ASSIGNMENT AND CONTROL IN ROAD TRAFFIC NETWORKS

by

Mustapha Omar Ghali

A thesis submitted to City University in fulfillment
of the requirements for the degree of
Doctor of Philosophy

London, 1991

To my parents

ACKNOWLEDGEMENT

I am greatly appreciative of the immensely motivating discussions with Mike Smith and wish to sincerely thank him for his suggestions, profoundly careful reading of the thesis, and for his sustaining encouragment to me. His inspiration has been invaluably helpful.

Thanks are also due to Professor Richard Allsop for his advice in the early stages of the research.

I would like to express my gratitude for the financial support provided by The Harriri Foundation, without which this research would have never been possible.

ABSTRACT

The thesis studies the traffic assignment problem in the context of:

(i) deterministic queue modelling where demand and link traffic flows are time-varying,

(ii) the steady-state network design problem, and

(iii) signal-controlled road network.

In studying the traffic assignment problem in the context of deterministic queue modelling, a model is proposed to determine time-varying link flows in congested road networks where drivers are assumed to be cooperative in minimising total transportation costs. The model is approximate for a network of general topology where there is more than a single commodity and many bottlenecks, but optimal when there is only one active bottleneck along the routes connecting each origin-destination pair.

In regards the second context, the thesis offers a method for solving the network design problem that is similar in outer form to the method given in Marcotte (1983). The difference here is being in the way the subproblem, step 2 in Marcotte's method, is attempted. Some computaional results are provided, after having implemented the method in a computer code. Further, the method is compared against other familiar methods that are found in the literature.

As for studying the traffic assignment problem in the context of signal-controlled road networks, the thesis deals with time-variant and time-invariant control and traffic assignment. In both, this is done by alternating between assignment and control, so as to keep the traffic lights in tune with the link flows. Control here is expressed by means of

three traffic control policies, with a view to comparing network performance under each of these policies and at different levels of congestion. The three control policies in time-invariant control and assignment are: the standard "delay minimisation" policy, as stated in Allsop (1971), the standard equisaturation method proposed by Webster (1966), and the P_0 policy, introduced in Smith (1979b). The control policies in the steady state are compared within a gently rising control model that is described in this thesis to simulate the long run effect of the signal control policies on traffic redistribution. Regarding time-variant control and assignment, CONTRAM (Leonard *et al* (1978)) was used as an assignment program, and modified to incorporate two redefined policies of the three control policies and account for vehicle occupancy.

CONTENTS

1. INTRODUCTION

1.1 GENERAL

A major problem in road traffic is how to design a road network so that the design parameters, such as street widths or the settings of traffic lights, are those which reduce congestion the most, result in least construction costs if measures such as the construction of new roads are to be taken, while acknowledging that link flows and the design parameters are interdependent. Link flows and the design parameters are being interdependent in that the changing of road conditions may affect the link flows or driver's behaviour, which in turn may affect the final design of the road network or links.

This problem is of particular interest, especially nowadays, where, due to the ever increasing number of private vehicles, road networks are consequently becoming exceedingly congested.

To express the interdependence between link flows and the design parameters, one of two principles that model drivers' behaviour, due to Wardrop (1952), is assumed.

Wardrop's Two Principles

Wardrop's two principles are:

1) "The journey time on all routes actually used are equal, and less than those which would be experienced by a single vehicle on any unused route."

2) "The average journey time is minimum."

Wardrop's first principle: This implies that each driver on the network is seeking to minimise his/her own travel costs, experiments with several routes and eventually chooses the least costly route. The travel cost could be considered as a generalised term that is used to indicate a composite of disutilities, such as travel time, level of service and, perhaps, discomfort. This principle actually gives rise to a *user equilibrium* pattern of flow, as all used routes have equal costs and any unused route is at least as costly as a used route.

Wardrop's second principle: Underlying this principle is the statement that flows are distributed over the network as such the sum of the total travel time of all road users is minimal. So that flows could be distributed in such a manner, this principle assumes drivers as cooperative in travelling in the road network, so as to consider the costs which they inflict on others and follow the routes that minimise the total travel time. In cooperating, some drivers may have to take longer routes, but those which reduce total travel time. Because this principle assumes drivers as cooperative in minimising total travel time, it indicates the minimum costs the system may incur, or the *system optimum* pattern of flow, though, from a traffic modelling point of view it may not be possible to assume drivers to be cooperative. This is due to that drivers reduce their own travel time rather than the total system costs. Obviously, by saying that when drivers cooperate to result in minimum travel time, they result also in travel time that is less than the travel time due to users following a user equilibrium strategy, or their own minimum travel time, as in Wardrop's first principle.

The Design of a Road Network in the Steady State

In a steady state context, where link flows and entry flow to the network are assumed to be time-invariant, the problem of designing a road network, or *the network design problem* as commonly known, has received much attention and algorithms already exist to solve a class of this problem when the second principle of Wardrop is assumed to express the interdependence between link flows and the design parameters. However, when it is needed to model drivers' behaviour in accordance with Wardrop's first principle, there is as yet no satisfactory algorithm to deal with large problems. As a result, heuristic methods, discussed in the next chapter amongst other methods, have been suggested by many authors. The problem with these heuristic methods is that they may result in a poor network performance, as it is shown in Smith (1979a) when the design parameters are signal settings.

The Design of a Road Network in the Dynamic State

Apart from the first substantial attempt by D'Ans and Gazis (1976) (discussed in Chapter 2) to model traffic in a time-varying fashion, all the work done on the network design problem has considered traffic in a steady state. This implies that vehicles started earlier than others are assumed to have *no effect* on those travelling later. In D'Ans and Gazis (1976), traffic is assumed to be in a dynamic state or *time-dependent* in both entry to the network and flow on links, and delays are expressed by means of queues, as they develop and dissipate on each link of the network. In contrast with the steady state, regarding entry flow to the network and link flows as time dependent, while taking queueing into account, is a more realistic approach for modelling peak hours and congested traffic networks.

Determining Design Parameters for Fixed Flows

Because of the absence of an algorithm to solve the network design problem efficiently, in some studies, link flows, or the flows of the routes connecting each entry and exit node in the road network, are considered as exogenously given and insensitive to the effect of road changing conditions or the resulting design parameters.

In the steady state, particularly when the design parameters of the network design problem are the settings of traffic lights, algorithms, such as TRANSYT (Vincent et al (1980)) and the method due to Allsop (1971), may be used.

In the dynamic state, also when the design parameters of the network design problem are the settings of traffic lights, some efforts have been made by D'Ans and Gazis (1974) to determine time-varying signal settings when the route flows are fixed. But, due to some problems that are pointed out in Chapter 5, no satisfactory algorithm is yet in common use.

Determining Link Flows for Fixed Design Parameters

If one is only concerned with resulting flows due to given road conditions, then the network design problem becomes the traffic assignment problem, for which efficient algorithms exist in the static state under some assumptions, but not in the dynamic state. These assumptions and the methods available to solve the traffic assignment problem in the steady state are given in the next section. Then, the following section is specified for the work and recent developements on the assignment problem in the dynamic state.

The static traffic assignment problem

The static traffic assignment problem has been under intensive research and the literature on its developement is vast. Many methods of solution and the different formulations are described. From a simple, idealistic planning tool and easy-to-solve formulation, where costs are assumed to be congestion-free, the traffic assignment problem has been expanded to deal with more general and sophisticated cases, such as the dependence of link costs on link flow alone (Beckmann et al (1956)), and, on flow on other links as well, such as at junctions (Smith (1979b) and Dafermos (1980).) When the travelling costs on a link are flow-dependent on the link alone, Beckmann *et al* (1956) formulated the user equilibrium problem as an equivalent convex optimisation problem in the static state; an efficient solution method has been suggested in LeBlanc *et al* (1975).

However, as soon as more general cases are accounted for, for example, at junctions where link travelling costs are dependent on flow on other links in addition to the flow on the link itself, the equivalent optimisation problem ceases to exist, least under strong conditions which might not be met in practice (see Heydecker (1983)). Furthermore, many solutions may exist. Smith (1979b) gives conditions on the link cost function in a steady state context which guarantee uniqueness and offers a condition which if satisfied, the resulting flow pattern is then in equilibrium. Also, he offers an objective function, together with a descent direction, that can be used to calculate a Wardropian equilibrium pattern of flow.

The dynamic traffic assignment problem

Some researchers in road traffic modelling have considered traffic flows in a dynamic and deterministic context to allow for the build up and dissipation of queues, together with time-varying but fixed demand. Others have also dealt with variable departure times, that is when drivers have the choice of travelling time, but restricted to only a single route and a single commodity problem, as in Smith (1984) and Hendrickson *et al* (1981), or for an idealised network as in Mahmassani *et al* (1985).

A number of the researchers who have studied the assignment problem with fixed demand or departure time, have formulated this problem as a mathematical programming problem, whilst others used simulation. Amongst those who approached the assignment problem as a mathemmatical programming problem are D'Ans and Gazis, Merchant and Nemhauser (1978) (M-N), Carey (1987), Zawack and Thompson (1987), and Wie *et al* (1989). As for using simulation, the model due to Yagar (1970), the work of Leonard *et al* (1978) embodied in CONTRAM, and the work of Hall *et al* (1980) in SATURN, are most prominent. Recently, Smith and Ghali (1990) studied the dynamic assignment problem also.

Next, a brief comparative study and a discussion of these approaches are attempted.

Simulation models: CONTRAM and SATURN: The model due to Leonard *et al* (1978) has been the most successful amongst the simulation models in the way it determines a Wardropian equilibrium. The model due to Leonard *et al* (1978), as opposed to the model in SATURN, which is designed to determine an equilibrium pattern of flow as well, is based on a valid queueing model that models queuing delays explicitly rather than using cost functions that are constructed to model queues, such as Webster's function or the BPR function (see § 2.2.2 for a definition of the BPR function.) Whereas the assignment in SATURN is a steady state assignment, CONTRAM has a unique feature in that the assignment treats packets of vehicle rather than flow profiles and the routing of packets is done in accordance with a time-varying minimum path. As the packet is routed, the delays at junctions are calculated as the differnece between the time when the packet joins the queue and the time when it exits from the queue.

Mathematical formulation; D'Ans and Gazis's model: In an effort to construct a mathematical programming model to deal with traffic signal control for oversaturated junctions, D'Ans and Gazis (1976) presents a store-and-forward network model which Zawack and Thompson (1987) seem to reformulate again and define as a time-space network model. The D'Ans and Gazis model results in a system optimised pattern of flow and is unable to deal with more than a single commodity problem.

Mathematical formulation; Carey's and M-N model: Likewise, the Carey (1987) model, which is in fact a modification of the Merchant and Nemhauser (1978) model to yield convexity in the constraint set, also results in a system optimum solution and deals only with a single commodity problem, as in D'Ans and Gazis (1976). In order to have a convex constraint set, Carey (1987) introduced flow control variables. These flow control variables are also introduced in the model due to D'Ans and Gazis (1976), as well as in Zawack and Thompson (1987), but, only Carey (1987) seems to have highlighted. (See Chapter 5 for an example on flow control variables.) It should be mentioned in passing that the model due to D'Ans and Gazis (1976) takes account of travelling costs as well as queuing costs, as opposed to the Merchant and Nemhauser (1978) model, where only queueing costs are optimised, though flow-dependent cost relations are incorporated into the constraint set to restrain the exit flow from each arc.

Mathematical formulation; Wie's *et al* model: By employing the theory of optimal control, Wie *et al* (1989) studied the model given in Merchant and Nemhauser (1978), as well as Carey (1987), as a time-continuous model and which they describe as an instantenous system optimum, after having been solved. Using a similar argument, Wie *et al* (1989) also extend the instantenous system optimum formulation to an instantenous user equilibrium formulation, which they define as a user optimsed traffic assignment model. However, the instantenous user optimised traffic assignment model they suggest, does not reflect driving habits, as drivers usually anticipate travelling costs before they arrive at bottlenecks or junctions. In addition, solving for the instantenous minimum path, may result in looping in some circumstances, as the example given in Ghali (1991) shows.

Recent developement; the dynamic equilibrium assignment problem: While much efforts have been expended to construct formulations that determine a system optimum solution, none has been in order to formulate a mathematical model to determine an equilibrium pattern of flow (uniqueness is still an open question besides existance). The reason perhaps could be attributed to the difficulty of finding an objective function that could determine an equilibrium pattern of flow. This is in contrast with the steady state equilibrium problem, where determining a user equilibrium is easy for separable (Beckmann *et al* (1956)) and non-separable, but monotone functions, as in Smith (1981), as there is an objective function that can be optimised to determine an equilibrium pattern of flow. Smith and Ghali (1990) employ montonicity, which had been earlier applied by Smith (1979b) to the steady state equilibrium problem, to prove that the user link cost function is monotone and non-symmetrical, as later arrivals do not affect earlier arrivals, but the opposite is not true. This directly rules out equivalent mathematical optimisation formulations, unless perhaps as in Smith (1984). In Smith and Ghali (1990) a mathematical argument is also given to show that montonicity at the route level is no longer attained at the link level, which is the opposite of what the case is in the steady state equilibrium problem. This implies that the problem is more difficult. Nonetheless, a montonicity proof is given in Smith and Ghali (1990) for some single bottleneck case with many origin-destination pairs, which is rather an advance on the conventional single bottleneck case with a single origin-destination pair found in the literature.

A related static assignment problem: A comment regarding the model offered by Zawack and Thompson (1987), as well as D'Ans and Gazis (1976), in the context of time-space, or store-and-forward, network is that this model has a common feature with the traffic assignment problem in the static state when the link flows are restricted by a scalar quantity or the capacity of the links and when the cost function is continuously increasing and convex, such as a linear function. This latter problem has been investigated by Thompson and Payne (1975), where they offer an argument which differs from the one given in Potts and Oliver (1972). The argument given by Thompson and Payne can be put in the following form: as the flow constraints might be active at the solution, thus giving rise to positive lagrange multipliers, the addition of link costs along used routes, together with the multipliers where they apply, make all route costs equal. Thompson and Payne (1975) interpret these lagarnge multipliers as implicit queuing delays. But, since the Zawack and Thompson (1987) model account for queues explicitly, the lagrange multipliers resulting on solving the time-space network, actually make up for the difference in arrival time.

1.2 CONTRIBUTION AND ORGANISATION OF THE THESIS

Throughout the thesis, demand is assumed to be given and fixed.

As indeed traffic is a dynamic phenomenon, it would be an ideal objective for this thesis to be involved solely with dynamic state models that assist in road design when the interaction between the design parameters and link flows is incorporated into a single model. However, the thesis falls short of fulfilling this objective and deals with the following:

1- On the static state side, the thesis offers an algorithm for the network design problem when the relation between the design parameters and link flows is expressed in accordance with Wardrop's first principle.

2- Further in the static state and when the design variables are signal settings, a commonly steeply rising cost function, due to Webster and Cobbe (1966), is employed. This function results in infeasible boundaries and any assignment method that incorporates such function requires an initial feasible solution. A method is suggested in this work for this problem and is applied to compare steady-state network performance under three different traffic control policies. The control policies are P_0, introduced by Smith (1979b), the delay minimisation policy of Allsop (1971) and Wester's policy (1966). A feature of the method suggested in this work is that it simulates how some control policies fail to accommodate all the demand as the network is increasingly loaded.

3- On the dynamic state side, the thesis presents a dynamic traffic assignment model which results in an approximately system optimum in a multicommodity and many bottleneck newtork, and system optimum in the case where there is a single bottleneck along each route joining each origin-destination pair, as in Smith and Ghali (1990) and Smith (1991). The pattern of flow obtained by this model can be compared with the dynamic equilibrium pattern of flow determined by CONTRAM (Leonard et al (1989)) and any congestion tolls deduced.

4- The thesis is also involved in comparing the above-mentioned three traffic control policies, but in the dynamic state. For this, it was first required finding a corresponding version for each of P_0 and delay minimisation in a dynamic context, while Webster's policy was used as it is implemented in CONTRAM.

The thesis is organised as follows. *In Chapter 2*, a literature review is included on the steady-state network design problem when the design parameters are the width of links and signal settings. Also, when the signal settings of traffic lights are the design parameters and drivers' behaviour is according to Wardrop's second principle a review is provided for this problem when both flows and the signal settings are time-dependent. *In Chapter 3*, the method mentioned in 2- above is given, together with results on the performance of each of the traffic control policies. *In Chapter 4* the algorithm for the static-state network design problem is presented. Test networks, as well as a comparison of the algorithm suggested against some other algorithms found in the literature, are given. *Chapter 5* describes the dynamic traffic assignment model mentioned in 3-, above. *In Chapter 6*, the dynamic-state traffic control policies investigated and the results of applying these policies to a number of test networks are presented.

2. THE NETWORK DESIGN PROBLEM: A SURVEY

2.1 INTRODUCTION

The equilibrium network design problem literature in the steady state is rich in the various methods undertaken to offer a solution method to this problem. The main reason basically for the various approaches attempted can be attributed to the user equilibrium constraint introduced to model drivers' behaviour in accordance with Wardrop's first principle. This constraint is rather a logical constraint and has been represented mathematically in different forms. Smith (1979b) devised a formulation in a variational inequility context; another approach has been due to Tan *et al* (1979). Although the user equilibrium constraint has been put in a mathematical form, it is non-convex, and this has made the problem difficult to solve and has left the problem of designing an efficient algorithm open. If this equilibrium constraint is discarded, the problem loses much of its significance as a road traffic model and becomes the system optimised problem for which an efficient method exists, as will be seen later.

To avoid the difficulty of solving the static network design problem exactly, many heuristic methods have been suggested as an alternative. Nonetheless, certain heuristic methods, might in some networks result in a very poor solution, as Smith (1979a) shows in a simple example.

While the static network design problem has been under intensive research, the problem in the dynamic state has received little or no attention. The last section of this chapter includes a background to the dynamic problem and underlines the major problems that are yet to be resolved before it may be conceivable to formulate a model for the dynamic network design problem.

This chapter surveys the different formulations found in the static and dynamic state literature on the network design problem and discusses the efforts made in order to find a solution method corresponding to each of the formulations.

The survey is confined to the problem of determining signal settings and capacity of links as design parameters of the network design problem in the steady state, and only signal settings in the dynamic state.

2.2 THE STATIC NETWORK DESIGN PROBLEM

The survey groups the methods into heuristic and exact. As the network design problem is non-convex, the word "exact" in this sense means that the solution point obtained is locally optimal, if not globally optimal.

Heydecker (1986) provides a good review to many of the algorithms that have been found in the literature. Here, this review complements Heydecker's review and gives remarks that are not mentioned there, and which have developed since that survey.

2.2.1 Heuristic Methods

Foremost amongst the heuristic methods is the iterative procedure suggested by Steenbrink (1974), as far as the link capacity problem is concerned, and that due to Allsop et al (1977) as far as traffic lights are concerned. Belonging to the same category as well is the method described in Poorzahedy *et al* (1982) for determining the capacity of links. The method of Poorzahedy *et al* (1982) becomes nearly exact for certain levels of demand and for some cost functions if a budget constraint is imposed.

Iterative Procedure

Notations: For each link *i*, the following is defined.

c_i = travelling cost on link *i*.

v_i = flow on link *i*.

w_i = capacity of link *i*.

a_i = uncongested travel time.

b_i = congestion coefficient.

