent literature related to the study including travelling salesman's problem, optimization and adaptive techniques, meta-heuristics and

waste routing heuristics. Chapter two also describes systems the different meta-heuristic algorithms for the study. It also describes mathematical algorithm for solving heuristics problems. Chapter three discusses the methodological issues of the study. Chapter four discusses the empirical results and interprets the results. The final chapter, which is chapter five, summarizes the main findings of the study and provides suggestions and recommendation.

1.9 Summary of the chapter

In this chapter, the following has been presented; the background of the study, the statement of the problem, the objectives of the study, the methodology, the significance of the study, the scope of the study and finally present the organization of the study.

The next chapter takes a looks closely at some relevant heuristics that has been applied in the study of route optimisation problem. These have given rise to a number of algorithms used to solve harder optimization problems. Indeed, a good number of publications regarding route optimisation, the old time, the travelling salesman problem (TSP), transportation problem, bin packing and garbage collection in waste management. The chapter ultimately looks in a greater detail, the recently proposed Chinese postman's problem and its associated algorithm.

CHAPTER 2

2.0 LITERATURE REVIEW

The study of heuristics has given rise to a number of algorithms used to solve harder optimization problems. Indeed, a good number of publications regarding route optimisation, the old time, the travelling salesman problem (TSP), transportation problem, bin packing and garbage collection in waste management.

Four widely known meta-heuristics are applied in finding near neighbourhood solutions optimal in all optimisation problem involving heuristics, notable among these are, Tabu search, the evolutionary algorithms, the simulated annealing method, Ant colony algorithms. Each one of these meta-heuristics is actually a family of methods.

2.1 The Tabu Search method

The method of search with tabus, or simply tabu search or tabu method, was formalized in 1986 by Glover (Glover, 1986). Its principal characteristic is based on the use of mechanisms inspired by the human memory. The tabu method takes, from this point of view, a path opposite to that of simulated annealing, which does not utilize memory at all, and thus is incompetent to learn the lessons from the past. On the other hand, the modelling of the memory introduces multiple degrees of freedom, which opposes — even in the opinion of the author (Glover and Laguna, 1997) — any rigorous mathematical analysis of the tabu method. The guiding principle of the tabu method is simple: like simulated annealing, the tabu method at the same time functions with only one “current configuration” (at the beginning, any solution), which is updated during successive “iterations”. In each iteration, the mechanism of passage of a configuration, called s , to the next one, called t ,

comprises of two stages: one builds the set of the neighbours of s , i.e. the set of the accessible configurations in only one elementary movement of s (if this set is too vast, one applies a technique of reduction of its size: for example, one utilizes a list of candidates, or one extracts at random a subset of neighbours of fixed size); let $V(s)$ be the set (or the subset) of these neighbours; one evaluates the objective function f of the problem for each configuration belonging to $V(s)$. The configuration t , which succeeds s in the series of the solutions built by the tabu method, is the configuration of $V(s)$. (Glover, 1986).

2.2 Genetic Algorithms and Evolutionary Algorithms

The evolutionary algorithms (EAs) are the search techniques inspired by the biological evolution of the species and appeared at the end of the 1950s (Fraser, 1957). Among several approaches (Holland, 1962) (Fogel et al., 1966) the genetic algorithms (GAs) are certainly the most well known example, following the publication of a book by Goldberg in 1989: Genetic Algorithms in Search, Optimization and Machine Learning. The evolutionary methods initially aroused a limited interest, because of their significant cost of execution. But they have experienced, for the last ten years, a considerable development, that can be attributed to the significant increase in the computing power of the computers, and in particular following the development of massively parallel architectures, which exploit their “intrinsic parallelism” (Fraser, 1957)

The principle of an evolutionary algorithm can be simply described. A set of N points in a search space, chosen a priori at random, constitutes the initial population; each individual x of the population has a certain fitness value, which measures its degree of adaptation to the objective aimed. In the case of the minimization of an

objective function z , the fitness of x will be higher, if $z(x)$ is smaller. An EA consists in evolving gradually, in successive generations, the composition of the population, by maintaining its size constant. During generations, the objective is to overall improve the fitness of the individuals; such a result is obtained by simulating the two principal mechanisms which govern the evolution of the living beings, according to the theory of Darwin. the selection, which supports the reproduction and the survival of the fittest individuals, and the reproduction, which allows mixing, the recombination and the variations of the hereditary features of the parents, to form offspring with new potentialities.

In practice, a representation must be chosen for the individuals of a population. Classically, an individual could be a list of integers for combinatorial problems, a vector of real numbers for numerical problems in continuous spaces, a string of binary digits for Boolean problems, or will be able to even combine these representations in complex structures, if it is required. The passage from one generation to the next one proceeds in four phases: a phase of selection, a phase of reproduction (or variation), a phase of fitness evaluation and a phase of replacement. The selection phase designates the individuals who take part in the reproduction. They are chosen, possibly several times, a priori all the more often as they have high fitness. The selected individuals are then available for the reproduction phase. Operators to copies of the individuals previously selected to generate new individuals; the operators most often used are crossover (or recombination).

2.3 Ant Colony Optimisation

The entomologists analyzed the collaboration which is established between the ants in seeking food outside the anthill. It is remarkable that the ants always follow the same path, and this path is the shortest possible one. This control is the result of a mode of indirect communication, via the environment: the “stigmergy”.

Each ant deposits, along its path, a chemical substance, called “pheromone”. All the members of the colony perceive this substance and preferentially direct their walk towards the more “odorous” or high pheromone-concentrated areas. It results particularly in a collective faculty to find the shortest path quickly, if this one is blocked fortuitously by an obstacle. This behaviour was taken as a starting point to model the algorithm.

The evaporation of pheromone also makes less desirable routes more difficult to detect and further decreases their use. However, the continued random selection of paths by individual ants helps the colony discover alternate routes and insures successful navigation around obstacles that interrupt a route. Trail selection by ants is a pseudo-random proportional process and is a key element of the simulation algorithm of ant colony optimization (Dorigo and Di Caro, 1999).

Dorigo and Di Caro also proposed a new algorithm for the solution of the travelling salesman, problem. Since this research work, the method was extended to many other optimization problems, some combinatorial and some continuous.

The ant colony algorithms have several interesting characteristics which could be adopted; to mention in particular high intrinsic parallelism, flexibility (a colony of ants is able to adapt to modifications of the environment), robustness (a colony is ready to maintain its activity even if some individuals are failing),

the decentralization (a colony does not obey a centralized authority) and the self-organization (a colony finds itself a solution, which is not known in advance). This method seems particularly useful for the problems which are distributed in nature, problems of dynamic evolution, which require a strong fault-tolerance. At this stage of development of these recent algorithms, the transposition with each optimization problem is not however trivial: it must be the subject of a specific treatment, which can be difficult.

- i.) The real ants follow a path between the nest and a source of food.
- ii.) An obstacle appears on the path, the ants choose to turn on the left or right, with equal probabilities; the pheromone is deposited more quickly on the shortest path.
- iii.) All the ants chose the shortest path.

2.4 Place of meta-heuristics in a classification of the optimization methods

In order to recapitulate the preceding considerations, a general classification of the mono-objective optimization methods, already published in (Dr'eo et al, 1998). One finds, that the combinatorial and the continuous optimizations are differentiated;

- i.) for combinatorial optimization, one can approach different methods, when one is confronted with a difficult problem; in this case, the choice is sometimes possible between “specialized” heuristics, entirely dedicated to the problem considered, and a meta heuristic;

- ii.) for continuous optimization, one summarily separates the linear case (which is concerned in particular with the linear programming) from the non-linear case, where the framework for difficult optimization can be found. In this case, a pragmatic solution can be to resort to the repeated application of a local method

which exploits, or not, the gradients of the objective function. If the number of local minima is very high, the recourse to a global method is essential: those meta-heuristics are then found, which offer an alternative to the traditional methods of global optimization, those requiring the restrictive mathematical properties of the objective function;

- i.)** among the meta-heuristics, one can differentiate the meta-heuristics “of neighbourhood”, which make progress by considering only one solution at a time (simulated annealing, tabu search) from the “distributed” meta-heuristics, which handle in parallel a complete population of solutions (genetic algorithms)
- ii.)** Finally, the hybrid methods often associate a meta-heuristic with a local method. This co-operation can take the simple form of a passage of relay between the meta-heuristic and the local technique, with the objective to refine the solution. But the two approaches can also be intermingled in more complex way.

One of the first heuristics addressing the problem of m tours in TSP with some side conditions is due to Russell (1997), although the solution procedure is based on transforming the problem to a single TSP on an expanded graph. The algorithm is an extended version of the Lin and Kernighan heuristic originally developed for the TSP. Another heuristic based on an exchange procedure for the mTSP is given by Potvin et al.

A parallel processing approach to solve the mTSP using evolutionary programming is proposed by Fogel (1990).

The approach considers two salesmen and an objective function minimizing the difference between the lengths of the routes of each salesman. Problems with 25 and

50 cities were solved and it is noted that the evolutionary approach obtained exceedingly good near-optimal solutions. Several artificial neural network (NN) approaches have also been proposed to solve the mTSP, but they are generally extended versions of the ones proposed for the TSP. Wacholder et al. have extended the Hopfield-Tank ANN model to the mTSP but their model has been evaluated to be too complex with its inability to guarantee feasible solutions. Hsu et al. presented a neural network approach to solve the mTSP, based on solving m standard TSPs. The authors state that their results are superior to that of Wacholder et al. A self-organizing NN approach for the mTSP is due to (Punnen 2002), which is based on the elastic net approach developed for the TSP.

Another self-organizing Neural Networks (NN) approach for the multiple Travel Salesman Problem Mtsp is proposed by Goldstein. Describe a self-organizing NN for the VRP based on an enhanced Mtsp NN model.

Recently, Modares et al. Developed self-organizing NN approach for the Mtsp with a min-max objective function, which minimizes the cost of the most expensive route among all salesmen. Their approach seems to outperform the elastic net approach. Utilizing genetic algorithms (GA) for the solution of Mtsp seems to be first. A recent application by Tang et al, uses genetic algorithms to solve the Mtsp model developed for hot rolling scheduling. The approach is based on 18odelled18 the problem as an Mtsp, converting it into a single TSP and applying a modified genetic algorithm to obtain a solution. Yu et al. Also use Gas to solve the Mtsp in path planning.

Mission planning generally arises in the context of autonomous mobile robots, where a variety of applications include construction, military reconnaissance, warehouse automation, post-office automation and planetary exploration. The

mission plan consists of determining the optimal path for each robot to accomplish the goals of the mission in the smallest possible time.

The mission planner uses a variation of the multiple Travel Salesman Problem (Mtsp, where there are n robots, m goals which must be visited by some robot, and a base city to which all robots must eventually return. The application of the Mtsp in mission planning is reported by Brummit and Stentz and in unstructured environments by the same authors.

Planning of autonomous robots is modelled as a variant of the Mtsp by Yu et al, 2002. In the field of cooperative robotics. Similarly, the routing problems arising in the planning of unmanned aerial vehicle applications, as investigated by Ryan et al, can be modelled as an Mtsp with time windows.

J. Dr'eo, et al discusses that the best strategy to approximate the solution of a combinatorial optimisation problem is to couple a constructive heuristics and local search.

2.5 The Travelling Salesman Problem

The Travelling Salesman Problem (TSP) is a problem in combinatorial optimization studied generally in some field of engineering, operations research and computer science. Given a number of nodes/location/ports/cities and their pair wise distances, the major task is to find the shortest possible tour that visits each node/location/port/city exactly once.

The Travelling Salesman Problem (TSP) has been studied during the last five decades and many exact and heuristic algorithms have been proposed and used to solve problem which otherwise have no direct ways of having an optimal solutions. Notable among such algorithms used include construction algorithms, iterative

improvement algorithms, branch-and-cut exact branch-and-bound and algorithms and many metaheuristic algorithms, such as tabu search (TS), simulated annealing (SA), genetic algorithm (GA) and ant colony (AC); which have been discussed above.

Lin and Kernighan (1973) made a great improvement in quality of tours that can be obtained by heuristic methods. Some of the well known tour construction procedures are the nearest neighbour procedure by Ahuja et al, the Clark and Wright savings' algorithm, the insertion procedures, the partitioning approach by Karp and the minimal spanning tree approach by Christofides.

Meta-heuristic algorithms have been applied successfully to the TSP by a number of researchers. SA algorithms for the TSP were developed by, Goldstein, and Nahr et al. etc. The ACO is a relative new metaheuristic algorithm which is applied successfully to solve the TSP. Some work based on SA technology was reported by Dorigo et al (2008)

solved a travelling salesman problem which models the production of printed circuit boards having 7,397 holes (cities), and in 1998, the same authors solved a problem over the 13,509 largest cities in the U.S. For problems with large number of nodes as cities the TSP becomes more difficult to solve.

Feasible exact solutions for the TSP have been found, but there are restrictions on the input sizes. An exact solution was found for a 318-City problem by Crowder and Lawler et al in (1980). The basic idea in achieving this solution involves three phases. In the first phase, a true lower bound on the optimal tour is found. In the second phase, the result in the first phase is used to eliminate about ninety-seven percent of all the possible tours. Thus, only about three percent of the possible tours need to be considered. In the third phase, the reduced problem is solved by brute

force. This solution has been implemented and used in practice. Experimental results by Apple Gate et al (1998) showed that running this algorithm, implemented in the C programming language and executed on a 400MHz machine, would produce a result in 24.6 seconds of running time.

However, none of the algorithms that provide an exact solution for input instances of over a thousand cities are practical for everyday use. Even with today's super computers, the execution time of such exact solution algorithms for TSPs involving thousands of cities could take days.