The BPR function: This is of the form $c_i(v_i,w_i)=a_i+b_i(v_i/w_i)^4$.

The iterative procedure has been long researched as an alternative and heuristic to the network design problem. It was first suggested by Steenbrick (1974) for the link capacity problem, and then by Allsop *et al* (1977) for the signal control problem, after Allsop (1974) had shown, by means of an example, that route choice can be beneficially influenced by changing the green setting of traffic lights in an example network.

The procedure consists essentially of solving an equilibrium assignment problem to determine link flows (Beckmann *et al* (1956); Dafermos *et al* (1969); Smith (1979b)) and then selecting the design parameters. These two steps are iterated until a pair of flow and design parameters is obtained, at which neither a traffic management body needs to change the design parameters, nor the road users who minimise their own costs need to change their routes. Thus the system has fallen into a stable point.

In the remanider of this section, the discussion is confined to the signal control problem.

In a stability study to investigate whether the iterative procedure does arrive at a stable flow, signal setting pair, Marcotte (1983), and later Smith (1985), introduced restrictions on the link cost function which guarantees a feasible and unique point for the iterative procedure (of the signal problem) by solving a single optimisation problem. Marcotte suggests the use of the BPR function which results in a unique solution. If this cost function is considered in the formulation given in Poorzahedy *et al* (1982) (see below), then the method of Poorzahedy *et al* becomes similar to the iterative procedure. Using the properties of monotonicity, Smith (1985) establishes that the policy suggested by Poorzahedy *et al* (1982) is one of a family of policies offered in Smith (1985). The link cost function suggested in Smith (1985) could be likened as the Webster's delay function.

An advantage of the iterative procedure, on the one hand, is that it solves the signal control problem if the conditions mentioned in Poorzahedy at al (1982) are met; that is if the cost function used is the BPR and the constant term, a_i, is zero. However, this method becomes inaccurate if it was applied to a problem where the cost function has both the constant and congestion terms and demand was moderate, and fairly accurate if demand was reasonably low or high, where for low demand the first term dominates, and for high demand the second term dominates. This can be seen clearly from the following. Since the total system costs, TC, are of the form

$$TC = \sum_{i=1}^{I} v_i(a_i+b_i(v_i/w_i)^4),$$

and if v_i is much less than w_i, then the term v_ia_i dominates the term

$v_ib_i(v_i/w_i)^4$. Also, if v_i is much greater than w_i, then the term $v_ib_i(v_i/w_i)^4$ dominates the term v_ia_i, and the problem becomes similar to that of Poorzahedy et al (1982). The reason why if v_i is much greater than w_i, the term $v_ib_i(v_i/w_i)^4$ dominates the term v_ia_i, is due to the power 4.

Moreover, the iterative procedure is able to solve large traffic networks, as algorithms are readily available to determine an equilibrium pattern of flow for a given and fixed signal settings, in addition to algorithms to obtain signal settings for given link flows. A further advantage of the iterative procedure is that it may yield an upper bound on the value of the system costs for the equilibrium network design problem (see Heydecker and Khoo (1990)).

On the other hand, Smith (1979a) was first to show clearly that if the cost function used to model delays is Webster's cost function in the iterative procedure and the traffic control policy is the delay minimising or Webster's policy, then there may not be a solution point to the iterative procedure, or a stable point; Heydecker shows also that Webster's policy is non-monotone. Smith (1979a) backs up his argument by means of a simple example and offers a policy, termed as P_0, that takes account of drivers' behaviour by keeping the road users away from points at which the system incurs extremely high costs or jams. The next chapter offers results on this policy in addition to results on the "delay minimisation" policy and Webster's policy, so as to compare the effect of different policies on congestion.

<u>System Optimised Approach</u>

To circumvent the difficulty introduced into the network design problem by the user equilibrium constraint, Dantzig et al (1979) suggested solving the network design problem without this constraint. This results in a system optimum. At the solution, Wardrop's second principle is consequently satisfied, but, if the design parameters were implemented and drivers were allowed to choose their least costly routes, it is not known by how much the system costs corresponding to the new pattern of flow, deviate from the system costs at the solution of the network design problem.

This problem is easy to solve and efficient methods are available, such as in Dantzig *et al* (1979).

The major advantage of this approach is the lower bound in costs it provides, while an upper bound may be obtained if the design parameters were implemented and an equilibrium flow found.

An Equilibrium Approach

This heuristic is due to Poorzahedy *at al* (1982) and could be described as integrating the user cost function with respect to the flow variable and then minimising the sum of the integrals for all links with respect to both the flow and design parameters.

As aforementioned, this heuristic is useful if the network is extremely congested or congestion free and the cost function used is similar to the BPR function, as Poorzahedy *et al* point out. The method does not account for all possibilities of level of demand and all cost functions. For instance, if Webster's cost function is used to determine signal settings, this does not have the properties of the BPR function.

Penalty Function Approach

As the network design problem involves two objective functions; the objective function of road users who minimise their own percieved costs, and the planner's objective function in endeavouring to keep the total costs minimal; Ben-Ayed *et al* (1988) formulated the problem as a Bi-Level Programming problem, by solving a convex combination of the planner's objective function and the road users' objective function. They suggested the use of Bard's algorithm as a solution method to their formulation of the equilibrium network design problem. However, Marcotte (1988) provided an example which shows the inability of Bard's algorithm to define an optimal point, and this has reduced the significance of their formulation.

In a different context, Heydecker (1986) indicates the difficulty with the approach suggested by Ben-Ayed *et al* (1988), which Heydecker likens as a penalty function approach, and, instead, suggests the use of the objective of Smith (1984) which is developed in a study to generalise the traffic assignment problem to cover inseparable cost functions and which can be used as an indicator to find a descent direction towards an equilibrium. This objective function has an advantage in that it carries information that could be used to find a descent direction at any point either with respect to the design parameters or the link flow variables, or, with respect to both. This feature is not shared by the equivalent optimisation formulation of the equilbrium assignment, due to Beckmann *et al* (1956), for separable cost functions.

Perhaps, Heydecker's suggestion of using Smith's objective function could be implemented in a heuristic to find a direction which reduces the drivers' costs, or Smith's objective function, while causing the least increase to the total system costs, if one started, say, from the system optimised solution (Dantzig *et al* (1979)).

One of the problems that might be faced, however, with such an approach is that although the direction offered in Smith (1984) is descent for the users' cost, and the gradient of the planner cost function could be determined while maintaining feasibility of flows, combining both somehow is not necessarily a gradient, unless perhaps they are steepest descent directions or that they are not obtained by solving a linear program or a minimum path search.

<u>A Constraint Approximation Approach</u>

Most recently, Heydecker and Khoo (1990) suggested a linearisation approximation of the equilibrium constraint by linearly regressing the flow values obtained upon solving an equilibrium assignment for each of five, or so, different step lengths chosen along a certain direction which relates the design parameters as a function of the step length made. More clearly, a relation is used to express the design parameters between two points in terms of a step length λ. Then for a number of step lengths, the design parameters can be calculated from this relation. The design parameters thus obtained are used to determine the corresponding equilibrium patterns of flow. Now the equilibrium patterns of flows are fitted by an approximate relation in terms of the step length. Thus, two relations are formed in terms of the step length. The first expresses the design parameters in terms of the step length, while the other expresses the equilibrium flows in terms of the step length. These two relations are used to determine the value of the step length which minimises the total system costs. Having obtained a value for the step length, hence a pair of flow and design parameters, another direction is then explored, similarly by fitting a new linear regression relation as above. This process is repeated until no further reduction in the total system costs is possible.

Heydecker and Khoo (1990) apply this method to determine signal settings, and they propose spanning the feasible set of green lights at each traffic light independently along several directions that are equivalent in number to the number of stages at each traffic light. They provide formulae for these directions, but it is not clear why and how these directions are specified, nor upon which mathematical argument these are based.

Viewing the method in the context of determining optimal link capacity variables, it could be likened with the method of Abdulaal and LeBlanc (1979) (see below.) In the method of Abdulaal and LeBlanc, a move is made as soon as a new point in the design parameter set is found favourable, after having determined the resulting equilibrium pattern of flow and monitored the system objective function. In Heydecker and Khoo (1990), a move is made only after having fitted an equlibrium flow relation with a number of different step lengths (they suggest five step lengths), and so on for all possible directions in the design parameter set. Therefore, this method does not appear to have made any significant improvement over the method of Abdulaal and LeBlanc (1978), nor it does seem to have cut down the number of equilibrium assignment problems need to be solved to find the response of drivers for any possible movement attempted.

2.2.2 Exact Methods

Unlike the heuristic methods which can deal with reasonably large networks, exact methods on the other hand have been less efficient, if not efficient at all. They have been mainly applied to small networks in order to check or offer an exact solution which could be useful only from a theoretical point of view. This is due to dimensionality and computational time problems.

The methods which are described next are those due to Gershwin et al (1978), Marcotte (1983) and Abdulaal et al (1979).

Constrained Minimisation Problem

Notations: The following notations are needed.

c^*_{od} = minimum cost path between the *o-d* pair, i.e. $c^*_{od}=\min c^r_{od}$, $\forall r \in \mathcal{P}_{od}$. [2.1]

c^r_{od} = travelling costs on path *r* from origin *o* to destination *d*.

f^r_{od} = flow on path *r* from origin *o* to destination *d*.

m_{od} = demand between the *o-d* pair.

$\mathcal{O}$ = set of origin nodes.

$\mathcal{D}$ = set of destination nodes.

$\mathcal{P}_{od}$ = set of paths connecting *o-d*.

$\mathcal{F}$ = feasible set of path flows that can de defined as:

$$\sum_{r \in \mathcal{P}_{od}} f^r_{od}=m_{od}, \qquad \forall o \in \mathcal{O},\ \forall d \in \mathcal{D},$$

$$f^r_{od} \geq 0, \qquad \forall r \in \mathcal{P}_{od},\ \forall o \in \mathcal{O},\ \forall d \in \mathcal{D}.$$

$$\delta^r_{ij,od}=\begin{cases} 1 \text{ if link } (i,j) \text{ is on path } \mathcal{P}^r_{od}, \\ 0 \text{ otherwise.} \end{cases}$$

v = vector of link flows, having as elements v_{ij}'s.

v_{ij} = flow on link (i,j) that is defined by $v_{ij}=\sum_o \sum_d \sum_r f^r_{od}\delta^r_{ij,od}$.

Λ = effective green time of signal settings.

$\mathfrak{F}$ = feasible set of the green settings.

p = objective function of the total system costs.

Tan *et al* (1979) investigated a direct approach into the problem of signal setting. They express route choice behaviour in a convenient set of inequility equations that are amenable to general optimisation algorithms in the path-flow formulation (see Potts and Oliver (1972) for path- and link-flow formulations). They observed that Wardrop's first principle can be cast into the following form:

$$f^r_{od}(c^r_{od}-c^*_{od}) = 0, \quad \forall r \in \mathfrak{P}_{od}, \forall o \in \mathfrak{O}, \forall d \in \mathfrak{D} \qquad [2.2]$$

Summing [2.2] over all paths r in $\mathfrak{P}_{od}$, this results in

$$c^*_{od}=(\sum_r f^r_{od}c^r_{od})/m_{od}, \quad \forall r \in \mathfrak{P}_{od}, \forall o \in \mathfrak{O}, \forall d \in \mathfrak{D} \qquad [2.3]$$

Relationship [2.1] and [2.2] imply that for any *o-d* pair any unused path r has:

$$c^r_{od} \geq c^*_{od}=(\sum_r f^r_{od}c^r_{od})/m_{od}, \quad \forall r \in \mathfrak{P}_{od} \qquad [2.4]$$

Having expressed drivers' behaviour in the form of exprression [2.4], then Tan *et al* (1979) suggest solving the nonlinear program:

Minimise $P(v,\Lambda)$ [2.5a]

subject to

$f \in \mathfrak{F}$ [2.5b]

$$c^{r}_{od} \geq c^{*}_{od} = \left(\sum_{r} f^{r}_{od} c^{r}_{od}\right)/m_{od}, \; \forall r \in \mathcal{P}_{od}, \; \forall o \in \mathcal{O}, \; \forall d \in \mathcal{D} \qquad [2.5c]$$

$$\Lambda \in \mathcal{F} \qquad [2.5d]$$

The method suggested to solve problem [2.5a-d] is an augmented Lagrangian approach, but found impractical for large scale networks. It is impractical because it requires the enumeration of all possible paths between each origin-destination pair to account for all possible route flows before implementing the augmented lagrangian. Besides, as it is formulated in the path-flow formulation, the constraints [2.5b] and [2.5d] are non-network constraints. This may require including these constraints in the augmented lagrangian objective function, thus more lagrange multilpliers and longer computational time. If these constraints are not included in the augmented lagrangian, then they cause dimensionality problem.

<u>Contraint Accumulation Approach</u>

<u>Notations</u>: These are as follows.

w = capacity variable of links.

$\mathcal{W}$ = feasible set of capacity of links (see Chapter 4).

v_{ij} = flow on link (i,j).

v = vector of link flows, having as elements v_{ij}'s.

$\mathcal{V}$ = feasible set of v.

c = vector of link costs.

p = objective function of the total system costs.

Using the variational inequality formulation of the equilibrium assignment problem as suggested in Smith (1979b), Marcotte (1983) presented an algorithm to calculate the design parameter of each link by applying a partial dual approach (Luenberger (1984)). His method requires first generating variational inequality constraints, or equilibrium constraints, and then solving a subproblem of the form:

$$\text{Minimise } P(\mathbf{v},\mathbf{w})$$

subject to

$$\mathbf{v} \in \mathcal{V}$$

$$c(\mathbf{v},\mathbf{w}).(\mathbf{v}-\mathbf{v}') \leq 0, \ \forall \mathbf{v}' \in \mathcal{V}$$

$$\mathbf{w} \in \mathcal{W}.$$

This subproblem is solved optimally for a working set of equilibrium constraints. The equilibrium constraints are generated on the basis of a minimum path search.

Marcotte originally proposed this method for the link capacity problem, where all links are considered for construction. The introduction of Chapter 4 explains the difficulty of applying this method to a network where only some links are considered for construction or improvement. The same difficulty also arises when solving the signal control problem. This difficulty is termed later as "insufficient control."

Substitution Approach

The last exact method illustrated in this review is the one described in Abdulaal *et al* (1979). This is mainly the Hooke and Jeeves (1961) method. It makes use of the uniqueness property of the equilibrium assignment problem, which follows from the strict convexity of the formulation for separable and strictly convex cost functions, for any set of specified design parameters. The method could be viewed as a constraint perturbation, a constraint satisfaction and then the monitoring of the system objective function, on the basis of which a move is made. If the equilbrium constraint of the network design problem is expressed in terms of the design parameters and flow variables, then the constraint perturbation is done by varying the design parameters along some feasible direction; the constraint satisfaction is achieved by solving just a single equilibrium assignment problem; and lastly the monitoring is achieved by evaluating the system objective function.

An advantage of the method of Abdulaal *et al* (1979) is that it can deal with large networks if the number of design parameters is small. However, as soon as this number is slightly increased then the computational task becomes very expensive.

Formally, the formulation can be put in the following non-differentiable form:

$$\text{Minimise} \quad \mathbf{p}(\mathbf{v},\mathbf{w})$$

Subject to

$$\mathbf{v} = \mathfrak{F}(\mathbf{w})$$

$$\mathbf{v} \in \Psi$$

$$\mathbf{w} \in \mathcal{W}$$

where the non-differentiable relation $\mathbf{v} = \mathfrak{F}(\mathbf{w})$ defines the equilibrium pattern of flow corresponding the design parameters $\mathbf{w}$.

This method some similarity with the method of Marcotte (1983). Here, the design parameters are varied and the equilibrium flow pattern found. There, the flow variables are varied initially and then the design parameters computed are those which bring the varied pattern of flow to equilibrium. In Abdulaal's method, the flow pattern corresponding to any varied design parameters, exists, whereas there may not be a design parameter set which induces equilibrium to any flow pattern as in Marcotte. This is due to the "insufficient control" problem, mentioned above.

2.2.3 An Example

To illustrate the difference between some of the various methods mentioned above, a simple example network is considered.

The network

The network shown in Fig 2.1 has two origin nodes, 1 and 3, and a destination node, 2. Node 4 is assumed a traffic signalised junction with two stages.

The demand from 1-2 is taken as 10 units, from 3-2 as 3 units, and the link cost functions are:

$$t_{14}=1+v_{14}/\gamma_1 c,\quad t_{12}=1+v_{12},\quad t_{43}=5v_{43}/\gamma_2 \text{ and } t_{42}=1+2v_{42}.$$

where γ_1 is the proportion of green time facing link (1,4), γ_2 is that of link (3,4) and c is the cycle time which is taken as 30 secs. No lost time is supposed between the green time of stages.

Exact solution

The problem may be represented in the form:

$$\text{minimise } D= v_{14}(1+v_{14}/\gamma_1 c)+v_{12}(1+v_{12})+$$

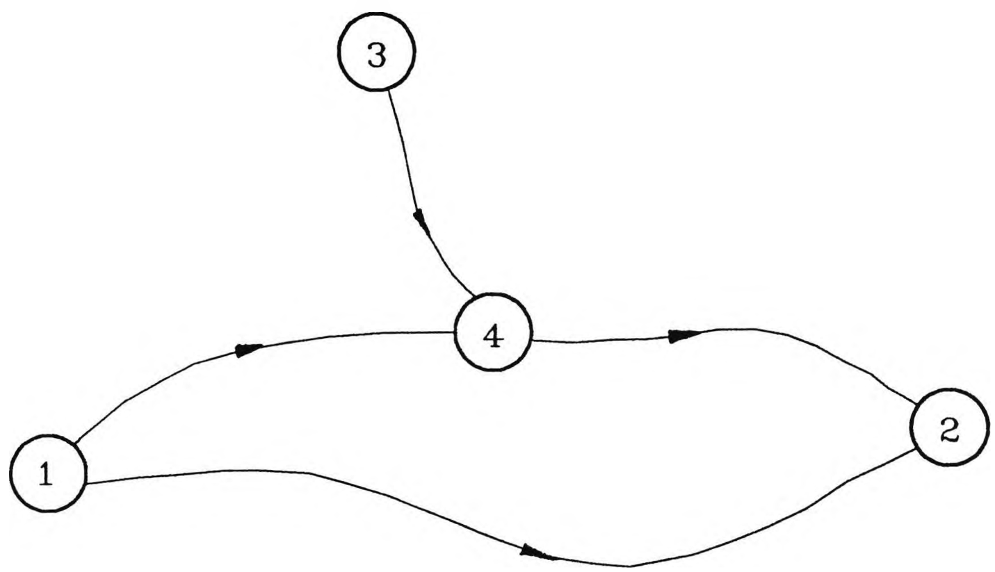


Figure 2.1

$$v_{34}(5v_{34}/c\gamma_2)+v_{42}(1+2v_{42}) \quad [3.6a]$$

subject to

$$\gamma_1+\gamma_2=1, \quad [3.6b]$$

$$1+v_{12}=1+v_{14}/c\gamma_1+1+2v_{42}, \quad [3.6c]$$

$$v_{34}=3, \quad [3.6d]$$

$$v_{14}+v_{12}=10, \quad [3.6e]$$

$$v_{34}+v_{14}=v_{42}, \quad [3.6f]$$

$$0\le\gamma_1\le 30, \quad 0\le\gamma_2\le 30, \quad v_{12}\ge 0, v_{14}\ge 0, v_{34}\ge 0, v_{42}\ge 0. \quad [3.6g]$$

The exact solution of this problem can be shown to corresponds to:

$v_{12}=9.39$, $v_{14}=0.61$, $v_{42}=3.61$, $\gamma_1=0.12$, $\gamma_2=0.88$, and at which D=127.489.

Heuristic solutions

Iterative Approach: This a two stage process. Start first stage of the algorithm with a feasible $\gamma_2=1/3$; hence $\gamma_1=2/3$. To determine the equilibrium flow pattern for the given values of γ_1 and γ_2, substitute in relation [2.6c]. This gives

$$1+v_{12}=1+v_{14}/10+1+2v_{42}$$

Further, constraining the flow variables to the demand, yields

$$v_{12}=550/61, \; v_{14}=60/61 \text{ and } v_{42}=243/61$$

The second stage is to determine, locally, optimal γ_1 and γ_2 for the flows of stage 1. This results in the objective:

$$\min_{\gamma} \quad D=(60/61)^2(1/30\gamma_2)+45/(30-30\gamma_1)$$

On solving, γ_1=0.0265 and γ_2=0.9735

Repeating stage 1 and stage 2 until no change in the flow and the signal variables, gives the values:

$v_{12}=10$, $v_{14}=0$, $v_{42}=3$, $\gamma_1=0$, $\gamma_2=1$ and D=131.3.

<u>System Optimised Approach</u>: If constraint [2.6c] is discarded, the flow and signal variables which result in the minimal value of the objective function, are:

$v_{12}=8.879$, $v_{14}=1.121$, $v_{42}=4.121$, $\gamma_1=8.16/30$, $\gamma_2=21.84/30$ and D=127.489.

Fixing the values of γ_1=8.16/30 and γ_2=21.84/30, and solving an equilibrium assignment problem [3.9], gives:

$v_{12}=9$, $v_{14}=1$, $v_{42}=4$ and D=129.184.

<u>Equilibrium Approach</u>: This requires solving:

$$\min_{v} \quad I=\int_0^{v14}(1+u/\gamma_1 c)du+\int_0^{v12}(1+u)du+\int_0^{v43}5u/\gamma_2 du+\int_0^{v42}(1+2u)du$$

subject to

$\gamma_1+\gamma_2=1$,

$v_{34}=3$

$1+v_{12}=1+v_{14}/30\gamma_1+1+2v_{42}$

$v_{14}+v_{12}=10$

$v_{34}+v_{14}=v_{42}$

$0\leq\gamma_1\leq 30$, $\quad 0\leq\gamma_2\leq 30$, $v_{12}\geq 0$, $v_{14}\geq 0$, $v_{42}\geq 0$, $v_{34}\geq 0$

The solution to this problem is:

$$v_{12}=9.07,\ v_{14}=0.92,\ v_{42}=3.92,\ \gamma_1=4.138/30,\ \gamma_2=26.861/30 \text{ and } D=129.03$$

<u>Comparison</u>

A comparison of the overall delays of the above results obtained on solving, firstly, the iterative method, secondly using an exact approach, and, thirdly by discarding the drivers route choice, shows that delays due to the first method are greater than the other two, and the second, the exact solution, is greater than the system optimised pattern, the third approach used in the example.

2.3 THE DYNAMIC SIGNAL SETTING CONTROL PROBLEM

The earliest attempts perhaps to relate time-varying signal settings as design parameters with time-varying link flows, should be attributed to Gazis (1964) in a model specifically designed to deal, however, with very severly limited network cases.

The model described in Gazis is constructed to calculate optimal time-varying signal settings where route-choice is absent. It is mainly for a single or two, or more, consecutive traffic lights, with no turning movements and with the output flow profile of an upstream traffic light feeding that at the downstream. For such simple network cases, the input flow profile of the downstream traffic light could be readily and analytically expressed in a closed form in terms of the output flow profile of the upstream traffic light.

Though the study cases explored in Gazis (1964) are limited to no route-choice and a single or more traffic light problem, they offer theoretically valuable insights into the problem of delay minimisation. Notably, they introduce contraints on delays that are not accounted for in the steady state, in addition to the need of coupling of the traffic lights so as to express the interaction amongst which.

In the steady state, minimsing delays at traffic lights independently, that is each traffic light is optimised in isolation (Allsop (1971)) from the rest and according to what the current flow on the approaches is, while drivers are routed according to Wardrop's second principle, results in the system optimum of the signal control problem.

But, in the dynamic state, treating traffic lights as isolated and minimising delays at each of which independently, while drivers follow the routing strategy due to Wardrop's second principle, does not unfortunately result in the system optimum of the signal control problem. In Chapter 6, results are included that show that this is indeed the case. There, the traffic lights are considered as isolated and an iterative procedure in a dynamic setting that alternates between solving the system optimum assignment problem and optimising delays at traffic lights, is performed.

In order to generalise the two consecutive intersection cases, treated in Gazis (1964), to complex transportation networks so as to determine optimal control of a system of oversaturated intersections, D'Ans and Gazis (1976) introduce what is currently known as 'store-and-forward' congested networks.

The store-and-forward network introduced in D'Ans and Gazis may be described as one in which a storage capacity is assumed before the exit of each arc and just in front of the node which connects this arc with the "after" arcs. The storage capacity on each arc could be used to store, or hold back, flow, due to the introduction of flow control variables in the D'Ans and Gazis model. As said in Chapter 1, the *flow control variables* are needed to have a convex set of flow constraints.

It will be seen in Chapter 5 that there is an implementation problem associated with the flow control variables for a multi-commodity newtork, where there are uncontrolled links in the network and the flow control variables are positive on these links.

For a multi-commodity network, D'Ans and Gazis state what could be described as that the optimal control of a store-and-forward network requires, in general, three operations:

(a) The optimum allocation of a route to each unit of traffic from its origin to its destination.

(b) The optimum switching at the nodes, determine the allocation of discharge of queues.

(c) Servive of queues be "first-in first-out".

The "first-in, first-out" (FIFO) discipline may be defined as follows: *given two vehicles, the one entering a link first, also exits the link first.*

Despite the rigorous treatments provided in D'Ans and Gazis on the signal setting side, they conclude with the statement that there is no complete methodology for the solution of the general optimisation problem in (a), (b) and (c). Instead, they assume that the route assignment is given and FIFO is satisfied by making an assumption on the various commodities that they are roughly uniformly distributed within each queue. As a second approach, they reformulate the problem as a multicommodity network with controlled turning movements and introduce a FIFO constraint. This constraint is non-convex. Hence, besides the problem associated with implementing the flow control variables, a non-convex problem needs to be solved.

Carey and Srinivasan (1987) also deal with a problem with variable control, but in a different application, in industrial processing and air traffic control, where FIFO contraints are not needed as it is possible to hold traffic back in various storage pockets.

In concluding, due to the difficulties already encountered with the system optimum of the dynamic signal control problem, it is hardly surprising therefore that work on the equilibrium dynamic signal control problem, or the equilibrium network design problem with dynamic demand, is nil.

3. A GENTLY RISING RUSH-HOUR CONTROL MODEL

3.1 INTRODUCTION

The chapter has two objectives. It is firstly concerned with offering a method for the problem of green light allocation of traffic lights, when the traffic lights are kept in tune with link flows and when the demand is gently and steadily increasing, for all origin-destination pairs of a road network. Though the method is a steady state, it does give some indications of congestion buildup in a traffic controllled network.

The second objective is to offer computational results obtained on applying the method suggested here to three road networks, while traffic lights are set in accordance to three different traffic control policies. As will be seen later, the method converges in the limit to the iterative assignment/control procedure which is discussed at some length in Chapter 2, and, hence, the results presented later serve, in addition, as a comparative study on the performance of different control policies of traffic light setting.

The control policies tested were as follows:

(i) the standard 'delay minimisation' policy, stated in Allsop (1971),
(ii) the standard equisaturation method proposed by Webster, and
(iii) P_0, discussed in Smith (1979b).

3.2 GENTLY DYNAMIC ASSIGNMENT AND CONTROL

Essentially, the method suggested is a simulation tool, and is intended as a fast way of comparing in general the effect of different control policies on route choice in a gently dynamic context.

The gently dynamic assignment/control method presented in this work seems practical and avoids the difficulties inherent in a fully dynamic approach to the assignment/control problem. (See Smith and Ghali (1990).) In fact, some results are given on the dynamic assignment/control problem in Chapter 6, but the method given here could deal with large networks. This may not be the case in the dynamic state.

The method allows the use of Webster's delay formula. This formula estimates the average long-run delay to a Poisson traffic stream, and rises asymptotically to infinity as the flow approaches the finite capacity of the road link (which depends on the signal-settings). This steep behaviour needs to be taken into account both in the assignment procedure itself and in the initial choice of signal-settings and flows. Any delay formula having similar features to the Webster's delay formula, could also be used.

In addition to its practicality in a gently dynamic assignment/control problem, an important feature of this method is also that it could be used to avoid the problem of finding an initial feasible point when solving for the system optimum of the signal network design problem if the cost function used is Webster's. This is explained further in Chapter 4.

Moreover, this method is useful to find an initial feasible solution or pattern of flow in a purely assignment context, as in Daganzo (1977), where the cost function is, again, steeply rising as the link capacity is approached, but the signal updating step is discarded in this context.

A Gently Dynamic Control/Assignment Procedure

Given a signal-controlled network and an origin-destination matrix, the procedure could be stated as follows:

1-Choose any initial signal settings that satisfy the green light constraints.

2-Do an all-or-nothing assignment and assign an allowable maximum percentage of the trip matrix.

3-Keeping traffic light settings fixed, solve for an approximate equilibrium pattern of flow if the trip matrix is not fully loaded yet, or, an exact equilibrium pattern if the trip matrix is fully loaded otherwise.

4-Update traffic lights to match new flow pattern due to 3.

5-If the trip matrix is completely loaded, proceed to 6. Otherwise, return to 2.

6-Check convergence criterion: if satisfied, then terminate, else return to 3.

Apart from step 2 in the above procedure and the loading of additional demand from all origins as the method progresses, it is just as in Allsop and Charlesworth (1977), and has been expalined earlier in Chapter 2.

To explain step 2, the network in Fig 3.1, taken from Smith (1979a), is used. This network is composed of three one-way links A, B and C to connect origin x with destination y through the signalised junction J. In Fig 3.2 line S is the supply of junction J and line D is the total demand at the

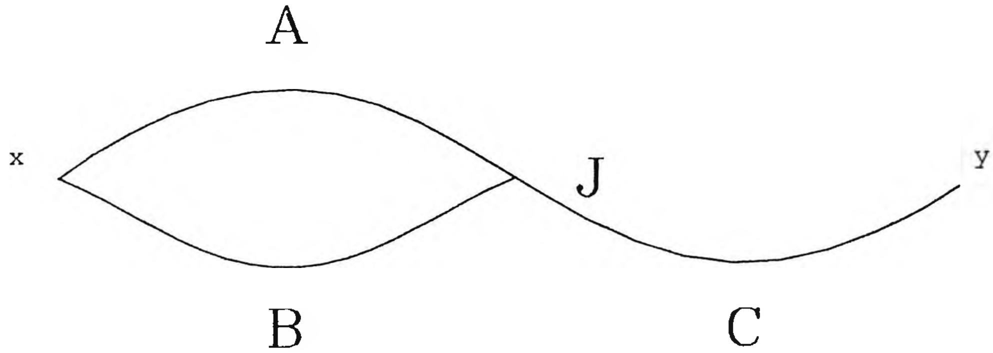


Figure 3.1

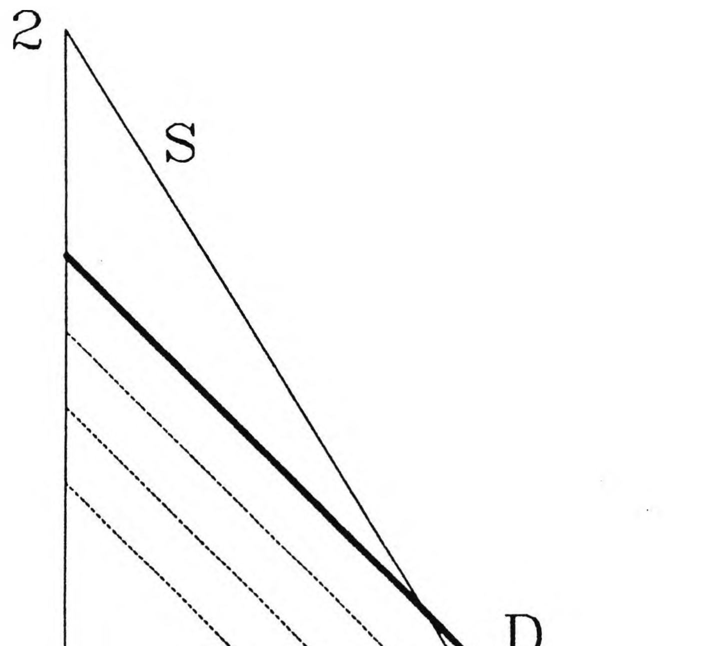


Figure 3.2

junction. The supply is determined by considering that flow v_a and v_b, on arms A and B, respectively, should each be less than the capacity determined by the green settings; that is $v_a < \lambda_a s_a$ and $v_b < \lambda_b s_b$, where λ_a is the effective green time facing arm A, s_a the saturation flow of arm A, λ_b the effective green time facing arm B, and s_b the saturation flow of arm B. But, since $\lambda_a+\lambda_b=1$, then the supply S is determined by

$$v_a/s_a+v_b/s_b < 1.$$

As for the demand D, it is taken such that $v_a+v_b=1$.

Now step 2, primarily determines the minimum path, which incidently may be relatively less congested than any other path. This allows for an increase in the demand corresponding to capacity of links determined by the signal settings, due to the policy employed, by moving the plane D in Fig 3.2 parallel to itself and as shown by the dashed lines for the two-link example. The minimum path determined in step 2 may not always be the path which could accommodate the largest possible increase in the demand, as the constant travelling costs may dominate congestion delays. Although, as the network gets reasonably congested, it is the spare capacity of links on the routes connecting each origin-desitnation pair which counts and determine the minimum path. Therefore, a load increment that is not greater than the available capacity of the most congested link lying on the minimum path could then be added to the flow on all links which are on the current minimum path.

Although an exact equilibrium could aways be solved for in step 3 whenever a load increment is assigned to the minimum path, the reason for determining only an approximate equilibrium pattern of flow, but not when all the demand is loaded, can be justified as follows. Since the concern in this is to study the long-run network performance and determine the capacity of the network so as to find the total demand that can be accommodated when different control strategies are applied, solving for an exact equilibrium becomes less important if the total demand is not yet loaded. However, while running the computer programs, in which the gently dynamic control/assignment procedure is implemented, on the network tests described later, it was noticed that a few Frank-Wolfe iterations (see LeBlanc *et al* (1975)) were indeed needed to achieve reasonable accuracy and drive the flow away from highly congested links.

3.3 TEST NETWORKS

Three networks were used as an application of the procedure proposed in this chapter and to give computational results for three different control policies. The first is shown in Figure 3.3, the second in Figure 3.4, and the third network Fig 3.5.

Network 1

The network of Fig. 3.3 has eight traffic lights, four denoted as A, with each approach having a saturation flow S_a, and four as B, with each approach having a saturation flow S_b, as well. Junctions denoted as F are assumed to be flyovers or have large capacities. In this network, no turning movements are allowed, hence only two origin-destination pairs are considered. The total demand from each origin was taken as twice the value of (S_a+S_b) for five different cases, where S_a was first assumed to be 1 Veh/Sec, and then incremented by 0.25, up to 2 Veh/Sec, while S_b was kept fixed at 1 Veh/Sec.

The link cost function of the network of Fig. 3.3 was assumed to be

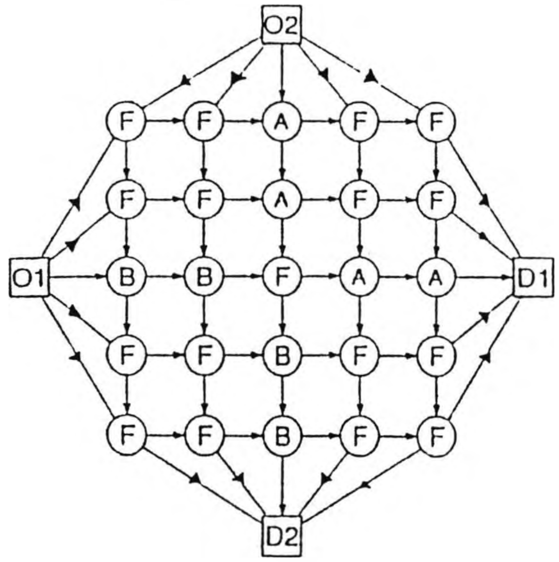


Figure 3.3

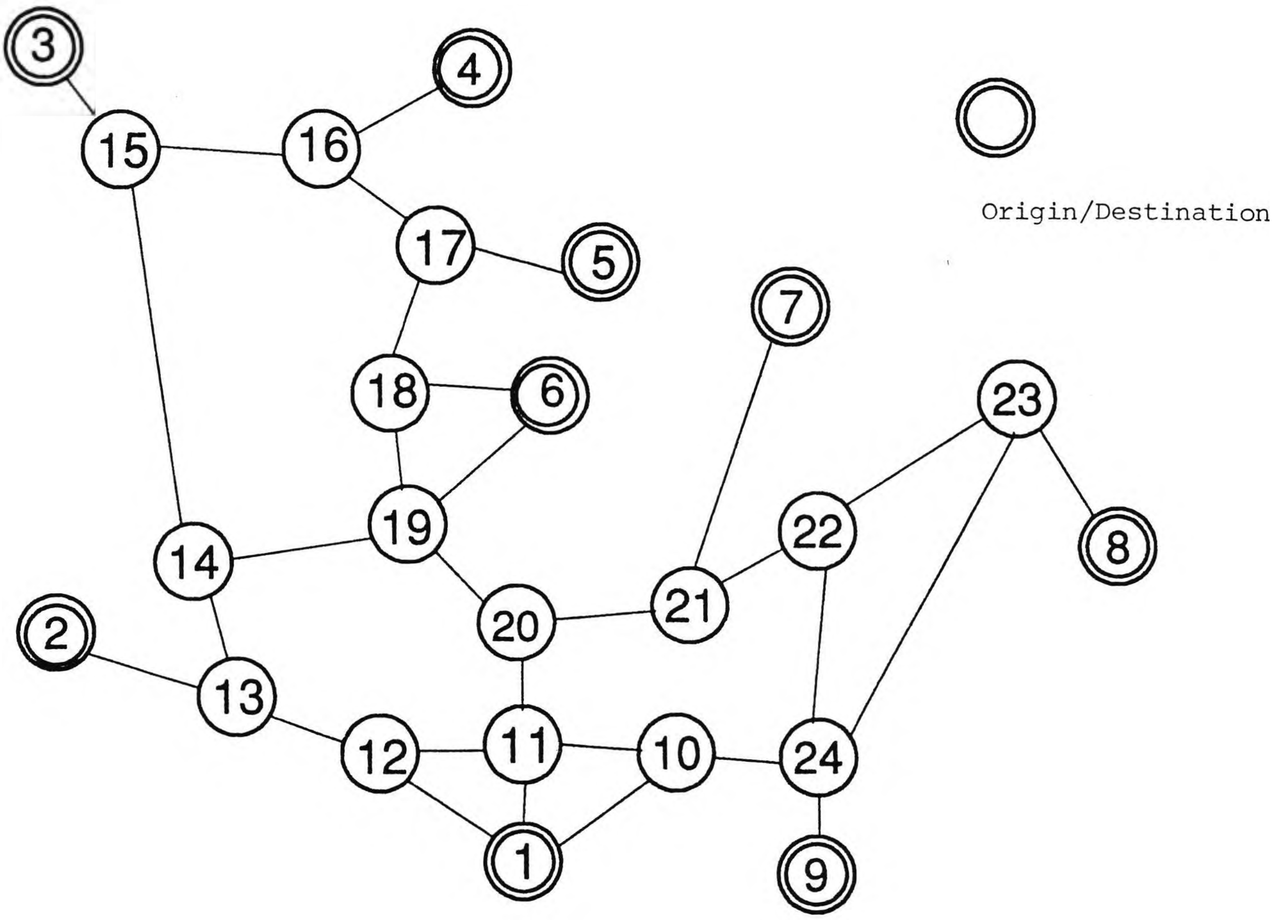

3
4
15
16
17
5
7
18
6
23
19
22
8
14
21
2
20
13
11
10
12
24
1
9
Origin/Destination