Computer hardware researchers have been making astonishing progress in manufacturing evermore powerful computing chips. Moore's Law (http://en.wikipedia.org/wiki/Moore's_law), which states that the number of transistors that can fit on a chip will double after every 18 months, has held ground since 1965. This basically means that computing power has doubled every 18 months since then. Thus, we have been able to solve larger instances of NP-hard problems, but algorithm complexity has still remained exponential. Moreover, it is highly speculated that this trend will come to an end because there is a limit to the miniaturization of transistors. Presently, the sizes of transistors are approaching the size of atoms. With the speeds of computer processors rounding the 5GHz mark, and talks about an exponential increase in speeds of up to 100GHz (http://en.wikipedia.org/wiki/Moore's_law), one might consider the possibility of us exceeding any further need of computational performance. However, this is not the case. Although computing speeds may increase exponentially, they are, and will continue to be, surpassed by the exponential increase in algorithmic complexity as problem sizes continue to grow. Moore's law may continue to hold true for another decade or so, but different methods of computing are being researched.

2.6 Capacitated Arc Routing Problems – (CARP)

Waste collection, as most logistic activities, can be studied at different levels: strategical, tactical, and operational. In this work we concentrate on tactical planning, where a vehicle fleet and the service demand are given and the objective is to design the vehicle trips in order to minimize operational costs subject to service constraints.

The operational problem, the definition of collection routes given the vehicle fleet, can greatly benefit computerized support already for medium sized town. While the operational constraints can greatly vary, the core problem can be identified as a capacitated arc routing problem on large directed graphs (DCARP).

Specifically, this work derives from an experience of decision support for the waste collection sectors of municipalities of towns with about 100,000 inhabitants, with the objective of designing vehicle collection routes subject to a number of operational constraints. The reported results are for an abstraction level which does not consider several very specific issues, such as union agreements, third-party or personal contracts, etc.

The problem to solve is modelled as a Capacitated Arc Routing Problem (CARP) on a directed graph and solved accordingly. Two main issues arose:

CARP optimization. The instances to be solved are far bigger than the state of the art ones. Original heuristic approaches had to be designed in order to meet the solution quality and the allowed computation time specifications. Operator interface it was needed to produce a system interface which could be effectively used by a service

operator allowing him to fully understand problem instance and solution details, together with any manual intervention desired.

Dror (2000) did a publication on “Arc Routing: Theory, Solutions and Applications”, the reported results have a relevance beyond the specific application, as several activities of real world relevance can be modelled as CARP, foremost among them are mail collection or delivery, snow removal, street sweeping. The CARP is in fact a powerful problem model, which was originally proposed by Golden and Wong, and which, given its actual interest, have then been studied by many researches. Dror (2000) collected a significant number of applications of CARP variants and of corresponding solution methodologies.

Residential refuse collection requires services at a large number of discrete points. These points are close together and distributed along the arcs. Algorithms for solid waste route are considered to belong to Capacitated Arc Routing Problems (CARP) (Amponsah, 2003). The Capacitated Arc Routing Problems (CARP) arises when arc has associated with it a positive demand and the vehicles to be routed have a finite capacity (Greistorter, 1994). One truck may not be able to service all the roads in a district due to its limited capacity. The CARP is to find a set of routes from a single depot that service all arcs in the network at minimal cost and subject to the constraints that the total demand on each route does not exceed the capacity of the vehicle. The cost of a trip comprises the cost of its serviced arcs and of its intermediate connecting paths. Demands are usually amount of waste to be collected along the streets (urban waste). The techniques combined computer and heuristics approaches. The study took into account the road network detail in finding solution to waste collection problem in undirected network.

2.7 The Chinese Postman Problem

The Chinese Postman Problem (CPP) is a close 'cousin' to TSP. In this routing problem the traveller must traverse every arc (i.e. road link) in the network. The name comes from the fact that a Chinese mathematician, Mei-Ko Kwan in 1962, developed the first algorithm to solve this problem for a rural postman. It is an extension to one of the earliest graph theory questions, the Königsberg Bridge Problem, which was studied by Euler in 1736. In general, graph theorists are interested in understanding whether or not a circuit exists that does not require traversing the same arc twice. Operations researchers are interested in finding the shortest route in any type of network.

The Chinese Postman class of problems is relevant to a number of other services. Garbage collection, street sweeping, salting or gritting of icy roads, and snow plowing are some of the other services for which vehicle routing algorithms have been applied. Meter readers also must travel up and down every street. Checking roads for potholes or serious deterioration or checking pipelines for weak spots also fall into this class of problems.

In the ever complex real-world additional constraints can arise that complicate the search for efficient routes. Labour contracts may require that the routes of different drivers must be approximately of equal length. There may be significant time restrictions or time windows on when a vehicle must visit a specific location to make a delivery or pick-up. The vehicle making pick-ups may also have capacity limitations such as a garbage truck which would restrict the maximum length of a route. Uncertainty can also complicate route planning. Trucks that deliver gasoline

or oil, can't be sure when they set out as to how much they will have to pump into each of the tanks on their route.

Harold (2000) discusses in a directed Chinese Postman Problem, a postman delivering letters in a village may wish to know a circuit that traverses each street (in the appropriate direction if one-way streets), starting and returning to their office. This is a graph theoretic problem: roads are directed edges (arcs), and road junctions are vertices. The postman requires a Chinese Postman Tour, which we abbreviate CPT. The postman probably wants a shortest tour, with few repeated street visits. The cost of a CPT is defined as the total arc weight, summed along the circuit (e.g., the total distance walked). An optimal CPT is a CPT of minimal cost. If some weights are negative, an optimal CPT may not be defined: if there is any circuit with an overall negative weight, the postman could arbitrarily repeat it and get a total cost lower and lower without bound. Conventional applications of the CPP are concerned with routing more generally than postmen, as in routing snow ploughs or planning street maintenance.

Many practical routing problems involve finding paths or cycles that traverse a set of arcs in a graph. In general, we call such problems arc routing problems (ARPs). The aim of solving such problems is to determine a least-cost traversal of a specified arc subset of a graph, with or without constraints. Routing is a decision-making process that plays an important role in most manufacturing and service industries. Billions of dollars are spent each year by governments and private enterprise on these operations. Enormous money is also wasted because of poor planning. Such problems have long been attended by mathematicians and operations researchers.

Another well-known and closely related problem is the so-called Chinese Postman Problem (CPP). The problem was first proposed by the Chinese mathematician, Meigu Guan in 1962. It says that a postman picks up mails at the post office, delivers it along a set of streets, and returns to the post office. Since he must cover over every street at least once, the CPP is referring to investigate how to cover every street and return to the post office under the least cost. In practice, apart from the requirement of travelling all streets, we consider the street direction, number of postmen, etc.

Typical classification of CPP is by street direction, which divides a CPP into three cases Hsiao-Fan and Yu-Pin (2001):

- i.)** undirected CPP, (UCPP)
- ii.)** directed CPP, (DCPP) and
- iii.)** mixed CPP, (MCP)

Although the first two conditions can be found exact solution efficiently, the mixed CPP has been shown to be NP-hard. In the past, most of researches devoted to whether the problem is directed and the service is capacitated. Researches on the practical condition of time-constraints are in a minority. But in present environment of “time is money”, the consideration is necessary. Therefore, in this study, we shall employ the concept of fuzzy set theory such that we can cope with a directed DCPP when time constraints are not certain.

Just like postman delivering letters in a town may wish to know a circuit that traverses each street (in the appropriate direction if one-way streets), starting and returning to the office, the meter inspectors are expected to move along every street; while inspecting the electricity meters of both small scale businesses and household. The inspectors may have to visit some streets more than once. This is obviously a

graph theoretic problem: roads are edges, and road junctions are vertices. The meter inspectors require a Chinese Postman Tour to optimise their route.

This thesis looks closely at one vital and motivating applications of the Chinese Postman Problem; that is the construction of an optimal route tour for the inspection of electricity meters in the Koforidua municipality. The CPP could be seen as a close ‘cousin’ to the TSP. But the difference lie in the fact that CPP is edge oriented heuristic while TSP is node oriented heuristic.

The next chapter zeroes in to provide a derivation of the mathematical formulation of the Chinese Postman Problem for mixed multi-graphs. The technical personnel require an optimal tour route to be traversed while undergoing the inspection without any priorities to any customer along each street.

2.7.1 Min-max k -Chinese postman problem

(Dino and Gerhard, 2005) proposed a tabu search algorithm for the min-max k -Chinese postman problem (MM k -CPP). Given an undirected edge-weighted graph and a distinguished depot node, the MM k -CPP consists of finding $k > 1$ tours (starting and ending at the depot node) such that each edge is traversed by at least one tour and the length of the longest tour is minimized. A special emphasis was put on investigating the trade-off between running time effort and solution quality when applying different improvement procedures in the course of the neighbourhood construction.

Furthermore, different neighbourhoods were analyzed. Their results showed that the tabu search algorithm outperforms all known heuristics and improvement procedures.

In road maintenance, garbage collection, mail delivery, etc. Since usually large road networks have to be serviced, the work load must be distributed among $k > 2$ vehicles. In contrast to the usual objective to minimize the total distance travelled by the k vehicles (k -CPP), for the min-max k -Chinese postman problem (MM k -CPP) the aim is to minimize the length of the longest of the k tours. This kind of objective is preferable when customers have to be served as early as possible. Furthermore, tours were enforced to be more balanced resulting in a fair scheduling of tours.

Although the CPP and the k -CPP are polynomially solvable (Hertz et al, 1999), the MM k -CPP is NP-hard. Hence, the employment of heuristics produces approximate solutions.

(Dino and Gerhard, 2005) presented a tabu search algorithm for the MM k -CPP which outperforms all known heuristics. In many cases, solutions obtained proved to be near-optimal.

Carlsson and Ye (2009), discussed in a paper titled “Practical distributed vehicle routing for street-level map scanning”. In their publications, a four-stage meta-heuristic for routing vehicles in an urban environment, with the goal of traversing every street. While the theoretical aspects of street traversal are well-studied, their algorithm was designed to accommodate several obstacles to practical street traversal that are known to be NP-hard, such as turning penalties, the usage of multiple vehicles, and the presence of one-way and two-way streets. Some encouraging results were presented from a case study.

In their applications, the goal was to traverse every street in a city with a survey vehicle, which typically has a camera attached. However, merely solving CPP gives only a route for a single vehicle, but the novel idea was pivoted on how to distribute the workload between a fleet of vehicles so that instead of treating the problem as an

instance of CPP, which obviously is a poor solution technique in most cases, the problem is rather modelled as a min-max k-CPP (MM k-CPP) which subdivide the entire geographic area into sub-areas and assigned to separate vehicles. So that the algorithm help minimise the maximum tour length undertaken by each vehicle which in the case of this project will be the meter reader.

Although approximation algorithms for the “k-postman problem” exist, they are strictly combinatorial and do not take advantage of the fact that our road map is a planar graph, and consequently vehicle tours may not be geographically separate. In a practical setting, it is desirable to clearly separate one vehicle’s route from another in an obvious geographic way.

The routes generated using conventional CPP solution methods also fail to take into account the amount of time spent turning at street intersections. In large-scale routing problems, turning costs are an important factor to consider. Carlsson and Ye (2009) described a multi-stage metaheuristic for traversing every street in a city with a fleet of vehicles that addresses these issues.

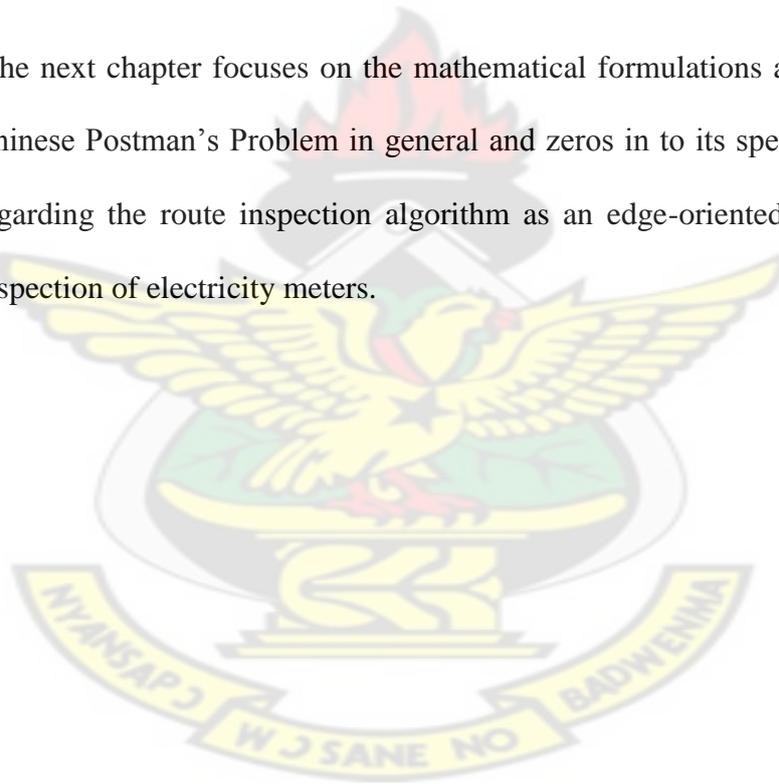
Carlsson and Ye (2009) proposed a model which was particularly distinguished by four (4) features:

- i.)** The presence of multiple vehicles located initially at various depots (an NP-hard addition),
- ii.)** The incorporation of turning costs in the objective function (an NP-hard addition),
- iii.)** The mixed nature of the graph edges (an NP-hard addition), and
- iv.)** The desire to keep vehicle tours as geographically separate as possible, without imposing significant additional costs.