Figure 3.4

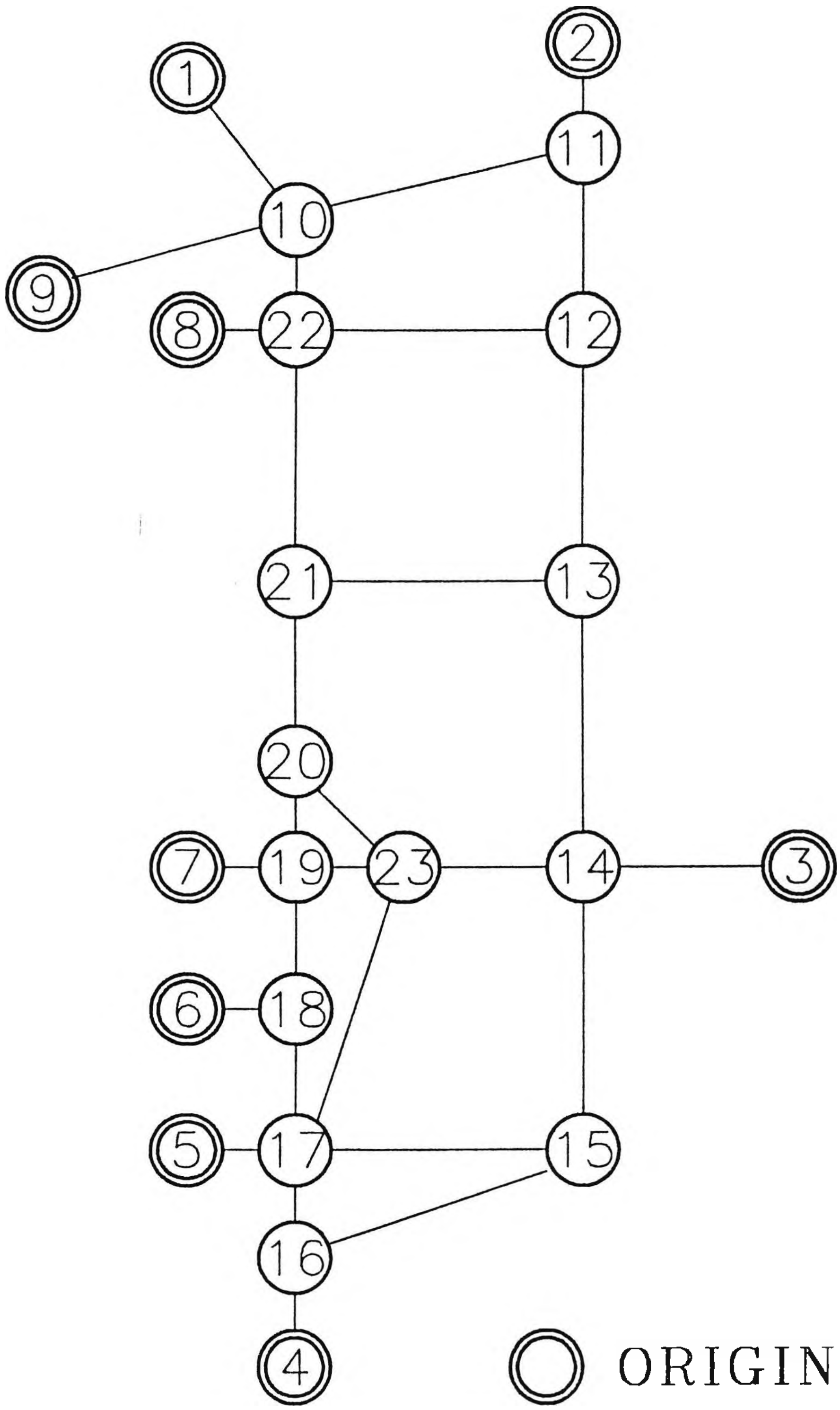


Figure 3.5

composed of a constant term (running costs) and a delay term, due to Webster, if the link in question has at its down stream end a traffic light. The constant term was taken as 280 seconds for the first links on the four outer routes and zero elsewhere in the network. Obviously, the central route is faster than the outer routes if only running costs are taken as a measure of costs between each origin-destination pair. The cycle time of all traffic lights of this network was assumed fixed and equal to 120 seconds. Two stages at each traffic light were needed. No lost time between stages was assumed and a minimum of 1 Sec. green time was imposed on each stage.

Network 2

This network is composed of 9 origin-destination pairs, and all the links are two-way links.

Control in this network is introduced by means of two traffic lights, at junction 11 and 14. The stage structure of each traffic signalised junction was assumed as shown in Table 3.1.

As in Network 1, the lost time in this case between consecutive stages was also taken as zero and the stage minimum green time as 1 second, mainly to disallow situations where a stage green time would otherwise be zero when there is no flow on links within the stage. If the minimum green time of a stage was allowed to be set to zero value, then this might not change during the assignment/control procedure once it has been set as such. Simply, the link with zero green time and modelled using Webster delay function would then have no spare capacity and thus would not be assigned any flow during the assignment process in a subsequent stage.

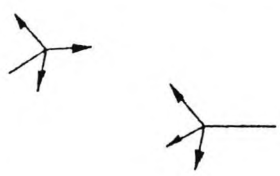
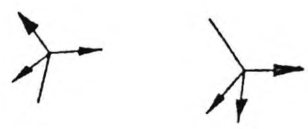
NODE	STAGE 1	STAGE 2
11		
14		

Table 3.1

The cycle time of both traffic lights was taken as 120 seconds, and the total demand to the network from all origin points was 400 Veh/Hr, that is to be loaded incremently as described in Steps 1-6 above.

Again, the link cost function was supposed as in Network 1, a combination of a constant travelling time, given in Table 3.2, and Webster's delay function if the link is signalised. or, the BPR congestion term if the link is uncontrolled. The congestion coefficient of the BPR function was choosen as 2 Sec^4/Veh^5 for all links.

Network 3

The network, shown in Fig 3.4, is composed of 9 origins, each acting as a destination point as well. All the links in the network are two-way links that are uncontrolled.

Apart from the origin nodes (1-9) and node 20, all the other nodes were regarded as signalised nodes, each with a stage structure as shown in Table 3.3. The minimum green time of each stage was taken as 5 seconds and no lost time between consecutive stages was assumed.

The demand between each origin-destination pair was as shown in Table 3.4, and the constant travelling cost of each link as in Table 3.5.

For this network, no BPR congestion term for uncontrolled links was assumed, only a constant travelling cost. For controlled links, Webster delay function in addition to the constant travelling cost was supposed.

Link	Constant Travel Time (Secs)	Link	Constant Travel Time (Secs)
1 - 10	50	1 - 11	20
1 - 12	50	10 - 11	50
10 - 24	50	10 - 1	50
11 - 1	50	11 - 10	50
11 - 12	50	11 - 20	50
12 - 1	18	12 - 11	12
12 - 13	10	13 - 2	16
13 - 12	15	13 - 14	18
2 - 13	15	14 - 13	15
14 - 19	21	14 - 15	21
15 - 14	15	15 - 3	15
15 - 16	17	3 - 15	41
16 - 4	13	16 - 15	11
16 - 17	16	4 - 16	21
17 - 5	12	17 - 16	150
17 - 18	14	5 - 17	21
18 - 6	41	18 - 17	150
18 - 19	150	19 - 14	11
19 - 18	11	19 - 6	31
19 - 20	150	6 - 18	41
16 - 19	21	20 - 11	20
20 - 19	150	20 - 21	71
21 - 20	150	21 - 7	19
21 - 22	16	7 - 21	11
22 - 24	16	22 - 21	150
22 - 23	11	23 - 24	40
23 - 22	150	23 - 8	40
8 - 23	43	24 - 9	12
24 - 10	11	24 - 22	13
24 - 23	41	9 - 24	31

Table 3.2: Assumed constant link travelling cost of Network 2.

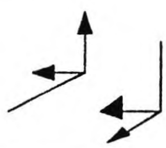
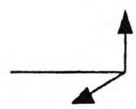
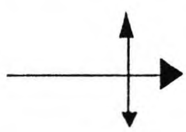
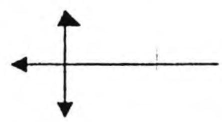
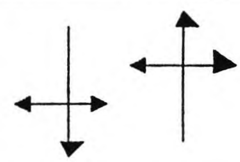
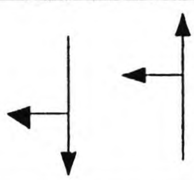
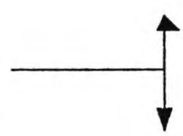
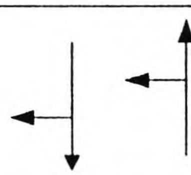
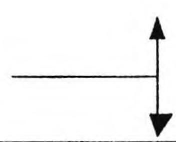
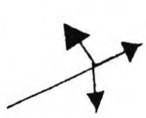
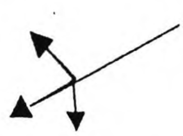
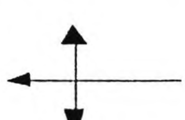
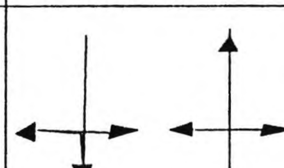
	STAGE		
NODE	1	2	3
16			
15			
14			
13			
12			
11			
10			
22			

Table 3.3

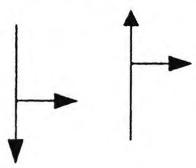
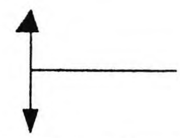
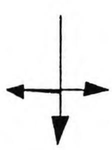
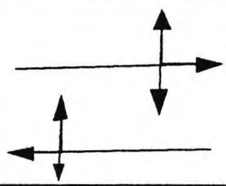
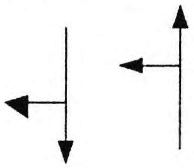
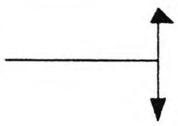
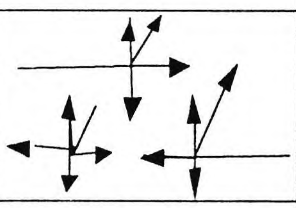
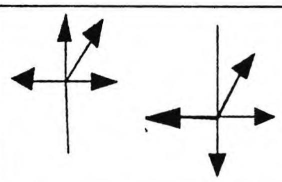
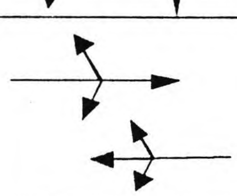
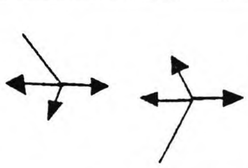
	STAGE		
NODE	1	2	3
21			
19			
18			
17			
23			

Table 3.3 (Continued)

	1	2	3	4	5	6	7	8	9
1	0	210	10	20	0	0	0	0	254
2	0	0	97	919	350	35	0	0	0
3	0	60	0	195	135	200	600	0	0
4	46	200	124	0	40	20	180	39	0
5	0	400	124	55	0	0	40	0	0
6	225	135	0	20	0	0	20	0	0
7	0	65	315	90	15	20	0	0	0
8	0	0	0	123	0	0	0	0	0
9	9	60	0	0	0	0	0	0	0

Table 3.4: Assumed origin-destination matrix of Network 3.

Link	Constant Travel Time (Secs)	Link	Constant Travel Time (Secs)
1 - 10	100	2 - 11	102
3 - 14	100	4 - 16	100
5 - 17	102	6 - 18	102
7 - 19	102	8 - 22	102
9 - 10	102	10 - 1	0
10 - 9	0	10 - 11	32
10 - 22	39	11 - 2	0
11 - 10	102	11 - 12	400
12 - 11	400	12 - 13	5
12 - 22	3	13 - 12	460
13 - 14	344	13 - 21	3
14 - 13	344	14 - 15	565
14 - 23	10	15 - 14	565
15 - 16	546	15 - 17	8
16 - 4	0	14 - 3	546
16 - 15	546	16 - 17	6
17 - 5	0	17 - 15	6
17 - 16	5	17 - 18	34
17 - 23	86	18 - 6	0
18 - 17	34	18 - 19	21
19 - 7	0	19 - 18	20
19 - 20	5	19 - 23	23
20 - 21	9	20 - 23	25
21 - 13	2	21 - 20	5
21 - 22	39	22 - 8	0
22 - 10	39	22 - 12	2
22 - 21	39	23 - 14	2
23 - 17	56	23 - 19	21
23 - 20	23	20 - 19	23

Table 3.5: Assumed constant link travelling cost of Network 3.

3.4 RESULTS AND COMMENTS

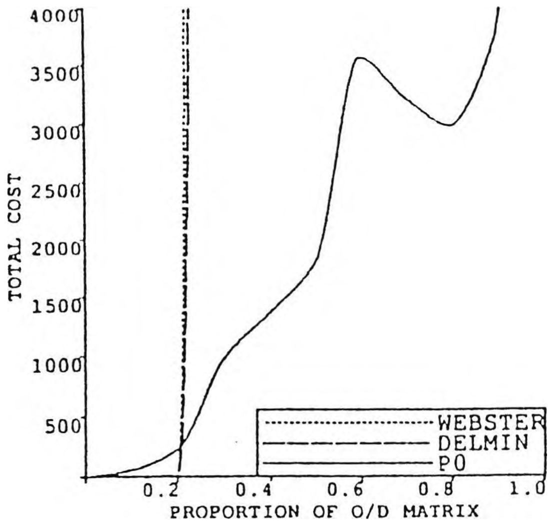
In what follows, the comments apply to the performance of one policy against another. Computational time and number of iterations needed to arrive at the solution are not regarded as important as the performance of each policy in this study, though these might differ largely within and between the three control policies.

Network 1

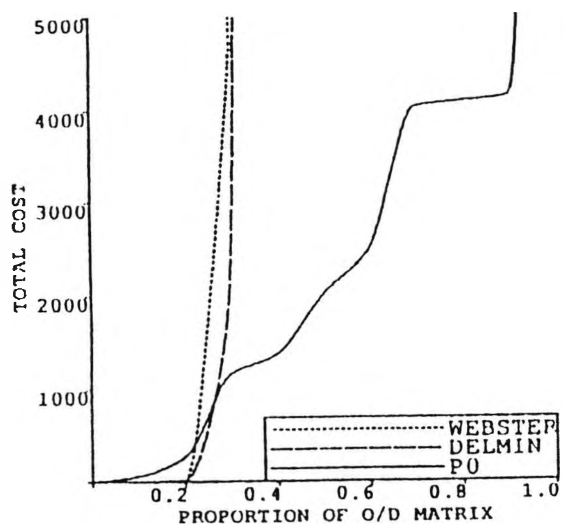
The results of applying the above algorithm to Network 1 are shown in Graphs 3.1-3.5, each graph corresponding to a different value of S_6 as specified in Section 3.3.

Graphs 3.1 and 3.2 show clearly how using different control policies with the algorithm suggested here affect the capacity of the network. In Graphs 3.1 and 3.2, policy P_0 could accommodate a far larger amount of the total demand than either Webster or Delmin Policy. The capacity of the network has almost quadrupled with policy P_0.

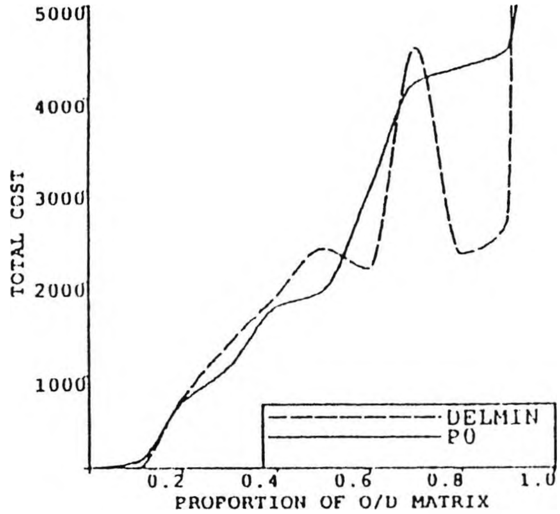
The reason for the small capacity of this network, for cases 1 and 2 and as shown in Graphs 3.1 and 3.2, when either Webster or Delmin policy is applied can be explained in general in the following manner. (The following argument has been given first by Smith (1979a) that shows that in setting traffic lights responsively, Webster's policy might not achieve an equilibrium solution, and then Heydecker (1980) elaborated further on this problem who studied the Jacobian when Delmin policy is used in an assignment process.) Webster and Delmin policies tend to give a greater proportion of green light to the more congested stage so as to reduce delays. The implication of this in an assignment context is that as the flow


4000
3500
3000
2500
2000
1500
1000
500
TOTAL COST
0.2
0.4
0.6
0.8
1.0
PROPORTION OF O/D MATRIX
WEBSTER
DELMIN
PO

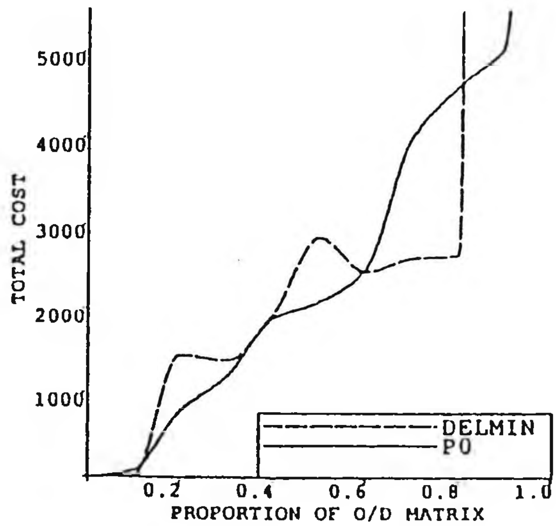
Graph 3.1



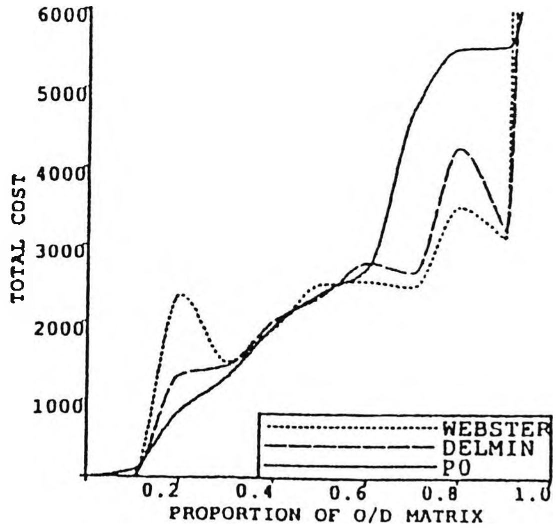
Graph 3.2



Graph 3.3



Graph 3.4



Graph 3.5

of a signal link increases, Webster and Delmin policies allocate more and more green light to this increasingly congested link. Though the link is getting more congested in the long run, these policies on adjusting the settings, reduce delays but encourage more use of the link. But since the capacity of the link is limited by the saturation flow if all the green proportion of the cycle was allocated to that link, then the costs rise steeply. The same statement could also be applied to cases 1 and 2, where the links on the inner routes get increasingly overloaded due to Webster or Delmin policy. In these cases, the central route has a capacity S_4 when either Webster or Delmin policy is used and when the maximum green light is given along this route. Hence, the network capacity becomes approximately equal to S_4.

On the other hand, P_0 tends to penalise the use of links with relatively small capacity, and this pushes drivers away from links with small capacity. But this may cause some concern, particularly where the demand is low and despite the possibility that the link might be able to cope with the flow. Thus, this may result in an unnecessary increase in costs, as in the first part of both Graphs 3.1 and 3.2. Alternatively, at low level of demand, P_0 does not allow a stage to have zero proportion of the total cycle if, for the sake of argument, the minimum green time of a stage could be taken as zero. Or, P_0 always allocates a green time that is greater than the minimum green if the latter is small enough, even though there might not be traffic flow making use of the stage provided. This explains the extra costs if P_0 is used at a low level of demand for the network of Fig. 3.3.

As for Graphs 3.3-3.5, again, at low level of demand, P_0 produces higher costs by giving the stages along the outer routes green time even though this is not needed. This is also observed at high level of demand when Webster and the delay minimising policy give maximum green time allowed for the outer routes, which are used, and minimum green time to the unused inner route: P_0 still gives green time that is greater than the minimum, as above. However, all policies in this case, with S_a, S_b significantly different, provide more or less an equal network capacity.

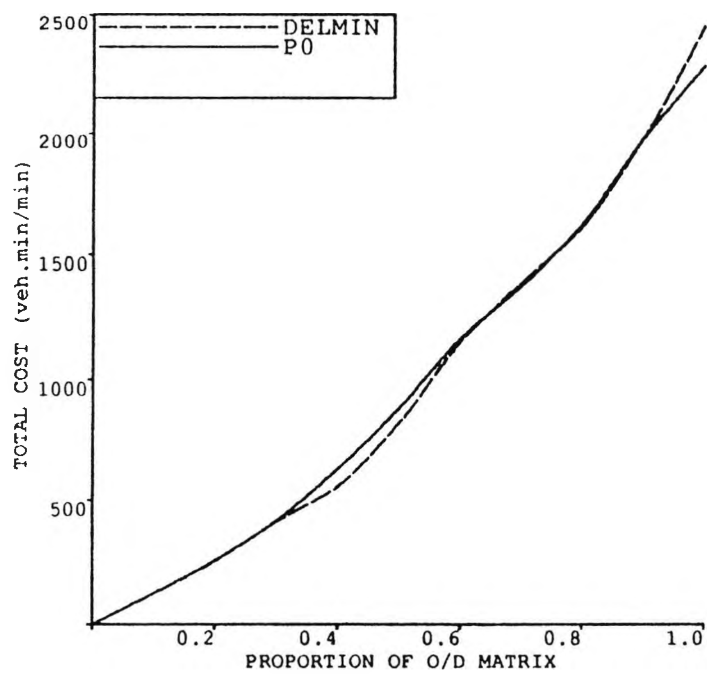
Network 2

The results of this network are given in Graph 3.6. All the policies in this example could accommodate the total demand from all origin to destination points. This is because the location of the traffic signals was delibretly choosen so that no restriction on the capacity of the network is caused.

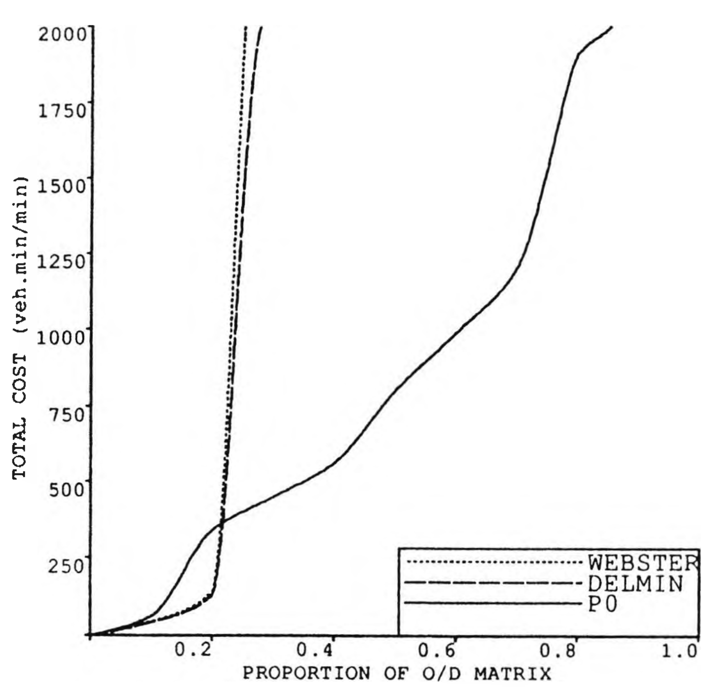
As for the performance of the network under each of the policies, no substantial difference is observed for the levels of demand considered.

Network 3

The results of this network are given in Graph 3.7. Considering the long-run performance of the three control policies, and as Graph 3.7 shows, P_0 markedly outperformed the other two policies by increasing the capacity of the network to almost 3.5 times the capacity obtained on running the signals under Webster or Delmin. Using P_0, about 85 percent of the the total demand was loaded, whereas only about 25 percent were loaded when Webster and Delmin were used.



Graph 3.6



Graph 3.7

4. A METHOD FOR SOLVING THE STATIC NETWORK DESIGN PROBLEM

4.1 INTRODUCTION

As a contribution of this thesis, a method for solving the network design problem, that applies to both, the signal control problem when optimal signal settings are sought, and the link capacity problem as far as optimal link capacities are concerned, is suggested in this chapter. Though the word optimal is used here to signify the design parameters obtained in solving the network design problem, whether it is the signal control problem or the link capacity problem, the design parameters thus obtained are not necessairly globally optimal, as the problem is non-convex, due to constraining the pattern of flow to be in equilibrium, that is in accordance with Wardrop's first principle.

The method suggested follows the outer form (see below § 4.2) given in Marcotte (1983), but differs in the way the main subproblem of Marcotte (see § 2.2.3.2) is solved. The method proposed to solve the main subproblem in Marcotte is for equality constraints rather than for inequality constraints (equilibrium or variational inequality constraints in this setting). Accordingly, the constraints that are allowed to be included in the main subproblem should to be active at the solution. Because of this, Marcotte (1983) ends up solving 2^j secondary subproblems, where j is the number of equilibrium constraints in the main subproblem each time a new constraint is generated if the solution of the main subrpoblem is not an equilibrium pattern of flow, which is rather computationally cumbersome. Moreover, in view of its formulation, the method given in Marcotte (1983) needs enough control or design parameters in order to bring any pattern of flow, while solving the main subproblem, to equilibrium by varying the decision variables. This is not often possible, particularly when some links do not constitute part of a management scheme to build new roads in a road traffic network, or, when optimal signal settings are sought as design parameters, one would not expect a traffic signal at each junction as the formulation in Marcotte (1983) implies, if the question of uniqueness was let alone.

Besides the above complications of the method given in Marcotte (1983), its implementation has been found by its originator to be difficult to attain, for it is needed to solve iteratively as many nonlinear equations as there is in the working set of a secondary subproblem of the main subproblem, using an iterative Newton method to calculate the lagrange multipliers which dualise the equilibrium constraints, for each variation of the flow parameter. Simply, this is computationally intractable.

This chapter is organised such that it presents first in Section 4.2 the method suggested, as applied to the link capacity problem, together with a discussion on how the subproblem, step 2 below, is solved. Then with some changes to the notations, the chapter explains in Section 4.3 how the signal control problem could also be solved using the method (steps 1-5 below) that is applied to the link capacity problem. Follows that, in Section 4.4 results are given for three networks, after having implemented the method in a computer code. Finally, in Section 4.5 a comparison is made between the method suggested here and that due to Marcotte (1983), on the one hand, and that due to Abdulaal et al (1979), on the other hand. Both methods have been explained in Chapter 2.

4.2 A METHOD FOR THE NETWORK DESIGN PROBLEM

The method is intended for continuous variables, that is as far as the link capacity problem is concerned, the capacity of a link is supposed to take on any positive value, unless restricted from above by an upper bound, though practically this may not be attainable. For example, it may not be possible to construct a link whose capacity implies a lane and a half lane width.

The cost function is supposed to be continuous, differentiable and separable; the BPR cost function for the link capacity problem, or Webster's cost function for the signal control problem, are two examples. The demand is assumed to be steady, fixed and given.

<u>Notations</u>

The following notations are needed.

$\mathcal{A}$ = set of arcs which have their end points (i, j) in $\mathcal{N}$.

$\mathcal{N}$ = set of nodes defining the network.

$\mathcal{V}$ = set of feasible link flows satisfying, for each commodity, the following constraints:

$$\sum_{j \in A(i)} v_{ij}^{\alpha} - \sum_{j \in B(i)} v_{ji}^{\alpha} = \begin{cases} 0 \ \forall i \in \mathcal{J} \subset \mathcal{N} \\ m_i^{\alpha} \ \forall i \in \mathcal{O} \subset \mathcal{N} \\ -m_i^{\alpha} \ \forall i \in \mathcal{D} \subset \mathcal{N} \end{cases}$$

and

$$v_{ij}^{\alpha} \geq 0, \ \forall \ (i, j) \text{ in } \mathcal{A}$$

v = vector of link flows, with $v_{ij} = \sum_{\alpha} v_{ij}^{\alpha}$ as an element.

m_i^{α} = demand at origin i for commodity α.

$A(i)$	=	$\{j\,|\,j \in \mathcal{N},\ (i,j) \in \mathcal{A}\}$
$B(i)$	=	$\{j\,|\,j \in \mathcal{N},\ (j,i) \in \mathcal{A}\}$
$\mathcal{I}$	=	set of intermediate nodes
$\mathcal{O}$	=	set of origin nodes
$\mathcal{D}$	=	set of destination nodes
w_{ij}	=	capacity of link (i, j)
$\mathcal{W}$	=	feasible set of capacity of links satisfying $l_{ij} \le w_{ij} \le u_{ij}$
l_{ij}	=	lower capcity limit of link (i, j)
u_{ij}	=	upper capcity limit of link (i, j)
c_{ij}	=	cost function of link (i, j)
β_{ij}	=	factor of link (i, j) that converts link capacity to monetary terms for construction
v_{ij}^{e}	=	flow on link (i, j) due to flow pattern v^{e}
ϵ	=	a given positive stopping criterion
q	=	a constant used to count the number of equilibrium constraints included in problem [4.1(b)]
$S(v,w)$	=	NDP objective function

<u>The Method</u>

The method is as follows:

1- $q=0$, $UB=\infty$.

2- Solve the subproblem:

$$S(v,w) = \min_{v,w} \sum_{ij \in \mathcal{A}} c_{ij}(v_{ij}, w_{ij})v_{ij} + \beta_{ij}w_{ij} \qquad [4.1(a)]$$

$$\text{s.t.} \quad v \in \mathcal{V} \qquad [4.1(b)]$$

$$w \in \mathcal{W} \qquad [4.1(c)]$$

$$\sum_{ij \in \mathcal{A}} c_{ij}(\underline{v}_{ij}, \underline{w}_{ij})(\underline{v}_{ij} - v_{ij}^{e}) \le 0 \qquad (e = 1, \ldots, q) \qquad [4.1(d)]$$

3- Let the solution of step 2 be $(\underline{v},\underline{w})$ and $LB=S(\underline{v},\underline{w})$.

4- Do a minimum path search for fixed $w=\underline{w}$ and denote the resulting pattern of flow v^q.

5- If $\sum_{i,j \in \mathcal{A}} c_{ij}(\underline{v}_{ij},\underline{w}_{ij})(\underline{v}_{ij}-v_{ij}^{q}) \leq \epsilon$, end. Else, increment q by 1 and return to step 2.

As it can be seen, the method is similar to that given in Marcotte (1983), but differs in two senses. In the first sense, it differs substantially in the way the main subproblem, step 2 in the method given in Marcotte (1983), is solved here. An augmented lagrangian approach is adopted in the method given in this chapter, which is explained in the next section. In the second sense, the method differs in the way it is terminated. In Marcotte's method, an upper and lower bounds on costs are used to terminate the algorithm. In the method given here, a check on the last generated equilibrium constraint is used as a termination criterion. Clearly, if the constraint generated in step 4 is satisfied, then the pattern of flow obtained in step 2 is in equilibrium.

4.2.1 Solving the Subproblem at Step 2

Notations

$L(v,w)$	=	augmented lagrangian objective function
ρ_1,ρ_2	=	penalty weights
ρ_i^{max}	=	maximum value of penalty weight
λ_e	=	lagrange multiplier that dualises the equilibrium constraint e of the form:

$$\sum_{i,j\in\mathcal{A}} c_{i,j}(v_{i,j},w_{i,j}).(v_{i,j}-v_{i,j}^{e}) \leq 0$$

$X \quad = \quad \{e|\lambda_e > 0\}$

The Subproblem at Step 2

When q=0, and when the cost function used is the BPR function, the solution of the subproblem at step 2 is the system optimum of the network design problem, for which the method described in Dantzig et al (1979) can be used. But for q values different from zero, an augmented lagrangian, which is due to Pierre et al (1975), is used. With the equilibrium constraints [4.(d)] only added to the objective function of the subproblem, the augmented lagrangian at iteration q takes the form:

$$L(v,w) = \min_{v,w} \sum_{i,j\in\mathcal{A}} c_{i,j}(v_{i,j}, w_{i,j})v_{i,j}+\beta_{i,j}w_{i,j}+$$

$$\sum_{e=1}^{q}\lambda_e\sum_{i,j\in\mathcal{A}} c_{i,j}(v_{i,j}, w_{i,j}).(v_{i,j}-v_{i,j}^{e})+$$

$$\rho_1\sum_{\substack{e=1\\ e\in X}}^{q}\ \sum_{i,j\in\mathcal{A}}[c_{i,j}(v_{i,j}, w_{i,j}).(v_{i,j}-v_{i,j}^{e})]^2 +$$

$$\rho_2\sum_{\substack{e=1\\ e\notin X}}^{q}\frac{1}{2}\sum_{i,j\in\mathcal{A}} c_{i,j}(v_{i,j}, w_{i,j}).(v_{i,j}-v_{i,j}^{e})$$

$$[\sum_{i,j\in\mathcal{A}} c_{i,j}(v_{i,j}, w_{i,j}).(v_{i,j}-v_{i,j}^{e})-|\sum_{i,j\in\mathcal{A}} c_{i,j}(v_{i,j}, w_{i,j}).(v_{i,j}-v_{i,j}^{e})|]$$

s.t. $\quad v\in\mathcal{V}$

$\qquad w\in\mathcal{W}$

Although now the subproblem has been arranged so that the sets $\mathcal{V}$ and $\mathcal{W}$, which are linear, are thus independent, which means that an extreme point $(\mathbf{v},\mathbf{w})$ can be easily determined, the separability property that makes the method given in Dantziq *et al* (1979) efficient with the BPR cost function, is nolonger attained, due to including the equilibrium constraints generated in step 4 into the objective function $S(\mathbf{v},\mathbf{w})$.

Solving an augmented lagrangian is a process which alternates between two phases. Initially, a relatively small postitive value for the penalty weights and zero-values for the lagrange multiplers, are assumed. Then, a minimisation phase is followed, after which, in the second phase, the lagrange multipliers are checked and updated, together with the penalty weights, if optimality is not satisfied. And so on. The rules for updating the multipliers and penalty weights, as in Pierre *et al* (1975), are given first, and then the minimisation method used to solve the augmented lagrangian is described.

4.2.2 Updating Phase

<u>Updating the Lagrange Multipliers</u>

The rules are:

If $e \in X$,

$$\lambda_e^{new} = \begin{cases} 0, \ \text{if } \lambda_e^{old} + 2\rho_1^{old} \sum_{i,j \in \mathcal{A}} c_{i,j}(\underline{v}_{i,j}, \underline{w}_{i,j})(\underline{v}_{i,j} - v_{i,j}{}^e) \leq 0 \\ \lambda_e^{old} + 2\rho_1^{old} \sum_{i,j \in \mathcal{A}} c_{i,j}(\underline{v}_{i,j}, \underline{w}_{i,j})(\underline{v}_{i,j} - v_{i,j}{}^e), \ \text{otherwise} \end{cases}$$

If $e \notin X$,

$$\lambda_e^{new} = \begin{cases} 0, \text{ if } \sum_{i,j \in \mathcal{A}} c_{ij}(\underline{v}_{ij}, \underline{w}_{ij})(\underline{v}_{ij} - v_{ij}^{e}) \leq 0 \\ 2\rho_2^{old} \sum_{i,j \in \mathcal{A}} c_{ij}(\underline{v}_{ij}, \underline{w}_{ij})(\underline{v}_{ij} - v_{ij}^{e}), \text{ otherwise} \end{cases}$$

<u>Updating the Penalty Weights</u>

The penalty weight update rule is:

$$\rho_i^{new} = \begin{cases} \rho_i^{max}, \text{ if } 4\rho_i^{old} \geq \rho_i^{max} \\ 4\rho_i^{old} \end{cases} \qquad (i = 1, 2)$$

4.2.3 Minimisation of L(v,**w**)

Basically, any nonlinear minimisation method for linearly costrained programs may be used, though the problem is non-convex. For example, the Frank-Wolfe (see LeBlanc *et al* (1975)) method, which is a first order method, could be used. But as it is well known, second order methods are a necessity if augmented lagrangian or penalty approaches are used. Here, an augmented lagrangian formulation, due to Pierre *et al* (1975), is favoured on penalty functions which have ill-conditioning problems. As the above augmented lagrangian is constrained by the set $\mathcal{V} \cup \mathcal{W}$, this requires more than two extreme points in the above set so that a second order method could be useful, otherwise, if just two extreme points were used to minimise the augmented lagrangian, then one would be applying a method like the Frank-Wolfe, which, as said above, has convergence problems with augmented lagrangians.