While each of these complications has been addressed individually in earlier literature, it was suggested that this was a novel attempt to combine these hurdles in a single problem.

However the model proposed in this thesis, is based on three (3) out of four (4) of the features proposed by Carlsson and Ye (2009); thus it does not incorporate the turning cost of the each vehicle being used for the inspection, since the municipality is not a vehicular traffic prone area. This takes out some complexities associated to getting an optimal solution.

The next chapter focuses on the mathematical formulations as it relates to the Chinese Postman's Problem in general and zeros in to its specific formulations regarding the route inspection algorithm as an edge-oriented heuristic for the inspection of electricity meters.



CHAPTER 3

3.0 METHODOLOGY

This chapter involves the chosen heuristic algorithm for the Chinese postman problem and the travelling salesman problem models and the specific meter inspection problem formulation.

There are many real-world situations that can be reduced as the Chinese postman problem. For example, a driver of a watering car, a garbage truck or meter inspection personnel, he wishes to choose his route in such a way that traverses as little as possible. In this section, we introduce an efficient algorithm for solving the Chinese postman problem, due to Carlsson and Ye (2009) model.

3.1 Chinese Postman Problem Formulation and Notation

In his job, a postman picks up mail at the post office, delivers it, and then returns to the post office. He must, of course, cover each street in his area at least once. Subject to this condition, he wishes to choose his route in such a way that he walks as little as possible. This problem is known as the Chinese postman problem, since it was first considered by a Chinese mathematician, Guan in 1960.

3.2 The graphic model

We refer to the street system as a weighted graph (G,W) whose vertices represent the intersections of the streets, whose edges represent the streets (one-way or two way), and the weight represents the distance between two intersections, of course, a positive real number. A closed walk that covers each edge at least once in G is called a postman tour. Clearly, the

Chinese postman problem is just that of finding a minimum-weight postman tour. We will refer to such a postman tour as an optimal tour.

If it is assumed first that the graph G is eulerian, then any Euler circuit is an optimal tour since it traverses each edge exactly once. The Chinese postman problem is easily solved in this case, since there exists an efficient algorithm determining an Euler circuit in an eulerian graph, no matter that it is directed or undirected.

Groves and Vuuren (2005) published local search framework for the (undirected) Rural Postman Problem (RPP). The framework allows local search approaches that have been applied successfully to the well-known Travelling Salesman Problem also to be applied to the RPP. Some efficient heuristics for the RPP, based on this framework, are introduced and these are capable of solving significantly larger instances of the RPP than have been reported in the literature. Test results that were presented for a number of benchmark RPP instances in a bid to compare efficiency and solution quality against known methods.

3.2.1 Description of the algorithm

Consider a weighted graph $G = (V, E)$, with vertex set $V = \{v_1, v_2, \dots, v_p\}$ edge set E , and edge weights denoted by $c(i, j), \forall v_i, v_j \in E$. The well-known Chinese Postman Problem (CPP) is the problem of determining a minimum-weight closed route traversing each edge $v_i, v_j \in E$ at least once (Guan, 1962).

The Rural Postman Problem (RPP) is a generalisation of the CPP in which a subset of the edges $E_r \in E$ (called required edges) have to be traversed. It is the problem of determining a minimum-weight closed route traversing each edge in E_r at least once. The RPP is NP-Hard (Lenstra and Rinnooy Kan, 1976), except when $E_r = E$, in which case the problem reduces to the CPP. The above CPP and RPP definitions for undirected graphs have been

generalised in many ways, and algorithms catering for directed and mixed graphs, for example, have been introduced by, Dror (2000)

3.2.2 Local Search Framework

Denote a solution to the RPP by the sequence $S = \langle (v_{s_1}, v_{t_1}), (v_{s_2}, v_{t_2}), \dots, (v_{s_n}, v_{t_n}) \rangle$ of required edges in the order in which they are traversed. Traversals taking place between the required traversals are omitted from the sequence and are assumed to take place along routes corresponding to shortest distances between the required edges of the sequence. The total weight of the route is therefore given by

$$c(s) = \sum_{j=1}^n c(s_j, t_j) + \sum_{j=1}^{n-1} d(t_j, s_{j+1}) + d(t_n, s_1) \text{-----(3.1)}$$

where $d(k, \ell)$ denotes the shortest distance between two vertices $v_k, v_\ell \in V(G)$ and $c(i, j)$ is the cost weight associated with the edge v_i, v_j , as before.

3.2.3 Applying Local Search Moves

In a local search framework moves are performed on candidate solutions to the RPP that directly specify the order in which required edges are traversed in the transformed solution. An example of such a move is one that simply exchanges the order in which two required edges are traversed.

Prototype example:

$$\text{The route, } S^1 = \langle (3,4), (4,1), (5,6), (5,4), (6,8) \rangle$$

the edges $(3,4)$ and $(5,6)$ might be exchanged, to yield the transformed route

$$S^{1*} = \langle (5,6), (4,1), (3,4), (5,4), (6,8) \rangle$$

Typically, one would consider all pairs of these exchanges during a single iteration of the search and then perform one that yields a route of minimum overall cost. In the above example, the traversal order of required edges was altered, but not their traversal directions. However, it may be better to traverse the edge (3, 4) in S^{1*} , for example, in the direction (4, 3) instead of in the direction (3, 4). Consequently, it is necessary to determine the optimal directions of traversals of required edges in S^{1*} after performing an exchange. Applying a move therefore involves altering the order of the required edges in the route, and then determining their directions of traversal. This requirement for determining traversal directions results in an increased time complexity, when compared to applying the same type of move to a VRP. However, by using a complexity reduction method presented later, it is, in fact, possible to determine the cost of a route without predetermining the traversal directions of all of its required edges. This allows for the development of comparatively efficient procedures for many ARPs.

3.3 Mathematical formulations for the route inspection problem.

3.3.1 Definitions:

Edges are distance between two distinct vertices.

A self-loop is an edge that joins a vertex to itself.

A multigraph is a graph where multiple edges and self-loops are allowed.

A simple graph is one which does not allow multiple edges or self-loops.

The degree or order of a vertex is the number of arcs incident to that vertex. A vertex is even (odd) if it has even (odd) order.

If all the vertices are even, then the graph is Eulerian.

If exactly two vertices in are odd, then the graph is semi-Eulerian.

A graph is traversable if it is possible to travel along (traverse) every arc exactly once without taking your pen off the paper.

A graph is traversable if all vertices have an even order (i.e. the graph is Eulerian).

A graph is semi-traversable if it has precisely two vertices that are odd (i.e. it is semi-Eulerian). In this case the start and finish points of the route must be the vertices with the odd order.

A graph is not traversable if it has more than two odd vertices.

A traversable graph is one that can be drawn without taking a pen from the paper and without retracing the same edge. In such a case the graph is said to have an Eulerian trail.

Consider a multi graph G :

A walk in G that traverses every edge in G exactly once is called an Eulerian trail.

On the other hand, if the trail begins and ends at the same vertex, it is called an Eulerian circuit or Eulerian tour. If G has an Eulerian tour, we say G is an Eulerian graph. The CPP (route inspection algorithm) can be modelled as a problem of finding an Eulerian tour in a graph that has minimum cost.

Let v be a vertex in G . The degree of v , $\text{degree}(v)$ is the number of edges that are attached to v . If $\text{degree}(v)$ is odd, we say v is an odd vertex; if $\text{degree}(v)$ is even, we say v is an even vertex.

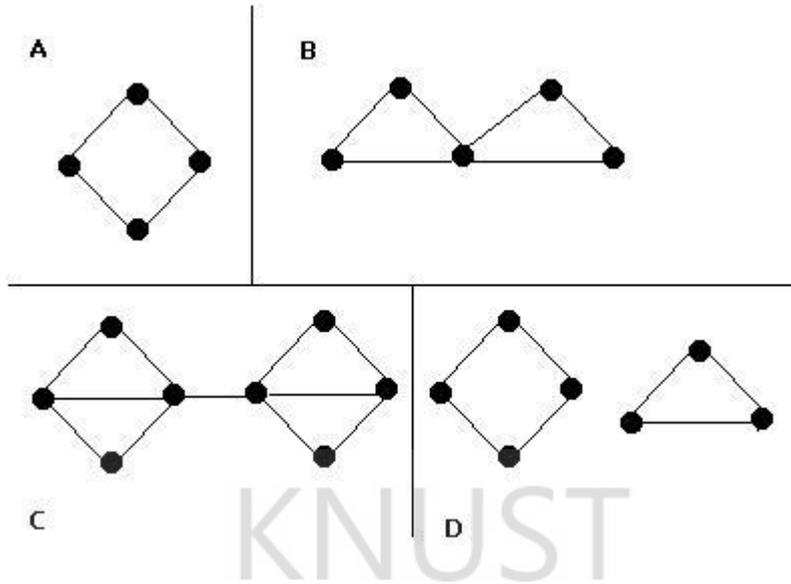


Figure 3.1: A typical representation of a multigraph

Let G be a graph. A nontrivial component of G is a subgraph containing at least two distinct vertices and at least one path between any pair of vertices in the subgraph.

3.3.2 Observations:

- i. The graph **A** and **B** in **figure 3.2** are Eulerian graphs but graph **C** and **D** are not.
- ii. Graph **C** is not Eulerian because it contains odd vertices. **D** is not Eulerian because it is composed of two nontrivial components.
- iii. The Eulerian trail would necessarily begin and end at the odd nodes.
- iv. If G is a graph with only even vertices and one nontrivial component then, the solution to the CPP uses every edge in G , Exactly once and the total cost of the tour is the sum of all of the edge weights.

3.3.3 Theorems:

Let G be a graph:

- i. There is always an even number m , of odd vertices in a graph.
- ii. G is an Eulerian graph if all the vertices are even and all the edges belong to a single component.
- iii. G has an Eulerian trail if all the edges belong to a single component and there are at most two odd vertices.

3.3.4 Route inspection heuristic ‘algorithm’ for optimal tour distance

An algorithm for finding an optimal Chinese postman route is proposed by Groves and Vuuren (2005) as follows:

Step 1: List all odd vertices.

Step 2: List all possible pairings of odd vertices.

Step 3: For each pairing, find the edges that connect the vertices with the minimum Weight (this could also be done by making use of Dijkstra’s algorithm)

Step 4: Find the pairings such that the sum of the weights is minimised.

Step 5: On the original graph, add the edges that have been found in **Step 4**.

Step 6: The length of an optimal Chinese postman route is the sum of all the edges added to the total found in **Step 4**.

Step 7: A route corresponding to this minimum weight can then be found using the Fleury’s algorithm.

This algorithm finds the shortest route that traverses every arc at least once and returns to the starting point. The algorithm consists of three possibilities:

- i.) If all the vertices have an even order then the graph is traversable. The length of the shortest route is therefore equal to the total weight of the network.

- ii.) If there are two odd vertices, then the whole network is traversed once and then the shortest path between the two odd vertices is repeated.
- iii.) If there are more than two odd vertices, then, the algorithm considers the length of the routes between all possible complete pairings of odd vertices and then repeat the pairings which add the smallest distance.

Prototype Example 3.1:

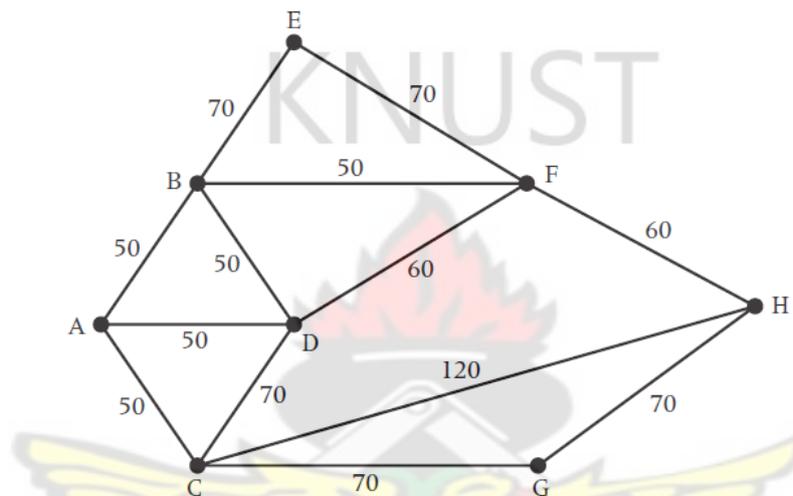


Figure 3.2 : Multi graph with only two odd vertices (Semi Eulerian)

The problem is to find a trail that uses all the edges of the graph in **figure 3.2** with minimum length.

The problem could be solved by using the following algorithm:

Step 1: The odd vertices are **A** and **H**.

Step 2: There is only one way of pairing these odd vertices, namely **AH**.

Step 3: The shortest way of joining A to H is using the path AB, BF, FH, a total length of 160.

Step 4: Draw these edges onto the original network.