On the basis of that, a restricted simplicial decomposition scheme with r as the maximum number of extreme points, as in Hearn and Lawphongpanich (1987), has been implemented to speed convergence and apply a second order method, such as the modified Newton method as adapted in Goldfarb (1969) for linearly constrained problems, so that the weights which relate the extreme points in $\mathcal{V}$ and $\mathcal{W}$ and minimise the augmented lagrangian objective function can be determined.

Hearn-Lawphongpanich Simiplicial Desompostion

Notations: These are as follows.

r = maximum number of extreme points of a simplex

$\mathcal{H}$ = convex hull of extreme point set ω

γ_i = the weight of extreme pattern i

The Simplicial Decomposition: This is as follows.

1- Let (v_0,w_0) be an initial feasible point in $\mathcal{V}\cup\mathcal{W}$. Set $\omega_0=\{(v_0,w_0)\}$ and t = 0

2- Solve

$$\nabla_v L(v_t,w_t)\cdot\tilde{v}_t = \text{minimise } \{\nabla_v L(v_t,w_t)\cdot\tilde{v} : \tilde{v}\in\mathcal{V}\}$$

$$\nabla_w L(v_t,w_t)\cdot\tilde{w}_t = \text{minimise } \{\nabla_w L(v_t,w_t)\cdot\tilde{w} : \tilde{w}\in\mathcal{W}\}$$

If $\nabla_v L(v_t,w_t)\cdot(\tilde{v}_t - v_t) + \nabla_w L(v_t,w_t)\cdot(\tilde{w}_t - w_t) \geq 0$, stop and (v_t,w_t) is a solution to subproblem at step 2 of § 4.2.

Otherwise,

(i) $|\omega^t| < r$, $\omega^{t+1} = \omega^t \cup \{(\tilde{v}_t,\tilde{w}_t)\}$

(ii) $|\omega^t| = s$, replace any two elements of ω^t with (v_t,w_t) and $(\tilde{v}_t,\tilde{w}_t)$ to obtain ω^{t+1}.

3- Solve:

$$L(v_{t+1},w_{t+1}) = \text{minimise}\{L(x,z) : (x,z)\in\mathcal{H}(\omega^{t+1})\}.$$

Purge ω^{t+1} of all extreme points with zero weight in the expression of (v_{t+1}, w_{t+1}) as a convex combination of the convex hull H, or the convex combination of the extreme points in ω^{t+1}. Set t=t+1 and return to 1.

Step 3 above is for determining a point in the convex hull H, or for determining the weights γ (vector notation) that relate the point (v_{t+1}, w_{t+1}) to the extreme points in ω^{t+1}. If, for instance, the set ω^{t+1} has r elements, then the point (x,z) can be expressed in terms of the extreme points such that:

$$x = (v^1 - v^r)\gamma^1 + \cdots + (v^{r-1} - v^r)\gamma^{r-1} + v^r \quad [4.2(a)]$$

$$z = (w^1 - w^r)\gamma^1 + \cdots + (w^{r-1} - w^r)\gamma^{r-1} + w^r \quad [4.2(b)]$$

The weights, while solving the problem in step 3 of the simplicial decomposition, should satisfy the constraints:

$$\gamma^1 + \cdots + \gamma^r = 1 \quad [4.3(a)]$$

$$\gamma^i \geq 0 \qquad (i = 1, \ldots, r) \quad [4.3(b)]$$

or

$$\gamma^1 + \cdots + \gamma^{r-1} \leq 1 \quad [4.4(a)]$$

$$\gamma^i \geq 0 \qquad (i = 1, \ldots, r-1) \quad [4.4(b)]$$

$$\gamma^r = 1 - \sum_i^{r-1} \gamma^i \qquad \text{(definitional constraint)}$$

Since now any point in $H(\omega^{t+1})$ can be expressed as in [4.2], the problem in step 3 becomes:

$$L(v_{t+1}(\gamma), w_{t+1}(\gamma)) = \text{minimise}\{L(x(\gamma), z(\gamma)) : \gamma \geq 0;\ \gamma^1 + \cdots + \gamma^r = 1\}. \quad [4.5]$$

To solve [4.5], a modified Newton method as given in Goldfarb (1969) was implemented. This is discussed in the next section. Following that a point in relation to solving

$$\nabla_v L(v_t, w_t) \cdot \tilde{v}_t = \text{minimise } (\nabla_v L(v_t, w_t) \cdot \tilde{v} : \tilde{v} \in \Upsilon)$$

is mentioned.

Modified Newton Method

The method outlined in this section is due to Goldfarb (1969), and the reader could refer to that paper for a more detailed discussion. Here, only the main steps of the method are included.

Notations: These are as follows.

$\mathfrak{G}$ = constraint set of γ, ie [4.4(a)] and [4.4(b)]

n_i = the unit normal vector of constraint i in $\mathfrak{G}$, ie $n_i^T n_i = 1$

N^p = $\{n_1, n_2, \ldots, n_i\}$ is an rxp matrix, whose columns are the p unit normals to the p linearly independent hyperplanes

H_i^p = approximated projected hessian of $L(\mathbf{x}(\gamma), \mathbf{z}(\gamma))$ on the flat of p linearly independent hyperplanes in N_p in iteration i

F_i^p = set of those active constraints of $\mathfrak{G}$ at point γ_i

Goldfarb's Method: The steps are:

1- Let γ_0 be an initial feasible point in $\mathfrak{G}$, and H_0^0 be choosen as positive definite matrix. If γ_0 lies in the intersection of p linearly independent hyperplanes of $\mathfrak{G}$, then these constraints should be added to H_0^0 to obtain H_0^p. Determine $\nabla_{\gamma_0} L_0(\mathbf{x}(\gamma_0), \mathbf{z}(\gamma_0))$.

2- In iteration i, γ_i, $\nabla_{\gamma_i} L_i(\mathbf{x}(\gamma_i), \mathbf{z}(\gamma_i))$ and H_i^p are used to determine

$H_i^p \nabla_{\gamma_i} L_i(x(\gamma_i),z(\gamma_i))$ and

$$\tau = (N^{p^T} N^p)^{-1} N^{p^T} \nabla_{\gamma_i} L_i(x(\gamma_i),z(\gamma_i)).$$

If $H_i^p \nabla_{\gamma_i} L_i(x(\gamma_i),z(\gamma_i)) = 0$ and $\tau_j = 1, \ldots, p$, then γ_i is a solution point.

3- If not, then either $\|H_i^p \nabla_{\gamma_i} L(x(\gamma_i),z(\gamma_i))\| > \max(0,\ 1/2\tau^p(b^{pp})^{-1/2})$ or $\|H_i^p \nabla_{\gamma_i} L(x(\gamma_i),z(\gamma_i))\| \leq 1/2\tau^p(b^{pp})^{-1/2}$, where it is assumed that $\tau^p(b^{pp})^{-1/2} \geq \tau^k(b^{kk})^{-1/2}$, $k = 1, \ldots, p-1$, and b^{kk} is the diagonal element of $(N^{p^T}N)^{-1}$. If the former holds, then go to step 4. Else, drop the pth hyperplane from H_i^p to determine H_i^{p-1}, using

$$H_i^{p-1} = H_i^p + \frac{P^{p-1} n^p n^{p^T} P^{p-1}}{n^{p^T} P^{p-1} n^p},$$

where $P^{p-1} = I - N^{p-1}(N^{p-1^T} N^{p-1})^{-1} N^{p-1^T}$.

Let $p = p - 1$ and retrun to step 2.

4- Calculate $s_i = H_i^p \nabla_{\gamma_i} L(x(\gamma_i),z(\gamma_i))$,

$$a_j = \frac{n_j^T \gamma_i - b_j}{n_j^T s_i} \qquad \forall\ j \notin F_i^p,$$

and

$$a_i = \min_j (a_j \geq 0),$$

where b_j is the right hand side constant of constraint j in $\mathcal{G}$. Then determine the step lenght l_i, $0 \leq l_i \leq a_i$, which minimises $L(x(\gamma_i + l_i s_i),z(\gamma_i + l_i s_i))$ with respect to l_i and compute $\nabla_{\gamma_{i+1}} L(x(\gamma_{i+1}),z(\gamma_{i+1}))$.

5- If $l_i = a_i$, then add to F_i^p the constraint p that corresponds to the $\min_j(a_j \geq 0)$ in step 4 to obtain F_{i+1}^{p+1}, and compute

$$H_{i+1}^{p+1} = H_i^p - \frac{H_i^p n^p n^{p^T} H^p}{n^{p^T} H^p n^p}$$

Increment p by 1, i by 1, and return to step 2.

6- Otherwise, determine $\sigma_i = \gamma_{i+1} - \gamma_i$, $u_i = \nabla_{\gamma_{i+1}} L(x(\gamma_{i+1}), z(\gamma_{i+1})) - \nabla_{\gamma_i} L(x(\gamma_i), z(\gamma_i))$, and update H_i^p using

$$H_{i+1}^p = H_i^p + \frac{\sigma_i \sigma_i^T}{\sigma_i^T u_i} - \frac{H_i^p u_i u_i^T H^p}{u_i^T H^p u_i}$$

Set $i = i + 1$, and go to step 2.

In step 2 and 3, $(N^{p^T} N^p)^{-1}$ and $(N^{p-1^T} N^{p-1})^{-1}$, do not have to be abtained using matrix inversion, as Goldfarb provides recursion formulae to obtain these, in addition to determining N^{p-1} and P^{p-1} whenever a plane is dropped, or, N^{p+1} and P^{p+1}, whenever a plane is added.

Minimising $\nabla_v L(v_t, w_t) \cdot \tilde{v}$

Minimisation of $\nabla_v L(v_t, w_t) \cdot \tilde{v}$ is an exereme pattern of flow in Ψ, which is a similar problem to that of the pure equilibrium static assignment problem (see LeBlanc *et al* (1975)), and for which a minimum path search, such as Dijkstra's (1959) minimum path algorithm, could be used. For the pure assignment problem, the cost function is continuously increasing and the gradient in terms of the flow variable implies positive costs, which is a requirement for using the minimum path search of Dijkstra. But, this no longer holds in minimising $\nabla_v L(v_t, w_t) \cdot \tilde{v}$, as the gradient $\nabla_v L(v_t, w_t)$ in terms of the flow variable might at some points be negative, which gives rise to negative circuits or loops if the minimum path algorithm of Dijkstra (1959) is implemented. The implication of negative circuits is simply that minimising $\nabla_v L(v_t, w_t) \cdot \tilde{v}$ becomes an unbounded problem if it is retained in its present form, or that drivers could reduce their costs infinitely by travelling into a loop and the flow for some origin-destination pair on some links becomes infinite and greater than the demand value for that particular origin-destination pair. To avoid this problem, a simple and natural constraint on the flow variable on each link, for each commodity or each origin-destination pair, is that the flow on any link, v_{ij}^{α}, should be no greater than the demand of commodity α. Thus, looping is avoided.

Now solving

$$\nabla_v L(v_t,w_t)\cdot\tilde{v}_t = \min_{v} \nabla_v L(v_t,w_t)\cdot\tilde{v}$$

$$\text{subject to} \quad \tilde{v} \in \Psi$$

becomes

$$\nabla_v L(v_t,w_t)\cdot\tilde{v}_t = \min_{v} \nabla_v L(v_t,w_t)\cdot\tilde{v}$$

$$\text{subject to } \tilde{v} \in \Psi$$
$$\tilde{v}_{ij}^{\alpha} \leq m_I^{\alpha} \qquad \forall\, I \in \mathcal{O}.$$

This problem is a network flow problem with an upper bound on each flow variable, for which an efficient minimum path algorithm is given in Bazaraa et al (1977, P. 420). Another method, which is also given in Bazaraa et al (1977, P. 516), could be used. This method attempts to remove the negative costs prior to using Dijkstra's algorithm. But it should be borne in mind that this method is not always capable of eliminating the negative costs. In implementing the method suggested for solving the network design problem, the method which attempts to eliminate the negative costs was used and no problem did occur.

4.3 THE SIGNAL CONTROL PROBLEM

As mentioned earlier, the method suggested above for solving the link capacity problem is also suggested for solving the signal control problem. This requires only replacing the set $\mathcal{W}$ with $\mathcal{J}$ and $\mathbf{w}$ with Λ in [4.1] and all that follows after [4.1], down to this section, while leaving out the term $\beta_{ij}w_{ij}$, $\forall(i,j)\in\mathcal{A}$, whereever it is added. $\mathcal{J}$ and Λ are defined below, and the cost function used for this problem, follows that.

To solve problem [4.1] when $q=0$, the iterative loading procedure discussed in Chapter 3 is needed to find an initial feasible point, for both signal settings and flow variables, if Webster's cost function is used. But in this case, in step 2 of the method given in Chapter 2, the path which has the least marginal costs is determined, instead of the path which has the least user costs.

<u>Notations</u>

a_{ij} = uncongested travelling cost

b_{ij} = congestion coefficient of the BPR cost function

X_{ij} = degree of saturation, i.e., $\dfrac{v_{ij}}{\Lambda_{ij}s_{ij}}$

$\mathcal{J}$ = the set of signalised nodes in $\mathcal{N}$

k_j = number of stages at traffic light j

$\gamma_{l,j}$ = the proportion of cycle that is effectively green for stage l

$g_{l,h}^{min}$ = the minimum green time for stage l

L_j = total lost time per cycle

$$a_{l,ij} = \begin{cases} 1 & \text{if stream } (i,j) \text{ belongs to stage } l \\ 0 & \text{otherwise} \end{cases}$$

A_j = the stage matrix whose elements are $a_{l,i,j}$'s

c_j = the cycle time of light j

c_j^{min} = the minimum cycle time

c_j^{max} = the maximum cycle time

$\Lambda_{i,j}$ = effective green time of stream (i,j)

$\mathfrak{F}$ = set of feasible settings Λ satisfying at each traffic light, $j\in\mathfrak{J}$, the constraints:

$$\sum_{l=1}^{k}\gamma_{l,h} = 1 - L_j/c_j \qquad [2.9d]$$

$$\gamma_{l,j} \geq g_{l,j}^{min}/c_j \qquad [2.9b]$$

$$v_{i,j} \leq \Lambda_{i,j}w_{i,j} \qquad [2.9e]$$

$$c_j^{min} \leq c_j \leq c_j^{max} \qquad [2.9c]$$

User Cost Function

The user cost function used for this problem, if node $j\in\mathfrak{J}\subset\mathcal{N}$, is the Webster's cost function which is of the form:

$$c_{i,j}(v_{i,j},\Lambda_{i,j}) = a_{i,j}+\frac{9}{10}\left(\frac{c_j(1-\Lambda_{i,j})^2}{2(1-\Lambda_{i,j}\ X_{i,j})}+\frac{X_{i,j}^2}{2v_{i,j}(1-X_{i,j})}\right).$$

If $j \notin\mathfrak{J}$, then the BPR, of the form

$$c_{i,j}(v_{i,j}) = a_{i,j}+b_{i,j}\left(\frac{v_{i,j}}{w_{i,j}}\right)^4,$$

is used

4.4 NETWORKS AND RESLUTS

The method suggested in this chapter for dealing with the network design problem was put into practice by applying it to three test networks. The application was confined only to the problem of determining the optimal capacity of links, in each of the three tests.

Network 1

The first network is as in Figure 4.1. The network properties are given in Table 4.1. The demand for each of the origin-destination pairs 1-2, 1-4, 3-2 and 3-4, was 1.2, .54, 0.6 and 0.9 flow units, respectively. Only two links in this example were assumed to have variable capacity, link 1-2 and 3-4, with the values of β_{ij}'s as in Table 4.1. The results are given in Table 4.2 and the total costs on solving for the system optimum of the NDP, the user equilibrium for fixed link capacity determined from the system optimum, and the NDP solution were 425.8, 443.75 and 431.32 cost units, respectivley.

For this network two equilibrium constraints were generated, and the results obtained were checked for validity using a self-written program of the method described in Abdulaal *et al* (1979).

Network 2

This is a 3-link example and as in Fig 4.2. The link properties are given in Table 4.3 and the demand for the origin-destination pair 1-2 was taken as 1 flow unit. Only the capacity of link 2 was assumed to be required. The results are given in Table 4.4. The total costs on solving for the system optimum of the NDP, the user equilibrium for fixed link capacity determined from the system optimum, and the NDP solution were 11863.61,

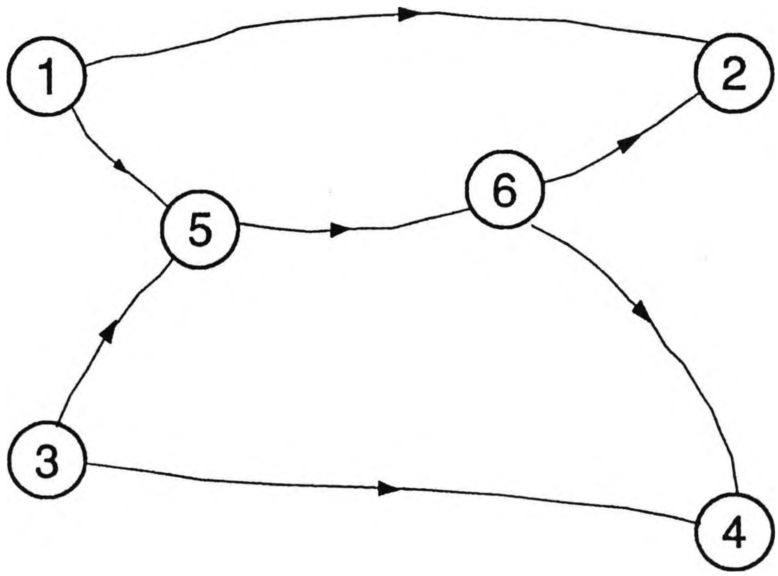


Figure 4.1

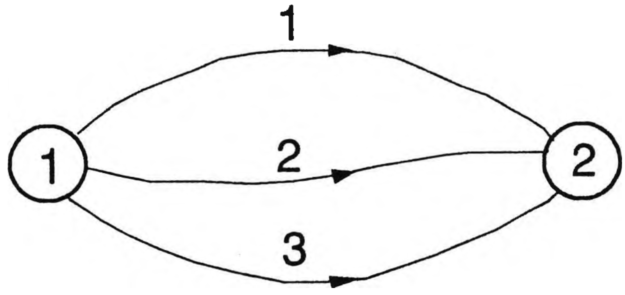


Figure 4.2

Link	a_{ij}	b_{ij}	l_{ij}	u_{ij}	β_{ij}
1-2	140	0	0.1	-	150
1-5	25	0	1	1	-
3-4	30	0	0.1	-	150
3-5	150	0	1	1	-
5-6	35	0	1	1	-
6-2	30	0	1	1	-
6-4	20	0	1	1	-

Table 4.1

Link	Cost	Flow	Capacity
1-2	73.3	1.20	1.07
1-5	25.3	0.54	1.00
3-4	112.	0.15	0.12
3-5	31.9	0.81	1.00
5-6	52.2	1.35	1.00
6-2	30.0	0.06	1.00
6-4	28.2	1.29	1.00

Table 4.2

Link	a_i	b_i	l_i	u_i	β_i
1	25	4000	0.2	0.2	-
2	5025	3000	0	-	12000
3	30	3500	0.3	0.3	-

Table 4.3

Link	Cost	Flow	Capacity
1	6960	0.23	0.20
2	6960	0.41	0.46
3	6960	0.36	0.30

Table 4.4

a_{ij}	b_{ij}	l_{ij}	u_{ij}	β_{ij}
60	9	0.01	0.25	100

Table 4.5

12540.4 and 12512.11 cost units, respectively. Two equilibrium constraints were generated for this problem. The solution using the method of Abdulaal et al (1979) confirmed that the results were correct. It is worth mentioning that the method due to Marcotte (1983) cannot be applied to this example, as it is not always possible with just one link considered for construction to bring any pattern of flow to equilibrium by varying the capacity of the link.

Network 3

This is shown in Figure 3.4, and it is a fairly large network. For this network, all links were considered for construction. The characteristics of each link are given in Table 4.5.

The demand between all origin-destination pairs was taken as 0.08 flow unit. The total costs on solving for the system optimum of the NDP, the user equilibrium for fixed link capacity determined from the system optimum, and the NDP solution were 15255.12, 15262.85 and 15263.87 cost units, respectively. The results are given in Table 4.6. Eight equilibrium constraints were generated to arrive at the solution. The method of Abdulaal et al (1979) could not be applied here due to excessively high computational cpu-time.

4.5 COMPARISON

Contrasting the method of the subproblem proposed here with that in Marcotte (1983), in the former there is no need to solve 2^{T} times a subproblem, because augmented lagrangian methods have a self-adjustment mechanism to deal with inequalities which are not active at the solution. Besides, the problem of sufficient control does not exist in this approach,

Link	Cost	Flow	Capacity	Link	Cost	Flow	Capacity
1-10	85.01	0.32	0.25	10-1	85.01	0.32	0.25
1-11	82.86	0.23	0.18	11-1	82.86	0.23	0.18
1-12	116.7	0.40	0.25	12-1	116.7	0.40	0.25
2-13	446.5	0.64	0.25	6-18	75.18	0.18	0.16
13-2	446.5	0.64	0.25	20-11	152.0	0.45	0.25
3-15	446.5	0.64	0.25	18-6	75.18	0.18	0.16
4-16	446.5	0.64	0.25	6-19	164.7	0.46	0.25
5-17	446.5	0.64	0.25	19-6	164.7	0.46	0.25
7-21	446.5	0.64	0.25	10-11	112.7	0.39	0.25
8-23	446.5	0.64	0.25	11-10	112.7	0.39	0.25
9-24	446.5	0.64	0.25	10-24	651.0	0.71	0.25
16-4	446.5	0.64	0.25	24-10	651.0	0.71	0.25
15-3	446.5	0.64	0.25	11-12	89.08	0.34	0.25
17-5	446.5	0.64	0.25	12-11	89.08	0.34	0.25
21-7	446.5	0.64	0.25	11-20	152.0	0.45	0.25
23-8	446.5	0.64	0.25	12-13	719.1	0.73	0.25
24-9	446.5	0.64	0.25	13-12	719.1	0.73	0.25
13-14	719.1	0.73	0.25	14-13	719.1	0.73	0.25
14-19	84.98	0.32	0.25	19-14	84.98	0.32	0.25
14-15	728.9	0.73	0.25	15-14	728.9	0.73	0.25
15-16	160.0	0.46	0.25	16-15	160.0	0.46	0.25
17-18	632.2	0.71	0.25	18-17	632.2	0.71	0.25
18-19	230.6	0.52	0.25	19-18	230.6	0.52	0.25
19-20	1371.	0.87	0.25	20-19	1371.	0.87	0.25
20-21	708.3	0.73	0.25	21-20	708.3	0.73	0.25
21-22	132.7	0.42	0.25	22-21	132.7	0.42	0.25
22-24	74.81	0.11	0.10	24-22	74.81	0.11	0.10
22-23	81.12	0.31	0.25	23-22	81.12	0.31	0.25
23-24	87.78	0.33	0.25	24-23	87.78	0.33	0.25
17-16	111.1	0.39	0.25	16-17	111.1	0.39	0.25

Table 4.6

which implies that more general cases could be studied. For instance, with the formulation given here, test 3 above could be solved.

The difference also between this approach and that of Tan *et al* (1979) is that the formulation here is link-space, whereas in Tan's it is path-space. Naturally, one expects to solve larger networks in our formulation, though the problem of the number of equilibrium constraints might be a difficulty.

Also, this formulation can deal with a relatively larger number of link width variables than the method of Abdulaal *et al* (1979). This is obvious in test 3, above.

5. A DYNAMIC TRAFFIC ASSIGNMENT MODEL

5.1 INTRODUCTION

The dynamic traffic assignment problem to determine time-varying link flows in a congested road network where drivers are assumed to be cooperative in minimising total transportation costs is an essential tool in modelling peak periods for three reasons. It 1) indicates the best network performance when drivers are guided by a central controller, given that guidance is accepted, 2) it could be used as a planning tool in a traffic management study if 1) was achievable, and 3) it is useful for road pricing.

Yet, as mentioned in Chapter 1, most of the work done up to date on this problem has been confined to a single commodity network. Amongst the authors who addressed the dynamic traffic assignment problem and whom we mentioned in Chapter 1 are D'Ans and Gazis (1976) in the earliest substantial efforts on this problem, Merchant and Nemhauser (1978), Carey (1987) and Zawack and Thompson (1987).

In contrast, in this Chapter we describe a model that can be applied to multi-commodity networks with a general topology. The idea is simple and based on the local marginal cost for each link.

Our assumptions are as follows. We shall be considering that travelling costs amount to travelling time that can be regarded as composed of running time (a constant reflecting the free-flow speed) and queuing delays. We will also assume for simplicity that the model has a vertical queuing property, so that blocking back is left out of consideration. Further, our approach of queue modelling is deterministic. It assumes that queues form on a link due to excess input flow into the link as compared to its service rate, which is determined by the capacity of a bottleneck located along the link or, perhaps, as it is common, situated at the exit of the link. Figure 5.1, which shows the relation between the arrival rate and the service rate, is an example of our queuing model. In this figure, the input flow rate, v, exceeds the service rate w of the bottleneck. The curve V(t) and W(t) represent the cummulative arrivals and cummulative departures, respectively, as a function of time, and q and d are the queue length and queuing delay at time t, repectively. V(t) is related to the arrival rate v(t) by

$$V(t) = \int_0^t v(\tau)d\tau,$$

while W(t) is related to w(t) by

$$W(t) = \int_0^t w(\tau)d\tau.$$

The area confined between the two curves, V and W in Figure 5.1, represents the total queuing delays of all drivers entering the link.

The rest of the chapter is organised as follows. In the next section, we mention the difficulties with the methods adopted by the above authors. In fact it is these difficulties which motivated our work and they distinguish the model proposed here, as compared to others. Then, in Section 5.3, firstly, the marginal cost for each link, for the queuing model of Figure 5.1, is defined, and, subsequently, the algorithm is sketched and followed by an explanation regarding its steps, in addition to further relevant points. Section 5.4 provides numerical results on two network tests, obtained by implementing the algorithm presented in Section 5.3. Section 5.5 indicates

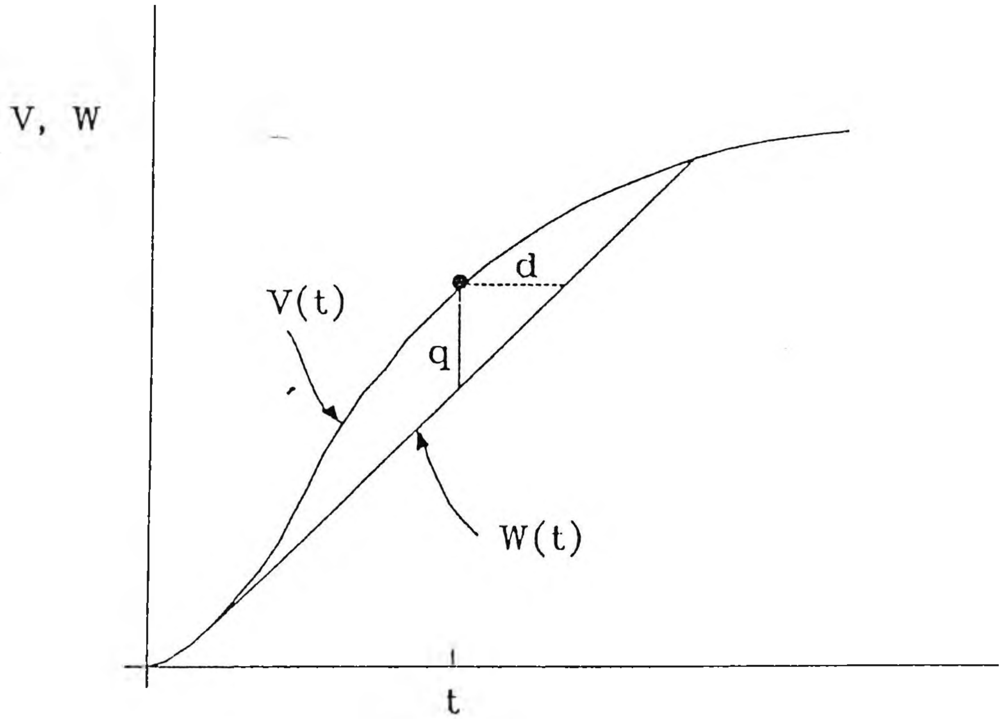


Figure 5.1

some limitations of the method presented here, which call for further research work.

5.2 DIFFICULTIES WITH CURRENT METHODS

All the authors whom we mentioned above dealt mainly with a single commodity network, due to the difficulty of modelling "first-in, first-out" queue discipline in a mutli-commodity case.

As mentioned in Chapter 2, D'Ans and Gazis (1976), in an attempt to resolve this difficulty and extend their model to a multi-commodity network, suggested constraining the flow of each commodity exiting from each link to be proportional to the mix of the commodities in the queue itself. Seemingly independently, these constraints re-appeared again in Carey (1987). The problem with these constraints is that they are nonlinear, non-convex, and non-network constraints. Such properties make their use impractical and compuationally cumbersome.

In addition to the problem of "first-in, first-out", another difficulty confronting these authors is the problem of non-zero flow control variables in a multi-commodity network. Carey (1987) highlighted this further problem, in modifying the single commodity model given in Merchant and Nemhauser (1978), to have a convex set of flow constarints.

The problem of non-zero flow control variables in a multi-commodity network arise because it may be beneficial sometimes to hold traffic from a certain commodity back, while traffic from a different commodity could proceed. To illustrate, we consider the following example.

<u>An example of non-zero flow control variables in a multi-commodity network</u>

Consider the network shown in Figure 5.2. In this network, three commodities can be observed, commodity 1 corresponding to the origin-destination pair O1-D1, commodity 2 to O2-D2, and commodity 3 to O3-D3. For simplicity, travelling time on links is taken as zero. Further, the network is assumed to have two bottlenecks, the first is located at the exit of link 1-2, denoted as B1, and the second at the exit of link 3-4, denoted as B2. Each bottleneck has a capacity of 10 veh/sec. The demand from O1 and O2 starts at time t = 0 seconds, at a rate of 10 veh/sec, for a period of 60 seconds, while the demand from O3 starts later at t = 60 seconds, at the same rate as O1 and O2, and for 60 seconds as well.

Now, we consider two cases. In the first case, if traffic of commodity 2 was held back at node 1, by means perhaps of a traffic controller which acts as a flow control variable in this case, then traffic from commodity 1 could proceed through the first and second bottleneck, without having to queue at either bottleneck. The total costs in this case are just queuing delays of value 36000 veh-sec, due to holding the traffic from commodity 2 at node 1. Note that traffic from commodity 3 in this case passes through the second bottleneck without having to queue, as all the traffic from commodity 1 would have arrived at its destination by the time traffic from commodity 3 starts entering the bottleneck. On the other hand, in the second case, if traffic from commodity 2 was allowed to merge with traffic from commodity 1, then queuing delays develop, and precisely half the demand from commodity 1 emerging from the first bottleneck would merge in this case with flow from commodity 3, thus increasing the queuing costs from 36000 to 49500 veh-sec.

Indeed, non-zero flow control variables may arise also in a single

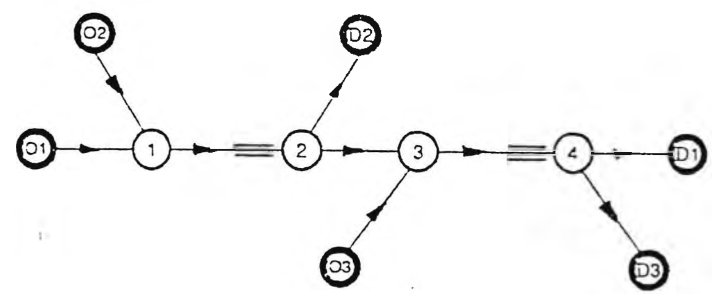


Figure 5.2

commodity network, as it is shown in Carey (1987), where, as well as in Merchant and Nemhauser (1978), a flow-dependent exit function to "explicitly treat congestion", together with a general objective function, are employed. As a special case, the general objective function employed in Carey (1987) and Merchant and Nemhauser (1978) includes the objective function employed in this paper, in D'Ans and Gazis (1976) and in Zawack and Thompson (1987). In their study, Merchant and Nemhauser give assumptions on the objective function that guarantee the optimality of the solution of their model, which the objective function of this paper, of D'Ans and Gazis (1976), and Zawack and Thompson (1987), all satisfy. These assumptions, amongst others on the exit function, arose again in the work of Carey (1987), where it is shown that the optimality conditions set out by Merchant and Nemhauser for their model, correspond to zero flow control variables of the Carey (1987) model. Because it is felt that the special case indicated in Merchant and Nemhauser (1978) and Carey (1987) that happens to coincide with the queuing model of Figure 5.1 is a reasonable approach to modelling traffic congestion, then it could be said that flow control variables should pose no difficulty for a single commodity network in this case.

5.3 DYNAMIC TRAFFIC ASSIGNMENT ALGORITHM

The model described is a combination of simulation and a vehicle routing algorithm. It requires a route storage for each vehicle or each packet of vehicles, if packets were used to reduce computational time and computer core, by considering that delays experienced by a certain packet as common to all the vehicles within the packet. A similar concept of packet flow is followed in CONTRAM (Leonard *et al* (1987)), and the model described here can be easily adapted in CONTRAM.

With the route of each packet specified, the model simulates the "first-in, first-out" queue discipline by mixing the packets, instant-by-instant. The route calculation of each packet is performed with reference to the local marginal costs of each link, but the time of entry to and from each link, represent the times when the packets actually enter and leave each link.

The link marginal travel cost

The marginal costs to a driver travelling on a link can be viewed as: 1) a flow independent or uncongested running cost incurred before arriving at the bottleneck (a constant reflecting the free-flow speed), and 2) a user queuing cost term, quantified as $d=q/w$, and 3) a term corresponding to

$$\mu = \frac{1}{w}\int_{t}^{T} v(\tau)d\tau,$$

which could be defined as the additional delays experienced by drivers arriving between time t and T due to the packet arriving at time t and which should consider this as social costs. T is the time at which the cummulative departure and arrival curves intersect.

To explain how this expression is arrived at, Figure 5.3 shows at time t the arrival of a driver or a unit flow, or, a packet flow, whose presence induce additional costs to the overall system by an amount that is equivalent to the solid area in Figure 5.1. Now, if the flow arriving at time t is a unit flow, then the solid area will be equal to the horizontal distance m. Or, since the algorithm deals with assignment of packets, which are kept of constant size throughout the assignment process, and each packet is assigned to only a single route connecting the corresponding origin-destination pair, then, in calculating the minimum marginal cost path, the size of the packet becomes superfluous, as the thickness of the solid

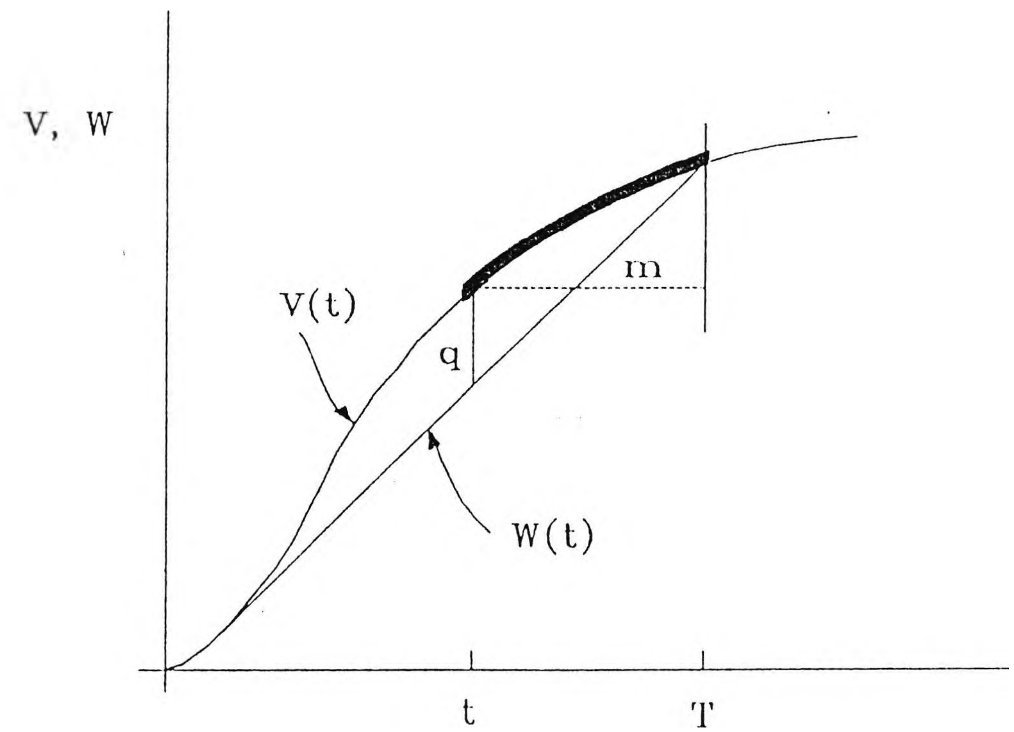


Figure 5.3

line in Figure 5.3, corresponding to the size of the packet, is common to all links when the minimum path is calculated. Accordingly, only the value of m is needed. Hence any packet joining at time t the back of a queue, should consider the value of m, computed as the difference between T and t, rather than d, in addition to the uncongested travel time. In fact, part of m is the user cost, d. Thus, μ obtains.

<u>The algorithm</u>

Essential to the algorithm before the assignment procedure is started is that the demand, assumed initially fixed and given for each origin-destination pair, should be substituted by an appropriate number of flow packets that are ready to leave each origin point in the network at times corresponding to the middle of the time interval matching the size and position of each packet in the original demand profile. The packet size is subject to descretion or the resolution of the packet size required. Obviously, more accuracy could be gained if the packet size is fairly small, but this may require more computational time in central processing and more storage core to store the packets' routes.