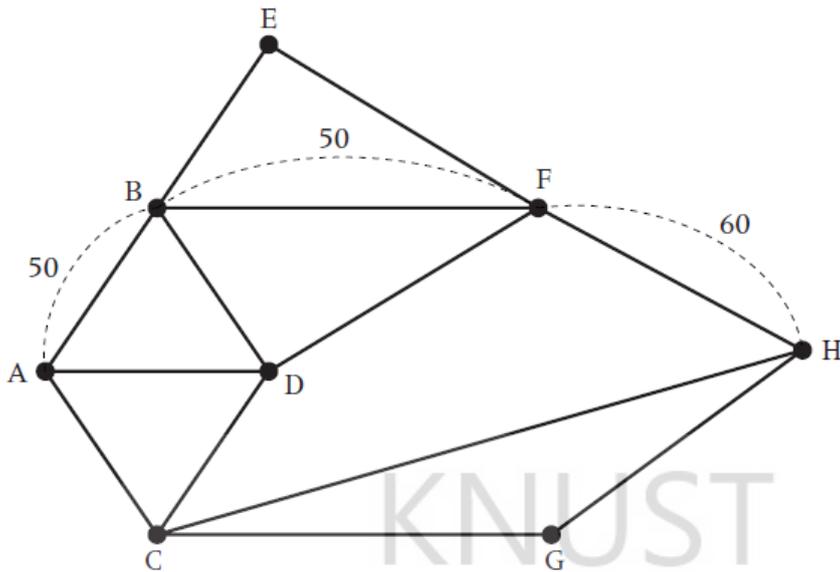


Figure 3.3: Finding the path to be repeated in a multi-graph of two odd vertices

Step 5: The length of the optimal Chinese postman route is the sum of all the edges in the original network, which is 840 m, plus the answer found in **Step 4**, which is 160m.

Hence the length of the optimal Chinese postman route is 1000 m.

Step 6: One possible route corresponding to this length is:

A D C G H C A B D F B E F H F B A, But it must be noted that many other possible routes of the same minimum length can be found.

Prototype Example 3.2: (Three Odd vertices)

A postman has to go around the following route starting and finishing at A, his postal depot. He has to go along each road, shown as lines, once and only once, the task is find the shortest possible route to do this and its associated length.

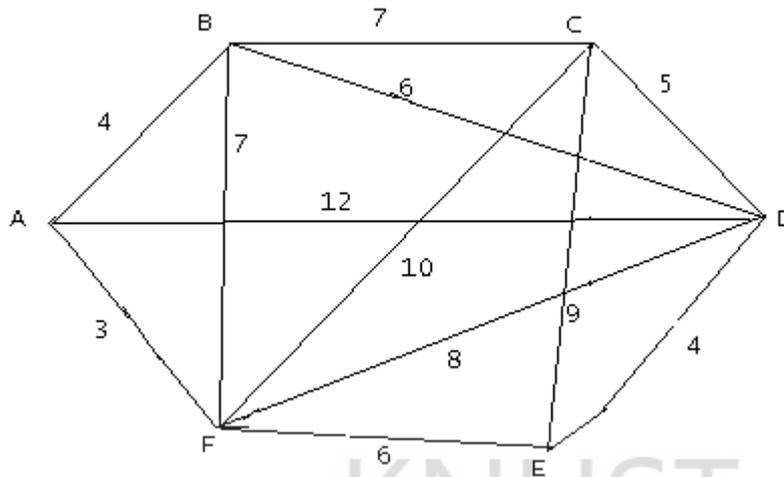


Figure 3.4: Prototype example for multi-graph with odd vertices

Following the steps of the route inspection algorithm:

STEP 1:

Odd Vertices	Degree
A	3
D	5
E	3
F	5

STEP 2 & 3: finding the shortest path to be repeated in the tour using Dijkstra's algorithm

Table 3.1 Using Dijkstra's algorithm to find the repeated edges

Possible pairings of odd vertices	Shortest Route	Shortest Distance
AD and EF	AD and EF	$12 + 6 = 18$
AE and DF	AFE and DF	$3 + 6 + 8 = 17$
AF and DE	AF and DE	$3 + 4 = 7$

STEP 4

From the above calculations it is clear that to create a Eulerian graph that is the shortest possible edges **AF** and **DE** must be repeated.

A possible route starting and finishing at vertex **A** is: **A F E D C B A D E C F D B F A**

The total length of this route is:

$$4 + 7 + 5 + 6 + 7 + 12 + 10 + 8 + 9 + 6 + 4 + 3 + \text{Repeated edges } (4 + 3) = 78 \text{ units}$$

3.4 Proposed mathematical heuristic model for electricity meter inspection.

Electric utility companies employ a crew of workers who periodically visit and read the electric meters of each customer in their service area. Each reader is transported by a vehicle from a central office to the first customer on his work list known as a round. At the end of his work shift time limit the reader is free to leave the area possibly returning home or to the central office by public bus. Taking a graph that corresponds to the city network of streets, meter readers must traverse each street while moving from house to house. It is possible that dead heading may be required—back tracking over a street that has already been covered. A working tour is an open path whose reading time plus deadheading time does not exceed the work limit. The problem is to find the minimum number of working tours. Stating the problem in this manner gives us an optimization problem closely related to the *k*-Chinese postman problem—an edge oriented routing problem.

Carlsson and Ye (2009) proposed a model which was particularly distinguished by four (4) features; and these are:

- i.) The presence of multiple vehicles located initially at various start points (an NP-hard addition)
- ii.) The incorporation of turning costs in the objective function (an NP-hard addition),
- iii.) The mixed nature of the graph edges (an NP-hard addition), and

- iv.) The desire to keep vehicle tours as geographically separate as possible, without imposing significant additional costs.

While each of these complications has been addressed individually in earlier literature, the researcher considers it as a novel attempt to combine these hurdles in a single problem.

However the model proposed in these thesis, mimics on three (3) out of the major features proposed by Carlsson and Ye (2009); thus it does not incorporate the turning cost of the each vehicle being used for the inspection, since the municipality is not a vehicular traffic prone area as compared to certain areas in Ashanti and Greater Accra regions.

3.5 Model assumptions

- i.) It is assumed that the traffic situation does not affect the weight (distance) on each edge of the graph (street).
- ii.) The model considers the weight of each edge of the graph in terms of distance instead of time.
- iii.) No weights (distance) are assigned for left, right or U-turns at each junction.
- iv.) Each street would have two (2) or more inspection personnel; one (1) on the left and the other on the right for those inspecting densely populated areas such as the central business district (CBD).
- v.) It is assumed that the number of prepaid meter users; be it commercial or domestic, are approximately equal on both sides of the street such that the personnel will finish two separate inspection officers on both side of the street are due to finish at the same time when inspection begins.

- vi.) They are no priorities in the application of the algorithm, in developing the formation of the optimal inspection route.
- vii.) The constraint of dividing the entire graph into clusters and developing the optimal route does not affect the optimal distance covered.
- viii.) All vehicles used in the inspection process have the same starting point taken to be the ECG regional office located around the vertices designated '19' in the multi graph of the Central Business District.
- ix.) 'Y' and 'T'- junctions or traffic lights are considered as odd vertices.
'X'- junctions, where there are no traffic control measures, or having traffic light or roundabouts are taken to be even vertices.
- x.) Straight routes with dead end or without any major street intersections are taken to be self-loops.

3.6 Distinguishing feature of the proposed model:

This proposed model is based on the Carlsson and Ye (2009) model. It has been proposed with two (2) particularly distinguishing features:

- i.) The presence of multiple meter inspectors located initially at various chosen locations in the municipality (an NP-hard addition),
- ii.) The desire to keep the meter inspectors tours as geographically separate as possible, without imposing significant additional costs.

The Electricity Company of Ghana uses an adhoc 'cluster first - route second' to perform periodic inspection on its electricity meter users. The municipal multi-graph developed include the Koforidua township and its immediate environs. The municipality is divided into blocks and the blocks are subdivided into rounds and

the individual residents or commercial buildings or shops are identified by their plot numbers.

We are given a map of all of the streets in a municipality representing each round.

This is represented with a mixed graph $G = (V, E \cup A)$, where

E denotes a set of undirected edges (road links)

A a set of directed arcs.

We refer to $E \cup A$ as the set of links in G .

Each vertex i represents a street intersection with given coordinates $x_i \in \mathbf{R}^2$.

The service region, C , is the convex hull of all nodes X_i .

Each street segment, either an edge (i, j) or an arc $h_{i, j, i}$, has length d_{ij} .

When a link's class is unspecified (i.e. when it may be either an edge or an arc), we use the notation $[i, j]$.

For purposes of exposition, we assume that each inspection team has exactly one associated vehicle and at least two inspection officer.

At each vertex (intersection) i , we have sets In_i and Out_i of incoming and outgoing edges and arcs.

Each of these sets includes all edges incident to V_i , since they are undirected and are both incoming and outgoing. If a particular turn from link $[i, j]$ to link $[j, k]$ at vertex j is forbidden then we set $t_{ijk} = \infty$. since the turning cost is not included in the model, this implies $t_{ijk} = 0$

The Electricity Company of Ghana uses an adhoc 'cluster first - route second' to perform periodic inspection on its electricity meter users. The town includes the Koforidua Township and its immediate environs. The district is divided into blocks and the blocks are subdivided into rounds and the individual residents or commercial buildings or shops are identified by their plot numbers.

3.6 General CPP Algorithm.

Step 1: List all the odd vertices in the graph. If there are no odd vertices, go to STEP 5 else go to step 2.

Step 2: Find all possible SETS pairings of the odd vertices.

Step 3: For each SET of pairings, find the shortest path between the two vertices in each pair. Compute the total cost of the SET of pairings by adding up the costs of the shortest paths.

Step 4: Select the SET of pairings with minimum weight and repeat these edges in the graph.

Step 5: Use Fleury's algorithm to find an Euler tour in the resulting graph, starting from any vertex. What has been accomplished in STEPS 2 through 4 is to convert a non-Eulerian graph into an Eulerian graph by adding edges to the graph. This is equivalent to having our postman/inspection officer walk up and down the same street.

3.7 CPP Algorithm for Undirected Graph

STEP 1: Identify all nodes of odd degree in $G(N, A)$ Say there are m of them, where m is an even number.

STEP 2: Find a minimum-length pair wise matching of the m odd-degree nodes and identify the $m/2$ shortest paths between the two nodes composing each of the $m/2$ pairs.

STEP 3: For each of the pairs of odd-degree nodes in the minimum-length pair wise matching found in Step 2, add to the graph $G(N, A)$ the edges of the shortest path

between the two nodes in the pair. The graph $G^1(N, A^1)$ thus obtained contains no nodes of odd degree.

STEP 4: Find an Euler tour on $G^1(N, A^1)$. This Euler tour is an optimal solution to the Chinese postman's problem on the original graph $G(N, A)$. The length of the optimal tour is equal to the total length of the edges in $G(N, A)$ plus the total length of the edges in the minimum-length matching.

3.7.1 Obtaining the optimal tour route using Fleury's Algorithm for a graph with only even vertices and one nontrivial component

If G is a graph with only even vertices and one nontrivial component then we may identify an Eulerian tour as follows:

- (i) Start with any vertex.
- (ii) From the current vertex traverse any unselected edge whose deletion would not result in a graph with two nontrivial components that is a disconnection between the remaining graph.
- (iii) Delete the selected edge from the graph. If there are no edges remaining STOP; otherwise, go back to STEP 2.

3.7.2 Obtaining the optimal tour route using Fleury's Algorithm for Graph with at most two odd vertices and one nontrivial component.

If G is a graph with at most two odd vertices and one nontrivial component then we may identify an Eulerian trail as follows:

- (i) Start with any odd vertex. (If all vertices are even, start anywhere)
- (ii) From the current vertex traverse any unselected edge whose deletion would not result in a graph with two nontrivial components.
- (iii) Delete the selected edge from the graph. If there are no edges remaining STOP; otherwise, go back to STEP 2.

Prototype Example 3.3

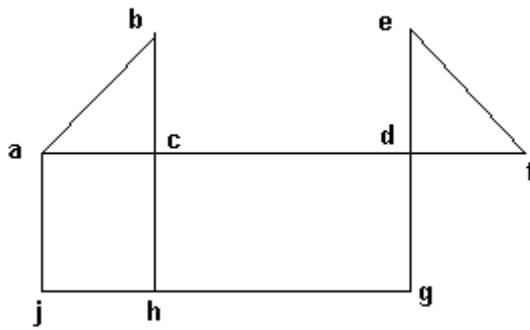


Figure. 3.5a Prototype example for finding the optimal route

In the graph above, degree (a) = 3 and degree (h) = 3. All other vertices are even. Hence, the graph does not have an Eulerian tour but it does have an Eulerian trail beginning at a and ending at h .

Starting at a , we have $a \rightarrow b \rightarrow c \rightarrow a \rightarrow j \rightarrow h \rightarrow c \rightarrow d$. At this point we have the graph shown in figure 3.6b and 3.6c below (with the edges that have been already traversed deleted).

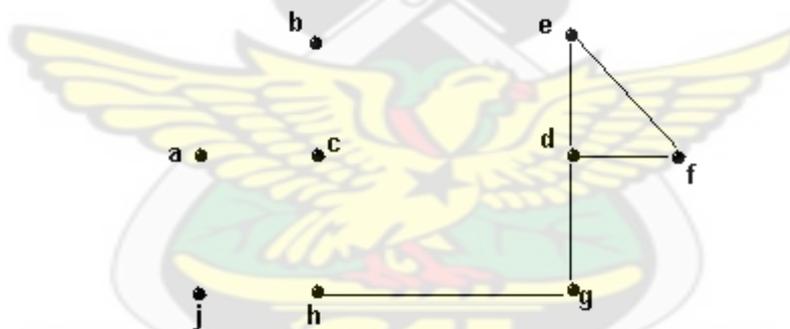


Figure 3.5b

We cannot visit g next because deleting dg would leave a graph with two nontrivial components:

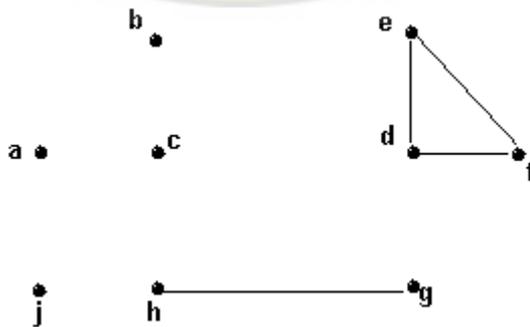


Figure 3.5c

So we visit e next and complete the trail with $e \rightarrow f \rightarrow d \rightarrow g \rightarrow h$.

3.8 Multi route Chinese Postman Problem

Just as in the case of node covering, multi route edge-covering problems are very meaningful and applicable in the urban environment. This is the idea of the cluster first and route second in the attempt to the route inspection problem. This is due to the fact that in the municipal areas are obviously subdivided on a routine basis into smaller constituencies and constituencies are also divided into smaller towns and smaller towns into suburbs that can be covered by a single mailman or meter inspector. This 'districting' aspect is an integral part of the multi-route Chinese postman problem. This problem is usually referred to as the constrained Chinese postman problem (CCPP), since the need to subdivide an area into many routes arises due to some constraint(s), such as the maximum distance that a mailman/inspection personnel can cover walking during a normal day or, very often, other limits on some measures of workload that have been agreed on in a labour contract.

The CCPP has not been investigated extensively to date, but practical approaches to it--in the context of the delivery of urban services--have been suggested for both undirected and directed networks. Because of the relative ease with which the single-tour Chinese postman problem can be solved, the "route first, cluster second" strategy seems to be the favoured one in this case: a giant tour is first found and then divided into m subtours, where m is the number of available vehicles/ team of inspectors.

However, there exist no "best way" available for breaking up the giant tour into shorter subtours, this approach depends to a large extent on the ability and experience of the researcher.

Indeed, the approach described below for an undirected network is most effective when carried out manually with the assistance of a good road network map.

3.8.1 Solving the Constrained Chinese Postman - CCPP problem

The proven approach is to subdivide the graph G' , on which the large, single tour is drawn, in such a way as not to create odd-degree nodes on the boundaries between subtours. Since G' has been derived by applying a CP algorithm to the original graph G , G' has no odd-degree nodes. Therefore, all the nodes in the interior of subtours will be even-degree nodes and the partitioning process can create odd-degree nodes only on the boundaries between subtours. To avoid this, it is important to draw continuous boundaries for each subtour, so that an even number of edges is incident on each node. The following describes informally a possible heuristic approach:

3.8.2 The Constrained Chinese Postman Problem– (CCPP) Algorithm

STEP 1: Using a CPP algorithm, create an Eulerian graph from the given network whose edges are to be covered. Sketch out roughly the boundaries of the m subtours in accordance with the given constraints on tour lengths.

STEP 2: Carefully draw a continuous boundary for each subtour so that an even number of edges is incident to every node.

STEP 3: Sketch out roughly the boundaries of the m subtours in accordance with the given constraints on tour lengths.

Prototype Example 3.4 (CCPP)

Consider again the route inspection problem of the modified graph G' tour this is clearly a network with no nodes of odd degree and with total edge length equal to 383 distance units after applying our CPP algorithm for the solution of the single tour.

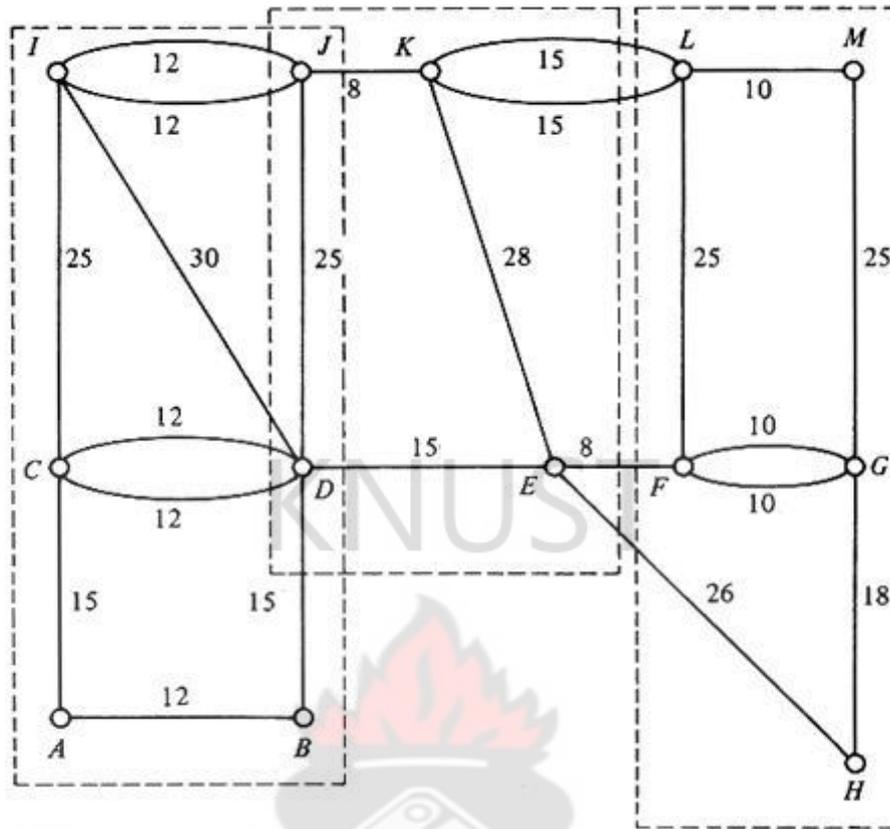


Figure 3.6: Tentative partitioning of the single – CP tour into three approximate equal sub tour.

Suppose now that an upper limit of 150 distance units is placed on the length of a mailman's tour. We then attempt to subdivide the single 383-unit tour into three approximately equal tours, each of which satisfies the 150-unit limit. (Alternatively, it might have been specified that the area/town in question must be covered by three mailmen.)

On **Figure 3.6** the rough outlines of three approximately equal-length tours are sketched in accordance with Step 2 of the CCPP algorithm. These outlines may overlap since they serve only as an aid in defining the approximate physical boundaries of the subtours. In Step 3 the three subtours are designed in detail with continuous boundaries to ensure both the existence of an Eulerian tour and an increase in the total distance covered, which is as small as possible. The three subtours shown in **Figure 3.7** are 121, 130, and 132 units long. Their

total length in this particular case turns out to be exactly equal to the length of the single tour from which they were derived.

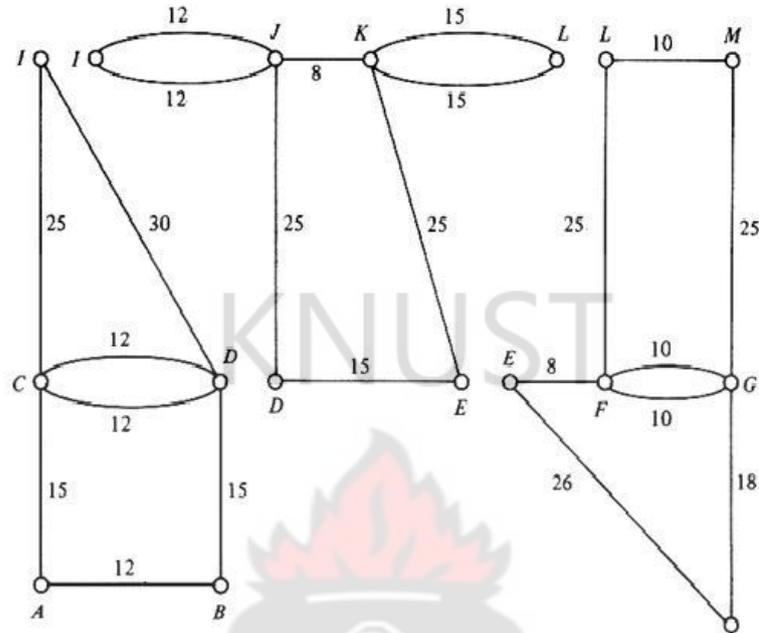


Figure 3.7 : Final partitioning of the single CP tour into three approximate subtour

Two disadvantages of the approach that we just illustrated are readily apparent. First, some trial-and-error work may be required before a set of feasible tours is obtained. This is due to the fact that the subdivision of the tour is initially made by inspection alone. Second, the algorithm above does not take into consideration the distances involved in getting to each district from the central station (post office, depot, etc.) and back. These distances--or, better, the time required in practice to cover them--are considered to be second-order-effect quantities.

The next chapter is focused on the analysis of the road network of the Koforidua municipality obtained from the ECG-regional office with permission. The distances of the vertices are extracted and the results are analysed.

CHAPTER 4

4.0 DATA COLLECTION AND ANALYSIS

4.1 Representation of the road layout in Koforidua township:

For the purpose of this work, numbers have been allocated to all major junctions, roundabouts and traffic lights in the Koforidua. The layout representing the multi-graph of the streets for Koforidua municipality is shown in **figure 4.1**:



Figure 4.1: Lay out of the Koforidua Municipality. (Source: ECG, Koforidua)



**Figure 4.2 Lay out of the road network divided into four subtours
(four inspection BLOCKS)**

For the purpose of the study and also to mimic the practicality of the inspection as a constraint Chinese postman's problem (CCPP) or the k-CPP, the municipality is divided into 4 blocks namely; block 1, block 2, block 3 and block 4, as show in figure 4.2. The interpretation of the blocks are as follows:

Block 1: Covers the Central Business District and Sorodae - suburbs.

Block 2: Covers the Old estate and the Old SSNIT flats and Nyamekrom- suburbs.

Block 3: Covers the Atekyem and Galloway - suburbs.

Block 4: Adweso, Poly and Adweso SSNIT flats - suburbs.

4.2 Formulation of the Constraints Chinese Postman's Problem (CCCP)

Assumptions revisited:

- i.) The analysis considers the weight of each edge of the graph in terms of distance instead of time.
- ii.) No weights (distance) are assigned for left, right or U-turns at each junction.
- iii.) It is assumed that the number of prepaid meter users; be it commercial or domestic, are approximately equal on both sides of the street such that the personnel will finish two separate inspection officers on both side of the street are due to finish at the same time when inspection begins.
- iv.) They are no priorities in the application of the algorithm, when carrying out the inspection along the optimal inspection route.
- v.) All vehicles used in the inspection process have the same starting point taken to be the ECG regional office located on the hospital road 0.25km from vertices 29 (in Block 2) near the of the Regional Hospital

- vi.) Since the inspection teams start at the same point, the distance covered for the movement of each vehicle to the specific starting point of the inspection exercise are not considered.
- vii.) ‘Y’ and ‘T’-junctions or traffic lights are considered as odd vertices. ‘X’-junctions, where there are no traffic control measures, or having traffic light or roundabouts are taken to be even vertices.
- viii.) Straight routes with dead end or without any major street intersections are taken to be self-loops. All dead ends created as a result of the segregation of the multi graph are treated as self-loops and the weight of the graph is doubled when finding the optimal distance covered.

This required routes to be services are represented with a mixed graph $G = (V, E \cup A)$, where E denotes a set of undirected edges (road links), and A is set of undirected arcs. We refer to $E \cup A$ as the set of links in G .

Each vertex i represents a street intersection which could be assigned coordinates $x_i \in \mathbf{R}^2$. Each street segment, either an edge (i, j) or an arc $h_{i, j, i}$, has length E_{ij} .

The problem can be defined as follows: Let $G = (V, E)$ be a complete undirected graph with vertices $V, = |V| = n$

For ‘n’ number of vertices and with all edges represented as E_{ij} (length of the edge between vertex i and vertex j), all repeated edges as E_{rk} and all roads with dead ends; E_{lk} , where $E_{rk} \in E_{ij}$

Each edge is undirected and symmetric, for all vertices (i, j) . $E_{ij} = E_{ji}$

The Optimal route for the Chinese Postman’s problem for an undirected multi graph could be generally formulated as a minimization problem given as:

$$P_1: \min \left(\sum_{\substack{i,j=1 \\ i \neq j}}^n E_{ij} + \sum_{\substack{r,k=1 \\ i \neq k}}^n E_{rk} + 2 \sum_{\substack{l,k=1 \\ l \neq k}}^n E_{lk} \right) \dots\dots\dots(4.1)$$

Subject to:

$$\sum_{i,j=1}^n E_{ij} - \sum_{r,k=1}^n E_{rk} > 0 \dots\dots\dots(4.2)$$

$$\forall E, E_{ij} > 0$$

The problem is a minimization problem with additional restrictions that guarantee the exclusion of subtours in the optimal solution. A subtour in V is a cycle that does not include all vertices (road links/intersections). Equation (4.1) is the general objective function, which minimizes the total distance to be travelled by the inspection teams/personnel.

Special case 1: where the graph is strictly Eulerian; with neither ‘Y’ nor ‘T’- junctions, there are no repeated edges (deadheading) nor road links with dead ends; the problem reduces to;

$$P_1: \text{Min} \left(\sum_{\substack{i,j=1 \\ i \neq j}}^n E_{ij} \right) \dots\dots\dots(4.3)$$

where

$$E_{rk} = 0, \text{ and } E_{lk} = 0$$

However, it is rarely the case that every vertex in a road network is even. (Examining a road map of any town or section of a city will confirm this.) According to Euler’s Theorem, when there are odd vertices, it is impossible to plan a circuit that traces every edge exactly once. Since every road needs to be traced, some roads must be retraced.

This poses real – world cases (as in case 2 and 3) to plan a route so that the total amount of retracing is as small as possible.

Special case 2: where the graph has two or more odd vertices (semi-Eulerian) with no road intersections which have dead ends, the objective function reduces to:

$$P_1 : \text{Min} \left(\sum_{\substack{i,j=1 \\ i \neq j}}^n E_{ij} + \sum_{\substack{r,k=1 \\ i \neq k}}^n E_{rk} \right) \dots\dots\dots(4.4) \quad ,$$

where $E_{lk} = 0$

Special case 3: where there are more than two odd vertices (Y and T- junctions or traffic lights) in addition to dead end road intersections, the problem assumes the general minimisation problem:

$$P_1 : \text{Min} \left(\sum_{\substack{i,j=1 \\ i \neq j}}^n E_{ij} + \sum_{\substack{r,k=1 \\ i \neq k}}^n E_{rk} + 2 \sum_{\substack{l,k=1 \\ l \neq k}}^n E_{lk} \right) \dots\dots\dots(4.1)$$

Subject to:

$$\sum_{i,j=1}^n E_{ij} - \sum_{r,k=1}^n E_{rk} > 0 \dots\dots\dots(4.2)$$

$$\forall E, E_{ij} > 0$$

Since All dead end roads are traversed and returned through the same road inspected and they are considered as self-loops.

4.3 Analysis of the Multi Graph and Results

4.3.1 Analysis of the Multi graph and Results for BLOCK 1 (CBD)

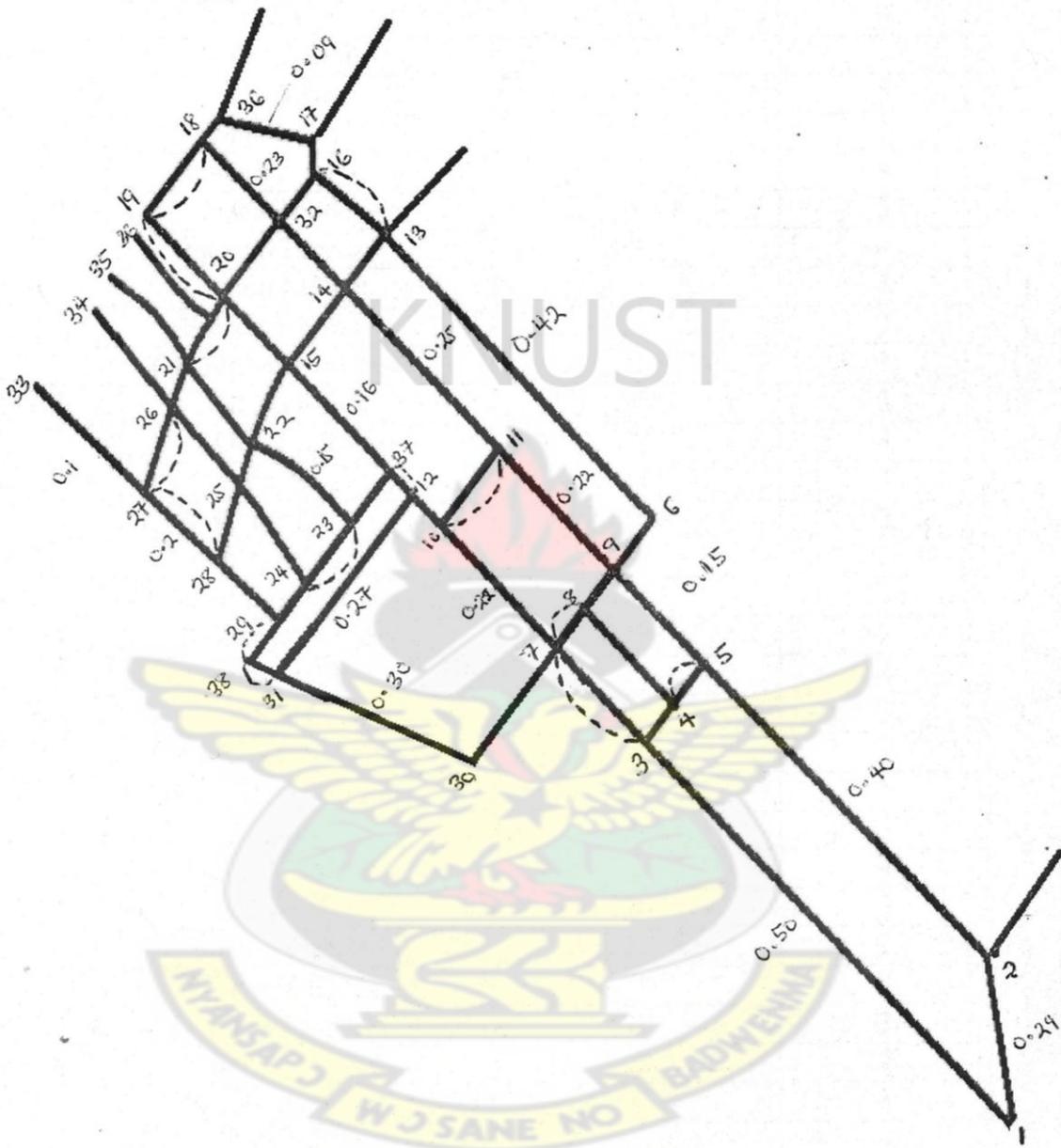


Figure 4.3 line and node diagram for the central business district (CBD) – Koforidua (BLOCK 1)

Table 4.31 Representation of Road links and their designation-**Block 1** (CBD)

NAME OF ROAD JUNCTION	VERTEX DESIGNATION	ORDER
Ayah	1	2
Debrahkrom	2	2
ICGC	3	3
Trotro station	4	3
Sorodoae	5	5
Advance Ghana	6	2
Laundry traffic	7	4
Nana Topen	8	3
Eastern Empire	9	4
Central Market	10	3
Freedom Stores	11	3
Old regional Library	12	3
All Nations	13	3
Ofose line	14	4
GCB	15	4
SIC traffic -1	16	3
SIC traffic -2	17	2
Central Apostolic	18	3
Railways traffic	19	2
Apex Bank	20	4
SSNIT	21	3
Sports council	22	4
IRS – Rent control	23	3
Betom	24	3
Jackson Park_1	25	4
PWD	26	3
Oguaa secretariat	27	2
Workers College	28	3
Gratis foundation	29	3
Betom-Riis	30	2
Jackson Park _2	31	3
Polyclinic	32	4
Coca Cola Main Depot	33	1
Agartha	34	1
UT-Bank	35	1
Regional Hospital	36	2
Antartic traffic	37	3
Anglican	38	2

Odd Vertices are in Block 1: 3, 4, 5, 8,10,11,12,13, 16, 18,21,23,24,26,28,29,31, 37

Table 4.32: Optimal repeated paths for **BLOCK 1**

Optimal Pairing of Odd Vertices	Shortest Route to be repeated	Shortest Distance
[3-8]	[3-7-8]	0.15+0.10
[4-5]	[4-5]	0.10
[10-11]	[10-11]	0.20
[12-37]	[12-37]	0.10
[13-16]	[13-16]	0.21
[18-21]	[18-19-20-21]	0.21+0.134+0.10
[26-28]	[26-27-28]	0.14+0.20
[23-24]	[23-24]	0.13
[29-31]	[29-38-31]	0.05+0.10
Total optimal repeated path		1.924km

Optimal inspection route (Eulerian circuit) for inspection for **BLOCK 1** (the central business district) is as follows:

18 ↗36 ↗17 ↗16 ↗32 ↗14 ↗13 ↗16 ↗13 ↗6 ↗9 ↗5 ↗1 ↗3 ↗7 ↗3 ↗4 ↗5 ↗
 ↗4 ↗8 ↗7 ↗8 ↗9 ↗11 ↗10 ↗11 ↗14 ↗15 ↗37 ↗12 ↗10 ↗7 ↗30 ↗31 ↗38 ↗
 ↗31 ↗12 ↗37 ↗23 ↗24 ↗29 ↗38 ↗29 ↗28 ↗27 ↗28 ↗25 ↗24 ↗23 ↗22 ↗
 ↗25 ↗26 ↗27 ↗26 ↗21 ↗22 ↗15 ↗20 ↗21 ↗20 ↗19 ↗20 ↗32 ↗18 ↗19 ↗18

Total Optimal distance covered: **12.931km**

4.3.2 Analysis of the Multi graph and Results for BLOCK 2

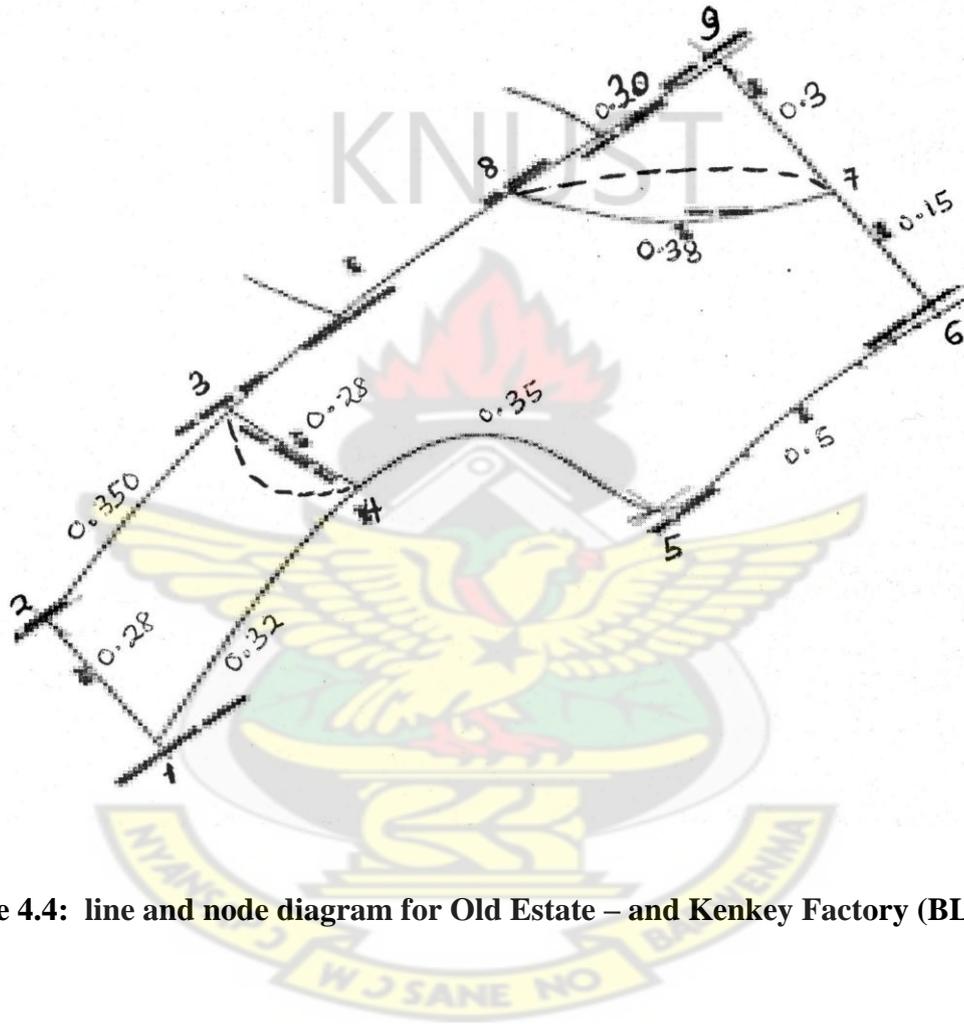


Figure 4.4: line and node diagram for Old Estate – and Kenkey Factory (BLOCK 2)

Table 4.33: Representation of Road links and their designation (Old Estate Suburb)

Name of road vertex	Vertices designation on line and node diagram	Order
Kenkey Factory	1	2
Old estate last stop	2	2
Pentecost Basic School junction	3	3
St. Batikha	4	3
Ministries Junction	5	3
Regional Hospital traffic light	6	2
Spread Eagle junction	7	3
SDA Hospital Junction	8	3
Old Estate Junction	9	2

Odd Vertices are 3, 4, 8 and 7 all with the order of 3

Table 4.34: Optimal repeated paths for **BLOCK 2**

Optimal Pairing of Odd Vertices	Shortest Route to be repeated	Shortest Distance
[3-4] and [8-7]	[3-4] and [8-7]	0.280+0.350 = 0.730km

Edges to be repeated are [3-4] and [8-7] (St. Batikha –Pentecost Basic School junction, and Spread Eagle-SDA junction)

Repeated Distance to be covered for **Block 2**: [3-4] and [8-7] = 0.73km

Total length covered as self loops =

$$= 2(0.390 + 0.350 + 0.180 + 0.150) = 2.14km$$

Total length of all edges =

$$\sum_{\substack{i,j=1 \\ i \neq j}}^n E_{ij} = 0.258 + 0.312 + 0.90 + 0.20 + 0.35 + 0.28 + 0.30 + 0.40 + 0.10 + 0.20 = 3.30km$$

Optimal distance covered in **Block 2** inspection = 3.30+0.730+2.140 = 6.17km

Applying the Fleury's algorithm, an optimal Eulerian walk through Block 2 is found as follows: **6-7-9-8-7-8-3-4-3-2-1-4-5-6**

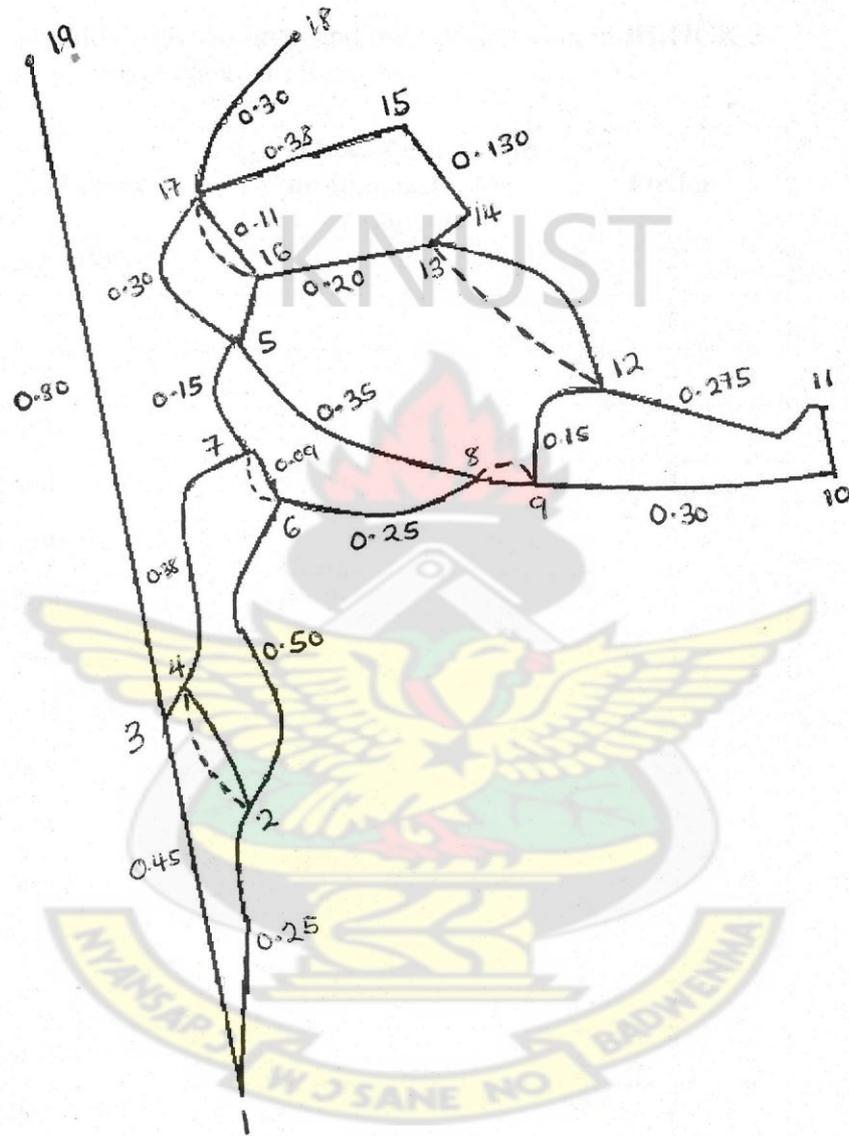


Figure 4.5: line and node diagram for Atekyem - Galloway Road Layout (BLOCK 3)

Table 4.35 Representation of Road links and their designation in **BLOCK 3**
(Galloway- Atekyem Suburbs)

Name of road vertex	Vertices designation on line and node Diagram	Order
New Juabeng - Poly	1	2
New Juabeng	2	3
New Capital view traffic	3	2
Post lodge	4	3
Eredec roundabout	5	4
S. C. Appenteng	6	3
Attekyem	7	3
Sectech	8	3
Galloway - Betom	9	3
Galloway	10	2
Freeman	11	2
Riis Presby	12	3
Ascension Presby	13	3
Laundry traffic	14	2
Municipal traffic	15	2
Partners may hotel	16	3
Residency	17	3
Latter day saint	18	1
Kenkey Factory	19	1

Odd vertices in Block 3: 2, 4, 8, 9, 12, 13, 16 and 17

Table 3.36 Repeated route in the **Block 3** (Atekyem - Galloway)

Optimal Pairing of Odd Vertices	Shortest Route to be repeated	Shortest Distance
[2-4]	[2-4]	0.300
[8-9]	[8-9]	0.125
[16-17]	[16-17]	0.110
[12-13]	[12-13]	0.350
Total optimal path repeated		0.885km

Table 4.37 Self loops in **Block 3**

Assumed Self Loops	Road Designation	Length Of Edge
[3-19]	New Capital View - Kenkey Factory	2x(0.80)
[17-18]	Residency – latter day saint	2x(0.30)
Total estimated distance covered as self loops		2.20km

Optimal Eulerian for the inspection of electricity customer meters in **BLOCK 3:**

4 → 3 → 1 → 2 → 4 → 7 → 5 → 16 → 13 → 14 → 15 → 17 → 16 → 17 →
 → 5 → 8 → 9 → 12 → 13 → 11 → 10 → 9 → 8 → 6 → 7 → 6 → 2 → 4

Optimal distance (without considering self loops): 5.68km+ 0.885km = 6.655km

Optimal distance (considering self loops): 5.68km+ 0.885km +1.1km = **9.955km**

Table 4.38: Representation of Road links and their designation in **BLOCK 4**
(Adweso Poly-SSNIT flats)

Name of road vertex	Vertices designation on line and node diagram	Order
Koforidua Polytechnic traffic	1	2
Acapulco_1	2	3
SSNIT Flat – Poly_2	3	2
Acapulco _2	4	4
Allan –Joy _1	5	4
Adweso 205_1	6	3
Achika - Estate junction	7	3
Starland Hotel	8	4
Allan-Joy _2	9	4
Juabeng Serwaa	10	3
Last Stop	11	3
Assemblies of God	12	4
Wesley International _2	13	3
Adweso 205_2	14	3
Wesley international _1	15	3
New Juabeng	16	3
Mahogany _1	17	2

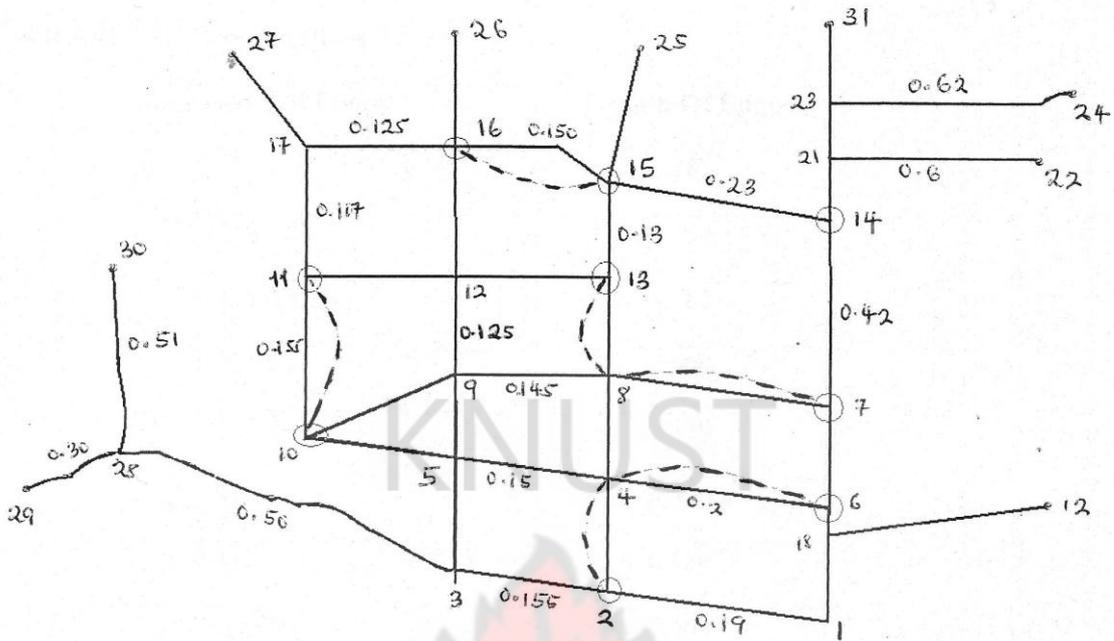


Figure 4.6 line and node diagram for Adweso, Koforidua Polytechnic and SSNIT Flat (BLOCK 4)

Odd vertices in BLOCK 4: 2, 6, 7, 13, 10, 11, 15 and 16

Table 4.39 Repeated routes in BLOCK 4

Optimal Pairing of Odd Vertices	Shortest Route to be repeated	Shortest Distance
[2-6]	[2-4-6]	0.33+0.20
[7-13]	[7-8-13]	0.210+0.125
[10-11]	[10-11]	0.155
[15-16]	[15-16]	0.150
Total optimal path repeated		1.17km

Table 4.40 Self loops in Block 3

Assumed Self Loops	Length Of Edge
[17-27]	0.285
[16-26]	0.154
[15-25]	0.185
[21-22]	0.600
[23-24]	0.720
[23-31]	0.400
[18-21]	0.650
[3-28]	0.400
[28-30]	0.210
[28-29]	0.200
Total estimated distance covered as self loops	3.154km

Applying the Fleury's algorithm, an optimal Eulerian walk is found as follows:

14 ↗ 7 ↗ 8 ↗ 4 ↗ 6 ↗ 1 ↗ 2 ↗ 4 ↗ 6 ↗ 7 ↗ 8 ↗ 13 ↗ 8 ↗ 9 ↗ 5 ↗ 3 ↗ 2 ↗ 4 ↗ 5
 ↗ 10 ↗ 9 ↗ 12 ↗ 11 ↗ 10 ↗ 11 ↗ 17 ↗ 16 ↗ 12 ↗ 16 ↗ 12 ↗ 13 ↗ 15 ↗ 16 ↗ 15 ↗ 14

Optimal distance (without considering self loops): 1.17km +4.691km = 5.861km

Optimal distance (considering self loops): 1.17km +4.691km + 3.154km = **9.015km**

CHAPTER 5

DISCUSSIONS, CONCLUSION AND RECOMMENDATIONS

5.0 INTRODUCTION

This chapter comprises discussion of the results through analysis, drawing conclusions and recommendations based on the outcome of the application of the CPP heuristic.

5.1 DISCUSSION

The focus of this work is centred on constructing optimal routes for the inspection of prepaid meters using the constrained Chinese postman problem heuristic. In the work, the routes are constructed by using the Dijkstra's algorithm with Matlab programme to find optimal tour length between all pairs of odd vertices (mostly 'T' and 'Y' junctions). The Fleury's algorithm is subsequently employed to find the optimal route for the inspection of the meters in both residential and commercial areas located on designated plot numbers along the roads.

In order to show the practicality and mimic what occurs in actual practice, the graph (the road network is divided into four (4) clusters into showing the different suburbs without creating a nontrivial component). The inspection algorithm is applied to find the optimal route taking into consideration all roads with dead-ends.

For optimality, the inspection team are put into four (4) different groups in accordance to the bocks used into which the ECG has put it and the tour of each should also be kept geographically apart as much as possible. However, it is important to note that the researcher performed several experiments on more than those presented, but for the sake of the objectives of this work only few different cases for each inspection team is presented.

5.2 CONCLUSION

The Chinese Postman's algorithm is a powerful heuristic algorithm that is edge centred unlike the TSP which is node centred.

Based on the results and the discussions above, the following conclusions can be made.

- (i) Number of repeated routes in the optimal route length increases directly with increase in the number of nodes
- (ii) Optimal route covered for **Block 1** is **6.17km**, **Block 2** is **9.955km**, **Block 3** **9.015km** is and **Block 4** is **12.931km**.
- (iii) For **Block 1** covering Central business District the inspection order is as follows:

ECG Regional → Regional Hospital → SIC Traffic_2 → SIC Traffic_1 →
→ Polyclinic → → Ofose line → All Nations → SIC Traffic_1 → All Nations →
Advance Ghana → Eastern Empire → Sorodae → debrah krom → Ayah →
ICGC → laundry traffic → ICGC → trotro station → Sorodae → Trotro station
→ Nana Topen → Laundry traffic → Nana Topen → Eastern
Empire → Freedom Stores → Central Market → Freedom stores → Ofose line
→ GCB → Antarctic traffic → Old Regional Library → laundry traffic → Betom
Riis → Jackson Park_2 → Anglican → Jackson park → Old regional
library → Antarctic traffic →
→ IRS - Rent Control → Betom → Gratis foundation → workers
college → Anglican → gratis foundation → workers college → Oguaa Secretariat
→ workers college → Jackson park 1 → betom → IRS Rent-Control → Sports
council → Jackson park_1 → PWD → SSNIT → Sports Council → Apex
Bank → SSNIT → Apex Bank → Railways Traffic → Apex Bank → Polyclinic
→ Central Apostolic → Railways traffic → ECG Regional

(iv) For **Block 2** covering the Old estate, Nsukwao and Kenkey Factory and the inspection order is as follows:

**Regional Hospital traffic → spread eagle → Old Estate → SDA Hospital →
→ Spread Eagle → SDA Hospital → Pentecost Basic School → Old Estate Last
stop → St Batikha Ministries → Regional Hospital traffic**

(v) For **Block 3** covering the Atekyem and Galloway suburbs, the inspection order is as follows:

**Post lodge → New Capital view traffic → New Juabeng – Koforidua Poly → New
Juabeng → → Post lodge → Atekyem → Ereddec roundabout → Partners May
Hotel → Ascension Presby → Laundry traffic → Municipal traffic → Residency →
Partners May Hotel → Residency → → Ereddec roundabout → Sectech → Galloway -
Betom → Riis Presby → Ascension Presby → → Freeman → Galloway → Galloway -
Betom → Sectech → S. C. Appenteng → Atekyem → S. C. Appenteng → New
Juabeng → Post lodge**

(vi) For **Block 4** covering Adweso, Koforidua Poly and SSNIT flats the inspection order is as follows:

**Adweso 205-Main → Achika-Estate junction → Starland Hotel → Adweso 205-1 →
→ Koforidua Polytechnic traffic → Acapulco_1 → Acapulco_2 → Adweso 205_1 →
→ Starland Hotel → Wesley International_2 → Starland Hotel → Allan-Joy_2 →
→ Allan -Joy_1 → SSNIT Flat – Poly_2 → Acapulco_1 → Acapulco_2 →
→ Allan -Joy_1 → Juabeng Serwaa → Allan-Joy_2 → Assemblies of God →
→ Last Stop → Mahogany_1 → New Juabeng → Assemblies of God →
→ Wesley International_2 → Wesley international_1 → New Juabeng →
→ Wesley international_1 → Adweso 205_2**

As we compare the results of using CPP the existing system in ECG, we may argue that the inspection systems has become more scientific and follow a more organised path that could easily be used.

Hence, to run an efficient inspection of meters, Electricity Company of Ghana (ECG), Northern Electrification Development (NED) as well as all organization whose work require that they service every street in the municipality could adopt system that is ‘cluster-first, route-second’ algorithm in CPP or the constrained CPP heuristic for inspection of the meters for optimal tour as part of their preventive maintenance programme.

5.3 RECOMMENDATIONS

- i.)** The Electricity Company of Ghana – ECG could consider the demarcation and the routes below in other to minimize the time/increase the frequency when undergoing periodic meter inspections exercise in the Koforidua Municipality.
- ii.)** Students can use this work for further research covering all subtours within the densely populated suburbs within the municipality such as the Central business District .(CBD)

REFERENCES

1. Amponsah, S. K. (2003). The investigation of a class of capacitated arc routing problems: the collection of garbage in developing countries, Ph.D Thesis, School of Mathematics and Statistics, University of Birmingham, UK.
2. Amponsah, S .K and F.K Darkwah (2007) .Lecture notes on operation Research IDL KNUST 62-67
3. Applegate D, Bixby R.E., Chvatal V., and Cook W. (1998), Finding cuts in the TSP, a preliminary report distributed at The Mathematical Programming Symposium, Ann Arbor, Michigan.
4. A. Tucker and L. Bodin, A Model for Municipal Street Sweeping Operations, Chapter 6 in Discrete and System Models, (Volume 3 of Models in Applied Mathematics), Springer-Verlag, New York, 1983, pp. 76–111.
5. Brummit B, Stentz A. Dynamic mission planning for multiple mobile robots. Proceedings of the IEEE international conference on robotics and automation, April 1996.
6. Brummit B, Stentz A. Grammps: a generalized mission planner for multiple mobile robots. Proceedings of the IEEE international conference on robotics and automation, May 1998.
7. B. L. Golden and R.T. Wong. Capacitated arc routing problems. Networks, 11:305.315, 1981.
8. buddhikagune.blogspot.com/2008/10/dijkstras-algorithm-for-matlab.html
9. Croes G A (1958). A Method for Solving Travelling Salesman Problems. Operations Research, 6(6), 791{812}
10. Dorigo, M. and Di Caro, G. (1999). —Ant colony Algorithm for Discrete Optimization, Artificial Life, 5:137-172.
11. Dino A. , Gerhard R.: A tabu search algorithm for the min–max k-Chinese postman problem, 2005
12. Fogel D. B. A parallel processing approach to a multiple travelling salesman problem using evolutionary programming. In: Proceedings of the fourth annual symposium on parallel processing. Fullerton, CA, 1990. p. 318–26.

13. Fraser, Alex S. (1957), Simulation of Genetic Systems by Automatic Digital Computers, Introduction. Australian Journal of Biological Sciences **10**: 484–491.
14. Goldstein M. Self-organizing feature maps for the multiple travelling salesmen problem. In: Proceedings of the IEEE international conference on neural network. San Diego, CA,1990. p. 258–61
15. Gutin & Punnen (2002) The Travelling Salesman Problem and its Variations. Applegate, Bixby, Chvatal & Cook (2007), The Travelling Salesman Problem: A Computational Study.
16. Greistorter, P (1994). Algorithms and implementations for the mixed capacitated Chinese Postman Problem, Working Paper 33, Department of Business, University of Graz, Austria.
17. G. W. Groves and J H van Vuuren, Efficient heuristics for the Rural Postman Problem, pages 1-12, 2005,
18. Harold T. : The directed chinese postman problem,2000, pages 1-5
19. Hertz A, Laporte G, Nanchen Hugo P., Improvement procedures for the undirected rural postman problem, informs Journal on Computing 1999;11(1):53–62
20. Hsu C, Tsai M, Chen W, A study of feature-mapped approach to the multiple travelling salesmen problem. IEEE International Symposium on Circuits and Systems 1991;3:1589–92.
21. Hsiao-Fan Wang and Yu-Pin Wen, Time-Constrained Chinese Postman Problems, 2001
22. hsor.org/what_is_or.cfm?name=network_routing, 23/11/11, 9:00pm
23. Holland, John H (1975), Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, pp 16-30
24. J. G. Carlsson, D. Ge, A. Subramaniam, and Y. Ye. Solving the min-max multi-depot vehicle routing problem. In Proceedings of the Fields Workshop on Global Optimization, 2007.
25. J. G. Carlsson, D. Ge, A. Subramaniam, and Y. Ye. Solving the multi-depot vehicle routing problem. In Proceedings of the Fields Workshop on Global Optimization, 2008.

26. J. L. Ryan, T.G. Bailey, J.T. Moore, W.B. Carlton, Reactive Tabu search in unmanned aerial reconnaissance simulations. Proceedings of the 1998 winter simulation conference, vol. 1,1998. p . 873–9.
27. J. G. Carlsson and Y. Ye, Practical distributed vehicle routing for street-level map scanning, 2009
28. J. Dr'eo, A. P'etrowski, P. Siarry E. Taillard, Metaheuristics for Hard Optimization, Methods and Case Studies With Simulated Annealing, Tabu Search, Evolutionary and Genetic Algorithms, Ant Colonies vol. 1,1998. p . 673–9.
29. J. Edmonds, The Chinese Postman Problem, Operations Research, Vol.13, Supplement 1, 1965, B73.
30. Lawler Lenstra, Rinnooy-Kan & Schmoys (1980) The Travelling Salesman Problem: a Guided Tour of Combinatorial Optimisation.
31. Lenstra A. T. and Rinnooy K. E, heuristic algorithm for the Rural postman problem. Operations Research 1976;
32. Lin S, Kernighan B. An effective heuristic algorithm for the travelling salesman problem. Operations Research 1973;21:498–516.
33. lancet.mit.edu/~mbwall/presentations/IntroToGAs, 12/10/2011 2:00 pm
34. M. Dror. Arc Routing: Theory, Solutions and Applications. Kluwer Academic Publishers, 2000. G.N. Frederickson. Approximation algorithm for some postman problems. ACM, 1979.
35. Microsoft Encarta 2009
36. people.bath.ac.uk/db230/CP.htm, 19/12/11, 7:00am
37. people.bath.ac.uk/ma3mb/example.htm, 9/12/11, 7:00pm
38. Potvin J, Lapalme G, Rousseau J. A generalized k-opt exchange procedure for the MTSP. INFOR 1989;27(4)
39. Punnen A (2002). The Travelling Salesman Problem: Applications, Formulations and Variations. In Gutin and Punnen (2002), chapter 1, pp. 1{28}
40. Russell R. A. An effective heuristic for the m-tour travelling salesman problem with some side conditions. Operations Research 1997: 517–524.
41. R. K. Ahuja, K. Mehlhorn, J. B. Orlin & R. E.Tarjan: Faster Algorithms for the Shortest Path Problem, Journal of the ACM, 1990, **37**(2): 213–223,

42. Somhom S, Modares A, Enkawa T. Competition-based neural network for the multiple traveling salesmen problem with min-max objective. *Computers and Operations Research* 1999; 26(4):395–407.
43. Tucker and L. Bodin, A Model for Municipal Street Sweeping Operations, Chapter 6 in *Discrete and System Models*, ed. W. Lucas, F. Roberts, and R. Thrall (Volume 3 of *Models in Applied Mathematics*), Springer-Verlag, New York, 1983, pp. 76–111.
44. Vittorio Maniezzo, Algorithms for large directed CARP instances: urban solid waste collection operational support, 2004 pg 2-6
45. Vakhutinsky I. A., Golden LB. Solving vehicle routing problems using elastic net. *Proceedings of the IEEE international conference on neural network*, 1994. p. 4535–40.
46. [Wikipedia.org/wiki/Greedy_ algorithm](http://Wikipedia.org/wiki/Greedy_algorithm), 22/10/11, 1:00pm
47. [Wikipedia.org/wiki/heuristic_ algorithm](http://Wikipedia.org/wiki/heuristic_algorithm), 15/10/11, 10:00am
48. [Wikipedia.org/wiki/Tabu_ search](http://Wikipedia.org/wiki/Tabu_search), 05/10/11, 2:00pm
49. [Wikipedia.org/wiki/Moore's_ law](http://Wikipedia.org/wiki/Moore's_law) 05/10/11, 2:00pm
50. Watcholder E, Han J, Mann RC. A neural network algorithm for the multiple travelling salesmen problem. *Biology in Cybernetics* 1989; 61
51. www-math.cudenver.edu/wcherowi/courses/m4408/gtln17.html, 9/12/11, 5:00pm
52. web.mit.edu/urban_or_book/www/book/chapter6/6.4.14.htm, 12/12/11, 6:00pm
53. Yu Z, Jinhai L, Guochang G, Rubo Z, Haiyan Y, An implementation of evolutionary computation for path planning of cooperative mobile robots. *Proceedings of the fourth world, congress on intelligent control and automation*, vol. 3, 2002. p. 1798–802.
54. Y. Lin & Y. Zhao, A New Algorithm for the Directed Chinese Postman Problem, 1988.

APPENDICES

APPENDIX A

Matlab code for Dijkstra's Algorithm

% An implementation of the Dijkstra's algorithm for MatLab

```
function ans = Dijkstra(src,ConMat)
```

```
N = size(ConMat,1);
```

```
Visited = -1*ones(N,1); % Here -1 means Not defined
```

```
PrevNode = -1*ones(N,1);
```

```
Distance = Inf*ones(N,1);
```

```
Distance(src) = 0;
```

```
CurrentNode = src;
```

```
nVisited = 0;
```

```
while (nVisited <N)
```

```
    Visited(CurrentNode) = 1;
```

```
    for i=1:N
```

```
        if (ConMat(CurrentNode,i)>0)
```

```
            temp = ConMat(CurrentNode,i) + Distance(CurrentNode);
```

```
            if (temp< Distance(i))
```

```
                Distance(i) = temp;
```

```
                PrevNode(i) = CurrentNode;
```

```
            end
```

```
        end
```

```
    end
```

```
    minimum = 0;
```

```
    for i= 1:N
```

```
        if (Visited(i)<0)&&(Distance(i)>0)
```

```
            if (minimum ==0)
```

```
                minimum = i;
```

```
            elseif (Distance(i)<Distance(minimum))
```

```
                minimum = i;
```

```
            end
```

```
        end
```

```
    end
```

```
    CurrentNode = minimum;
```

```
    nVisited = nVisited +1;
```

```
end
```

```
ans = PrevNode;
```

APPENDIX B

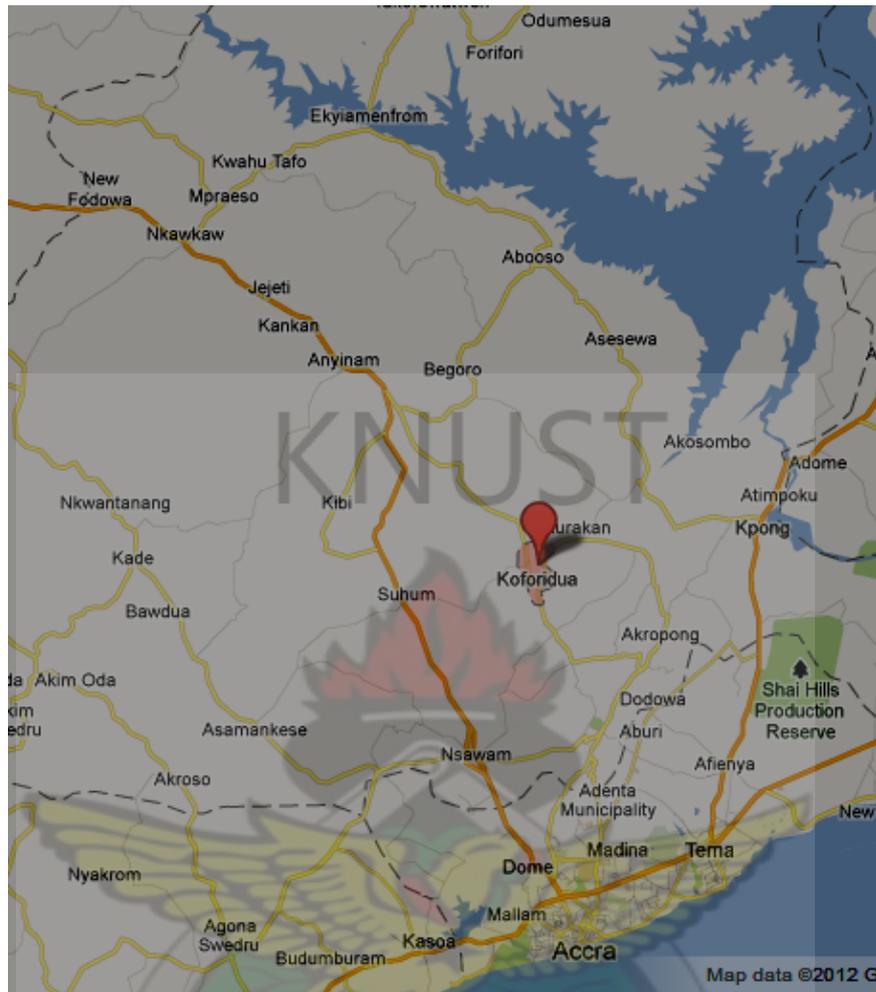


Figure 5.1 Map of Koforidua (Source: google.com.gh/maps, 01,2012)