<u>Notations</u>: The following notations are needed.

p = the number of packets

k = packet number

i = iteration counter

ρ_k^i = route of packet k in iteration i

C^i = total costs in iteration i

c_k^i = total costs due to routing packet k in iteration i

c_0^i = total initial costs before assignment of packets.

Steps: The steps of the algorithm are as follows.

1) i=0

2) Assign all packets to their minimum marginal journey cost routes while taking into account "first-in, first-out" discipline by mixing the packet inflows, instant-by-instant. Let the route of each packet k be ρ_k^i.

3) Calculate total link costs, C^i, by adding travelling costs of all packets together with queuing costs abtained from profiles similar to Figure 1, for each link. Let $c_0^i = C^i$.

4) Let k=1. For each packet:

   a) Subtract the flow of packet k from each link on the route ρ_k^i and in the corresponding time slice so that "first-in, first-out" discipline is attained.

   b) Determine new minimum marginal journey cost route ρ_k^{i+1}, on the basis of the values of m obtained from profiles similar to Figure 1 for each link.

   c) Assign packet k to its new route ρ_k^{i+1} while accounting for "first-in, first-out".

   d) Calculate total costs c_k^i as in 2). If $c_k^i \leq c_{k-1}^i$, then the new path of packet k is favourable and the old route ρ_k^i is replaced by ρ_k^{i+1}. Otherwise, $c_k^i = c_{k-1}^i$ and ρ_k^i is left unchanged.

   e) If k=p, then go to step 5). Otherwise, increment k by 1 and return to 4-a).

5) If $c_p^i \leq C^i$ then increment i by 1, let $C^i = c_p^i$ and $c_0^i = c_p^i$, and return to step 3). Otherwise, the algorithm is terminated.

The algorithm is basically, in outer structure, similar to CONTRAM, but here drivers are routed with refernce to their local marginal costs rather than perceived costs. In other words, each packet of flow is penalised by an additional cost that is equivalent to the costs incurred by other drivers, arriving later and using the same link, due to the presence of the packet that is being routed.

Because queues are formed when input flows exceed some capacity limit for a period of time, the model seeks essentially to keep queuing delays and the period during which queues occur to a minimum, as long as travelling costs on longer routes is still beneficial, as compared to the total local marginal costs of all links, or the sum of m's, along any other route.

The convergence of the algorithm is trivial: it stops when no further cost reduction is possible. The convergence here can be guaranteed, as there is an obvious objective function to minimise, unlike the related equilibrium assignment problem (see Smith and Ghali (1990) for which there is no apparent objective function.

As compared to the methods mentioned in the introduction, holding back of traffic does not occur in the method described here, in addition to overcoming the problems associated with the first-in, first-out discipline. The reason no-holding back of traffic arise in our model, can be attributed to the way the packets are routed. While each packet is routed down, so as to determine its minimum marginal cost path, the packet exit time from a queue is considered as the time it joins the queue plus the time needed to dissipate the queue length in front of the packet, whereby the packet is then input into the next relevant downstream link, at that exit time. To say it differently, no packet is allowed to take more, or less, time than the time required to exit from a queue; this is equivalent to d in Figure 5.1. For instance, since holding back of traffic in this model is not possible, applying steps 1-5 above to the example of Sect. 5.2, should result in total costs of 49500 veh-sec.

5.4 TEST NETWORKS

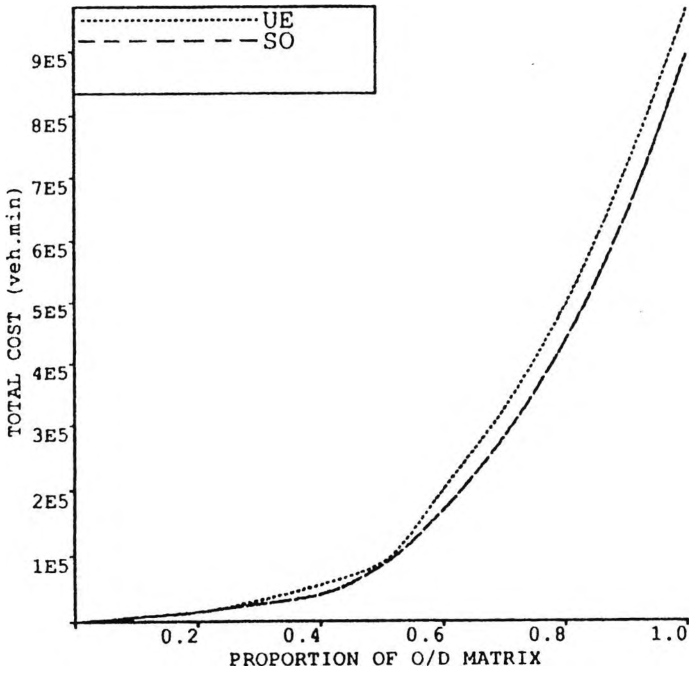
To provide some numerical results, two networks were used. For each network, we compare the network performance due to the method suggested here against that of the user equilibrium pattern of flow due to CONTRAM, for different levels of congestion, so as to study the difference between each as congestion becomes more severe. By factoring the demand of each of the two networks by a value p, where p was initially taken as 0.1, and then incremented by 0.1, up to 1.0, ten different levels of demand were considered, and a smooth curve was plotted between the corresponding network performance points.

Curve SO in the Graphs below denotes the network performance due the method given in this paper, while UE is due to the user equilibrium of CONTRAM.

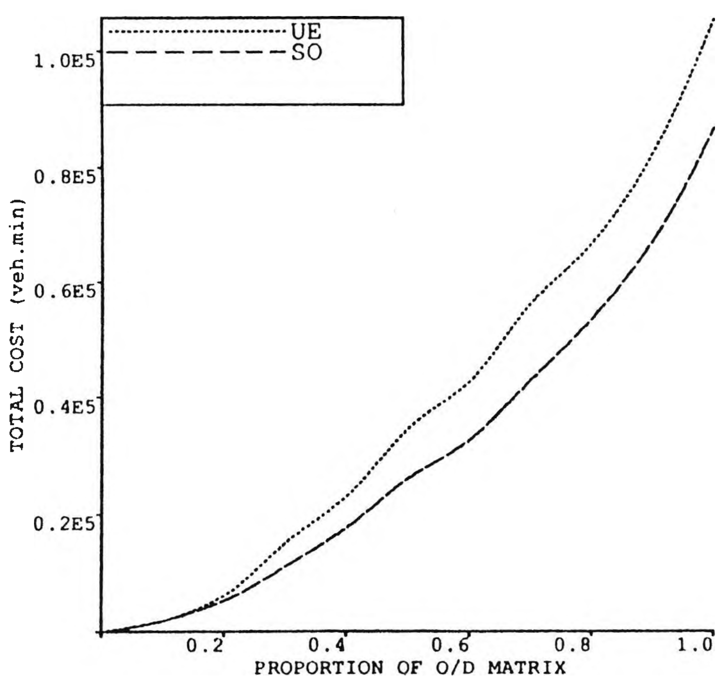
Network 1

The first network is shown in Figure 3.4 of Chapter 3. The link constant travelling time is given in Table 3.2. The demand was taken as 400 veh/hr for a period of 1 hour. Apart from links 13-14, 19-14, 15-14, 1-11, 12-11, 10-11 and 20-11, which each had a capacity of 1000 veh/hr, the capacity of every other link was taken as 2000 veh/hr.

The results given in Graph 5.1 show clearly that there are benefits of routing vehicles according to our method rather than the user equilibrium



Graph 5.1



Graph 5.2

for the levels of congestion specified.

Network 2

This network is shown in Figure 3.5 of Chapter 3. The capacity of each link is given in Table 5.1, and the link constant travel time of each link in Table 3.5. The demand for each origin-destination pair is as in Table 3.4.

Again, Graph 5.2 shows that the performance of the network due to our method is better than the performance due the user equilibrium of CONTRAM.

5.5 FURTHER RESEARCH

The ability of the method to deal with many origin-destination and many bottleneck networks in a dynamic context has been demonstrated in the network tests provided. The method, on the other hand, has the following limitations:

1- The model considers only the local marginal costs of each bottleneck while routing each packet. In the steady state, routing vehicles along the routes which have least link marginal costs, determines a least or the least costly pattern of flow if the cost function is convex (see Dafermos (1969).) Regrettably, this feature does not carry over to the dynamic state. To show that, we consider the example of Section 2 again. But in this instance, to allow for route choice we connect O1 to D1 by link O2-D2, as shown in Figure 5.4. Travelling time on link O2-D2 is assumed to be equal to Δ seconds. Now, the origin-destination pair O2-D2 has two paths, O2-D2 and O1-1-2-D2.

Link	Link Capacity (Veh/Hr)	Link	Link Capacity (Veh/Hr)
1 - 10	1181	2 - 11	920
3 - 14	789	4 - 16	1444
5 - 17	1087	6 - 18	905
7 - 19	1087	8 - 22	740
9 - 10	920	10 - 1	3750
10 - 9	10000	10 - 11	187
10 - 22	125	11 - 2	10000
11 - 10	920	11 - 12	1627
12 - 11	1037	12 - 13	229
12 - 22	787	13 - 12	1445
13 - 14	3775	13 - 21	1169
14 - 13	1627	14 - 15	1902
14 - 23	1168	15 - 14	2715
15 - 16	1627	15 - 17	1455
16 - 4	10000	16 - 15	1627
16 - 17	229	14 - 3	10000
17 - 5	10000	17 - 15	1169
17 - 16	172	17 - 18	229
17 - 23	1329	18 - 6	10000
18 - 17	229	18 - 19	125
19 - 7	10000	19 - 18	229
19 - 20	500	19 - 23	1168
20 - 21	252	20 - 23	229
21 - 13	1329	21 - 20	2550
21 - 22	125	22 - 8	10000
22 - 10	187	22 - 12	1444
22 - 21	229	23 - 14	638
23 - 17	1145	23 - 19	1181
23 - 20	1975	20 - 19	125

Table 5.1: Assumed capacity of links of Network 2.

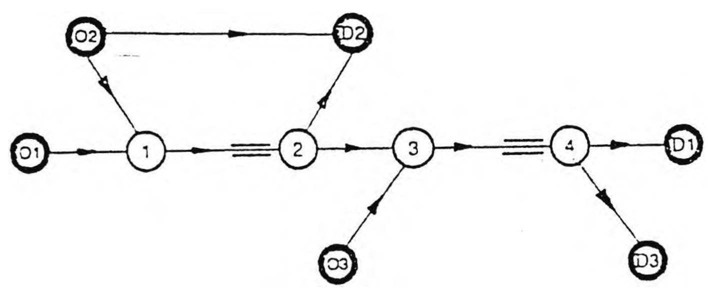


Figure 5.4

For this example, the optimality of the solution obtained by steps 1-5, will depend on the value of Δ. If $\Delta > 82.5$ or $\Delta < 60$, steps 1-5 produce the optimal time-varying pattern of flow and this is of no interest here. But, if $60 < \Delta < 82.5$, steps 1-5 produce a sub-optimal pattern of flow. To see that, allow the flow of commodity 2 to follow path O2-1-2-D2. In this case the total costs are, as before, 49500 veh-sec. But, if the flow of commodity 2 follows path O2-D2, the costs become 600Δ, which is less than 49500, as we assumed $\Delta < 82.5$. Though the latter is favourable, the total local costs due to travellers of commodity 2 following path O2-D2 instead of path O2-1-2-D2, have increased by 600Δ, which is greater than 36000 veh-sec, as, again, we assumed $\Delta > 60$. In other words, the failure of our algorithm to determine the optimal solution is due to allowing travellers of commodity 2 to follow path O2-1-2-D3, where their marginal costs is least along this path, rather than path O2-D2, where they could reduce the global costs. When travellers of commodity 2 follow path O2-1-2-D2, they delay travellers of commodity 1 clearing bottleneck B2 before any arrival of commodity 3, thus forming a queue at this bottleneck and increasing the total system costs.

In view of this example, the model hence does not in general determine a system optimum pattern of flow, though it is highly likely to determine an approximately system optimal pattern of flow that is least costly than a user equilibrium pattern of flow, as the network tests of Sect. 5.4 show.

Nonetheless, there is a single-bottleneck case where our algorithm can be guranteed to determine the system optimal pattern of flow. This single-bottleneck case is given in Smith and Ghali (1990) and Smith (1991), and corresponds to a network where the routes connecting each origin-destination pair in the network passes through no more than a single active bottleneck.

In addition to the single-bottleneck case, where the optimality of the solution can be guranteed, we conjecture that the algorithm is also optimal for a single commodity network, as in D'Ans and Gazis (1976) and Carey (1987), amongst others. But, we have not been able to prove this.

2- Another limitation of the method suggested in this paper is the need to store the routes of all the packets. This may not indeed be possible for reasonably large networks. This limitation is difficulty to get by, for the routes of the packets are firstly needed to maintain the "first-in, first-out" discipline. Secondly, if we consider two consecutive iterations of our alogorithm and defined the first of these as the previous iteration and the second as the current iteration, then the route of each packet is needed to take into account the size of the packet that was assigned in the previous iteration, while routing the packet in the current iteration. Perhaps this latter difficulty may be resolved if we allow splitting of the packets, but, maintaining FIFO would become a problem.

6. DYNAMICALLY-CONTROLLED CONGESTED NETWORKS

6.1 INTRODUCTION

In Chapter 3, results in the steady state for controlled and incremently congested road networks were presented within the context of the iterative assignment/control procedure. In this procedure each traffic light is considered in isolation and a solution is thought to have been obtained when there is no more change in the signal settings due to updated flows that are produced by a traffic assignment step. The iterative assignment/control procedure in Chapter 3 is a modification of that in Allsop *et al* (1977), to allow for a gently rising demand in the steady state.

In that chapter we compared three different traffic control policies, namely: the standard "delay minimisation" policy of Allsop (1971), Webster's Policy (1966), and P_0 policy, devised by Smith (1979). The cost function was Webster's. The resulting signal settings as well as the traffic flows were there assumed to be time-independent, and our first aim was particularly to show that when the demand is steadily and gently increasing for all origin-destination pairs of the network, some signal control policies may not accommodate the total demand, in which case total costs tend to infinity if a delay formula, such as Webster's, was used to set the signals. Or, if all the demand was accommodated, our second, but equally important, aim was also to show any differences in network performance arising from using one policy against another.

In this chapter, because it is rather unrealistic to model delays by a function such as Webster's, where costs rise to infinity if the capcity of some link is exceeded and the solution of the equilibrium assignment problem becomes infeasible, a different approach is adopted. Namely, we assume that queues as well as link flows are a function of time and that the traffic light settings are time-dependent. Here, queues are modelled explicitly, and, more importantly, costs are functions of both queue length and travelling on links. The deterministic queuing model of Figure 5.1 was used.

To set the traffic lights in a time-dependent fashion, this required first formulating a corresponding policy to each of "delay minimisation" and P_0 policy in the dynamic state, while Webster's was considered as in CONTRAM. These corresponding policies then become the three traffic control policies of this paper.

For the queuing model of Figure 5.1, infeasibility due to costs rising to infinity do not occur. Accordingly, the first aim of Chapter 3 is no longer the issue, but we retain the other, and compare network performance under these corresponding control policies by similarly alternating between assignment and control as in Chapter 3. The assignment here is a dynamic assignment, and the demand for each origin-destination pair is supposed to be fixed, given and generally time-varying.

The dynamic equilibrium assignment step in this frame of work corresponds to an equilibrium at which the costs experienced by drivers on arriving at their destination and travelling at reasonably close intervals of time, are more or less equal. In fact, this statement, concerning costs experienced by drivers, is an approximation of Wardrop's first principle, as here the term "drivers travelling at reasonably close interval of time", rather than at the same time, as in Wardrop (1952), is assumed instead. The reason for our approximation of Wardrop's first principle is because the dynamic equilibrium assignment model applied here is that of CONTRAM (Leonard *et al* (1978)); this regards the demand as formed of packets that leave the origin points at times corresponding to the midlle of the packet departure time interval in the original demand profile, and all the vehicles within a packet are assigned to only a single path. Hence packet splitting is not allowed, and costs cannot be precisely equilibrated.

In addition to the purpose of comparing the three control policies in a dynamic context, a further aim of this chapter is also to provide some results obtained by incorporating an approximate algorithm for the dynamic system optimum traffic assignment problem, which we describe in Chapter 5 (See also Ghali and Smith (1991)), instead of the dynamic user equilibrium assignment. As already mentioned in Chapter 5, the algorithm has a property that it is optimal for the "single bottleneck per route" case mentioned in Smith and Ghali (1990) and Smith (1991), and could be used for route guidance and for levying congestion tolls by charging a vehicle according to the costs it inflicts on others. An immediate outcome of these results is that alternating between a locally delay minimising policy and the approximate system optimal routeing strategy, does not generally solve the dynamic optimal control problem, as it does in the steady state.

With a view to comparing results obtained in the steady state with others in the dynamic state, as a further aim of this paper, we ran our static-assignment/control and dynamic-assignment/control programs for each of the networks which we describe later, and included the results of both the static and dynamic state.

The chapter is in the following format. The control policies incorporated in the dynamic traffic assignment models used, are given in Section 6.2. Then, in Section 6.3, the networks modelled are described and their results included. Section 6.4 gives some conclusions regarding the three control policies. In Section 6.5, we describe a method that monitors costs as it alternates between the approximate system optimum, given in Chapter 5, and locally delay minimising. This method is believed to be optimal for certain cases of network topology.

6.2 THREE SIGNAL CONTROL POLICIES

The setting of traffic lights for each of the three policies follows the line of CONTRAM. CONTRAM is a time-varying equilibrium assignment program that descretises the planning horizon (or the modelling period) into a number of time slices; during each the flow rate arriving into each link is assumed to be fixed. Though the signal settings are allowed to vary with time here, they are allowed to do so only from one time slice to another. By adding the flow rate arriving at a signal approach within a certain time slice to the initial queue, if any, from the previous time slice, for each time slice different settings can be calculated. Obviously, shortening of the time slice, results in higher accuracy. It should be incidently borne in mind that the settings may not vary within the light cycle if the time slice was shorter than the cycle length.

Notations

The following notations are needed.

q_{ij} denotes the initial queue for a certain time-slice added to the flow arriving within the time slice of stream j that belongs to stage i.

λ_i	=	the proportion of cycle that is effectively green for stage i.
w_{ij}	=	the saturation flow of stream j belonging to stage i.
n	=	the number of stages.
$i \sim j$		means stream j is in stage i.
$\max\limits_{i \sim j}[x_j]$	=	the maximum value of x_j in stage i.

Next, we give the traffic control policies we have implemented in our computer models.

Equisaturation policy

This policy in the static state is known as Webster's policy, but here we refrain from using this phrase, as its extension to the dynamic state was not suggested by Webster.

As formulated in CONTRAM, the equisaturation policy is such that the proportion of cycle that is effectively green, for stage i, is:

$$\lambda_i = \frac{\max\limits_{i \sim j}[q_{ij}/w_{ij}]}{\sum\limits_i \max\limits_{i \sim j}[q_{ij}/w_{ij}]} \qquad (i = 1, \ldots, n)$$

With the λ_i's determined by the above equation, the green time of each stage becomes $\lambda_i c$, where c is the light cycle. If any of the green stages does not satisfy minimum green contraints, then the settings are adjusted appproriately.

belongs to stage i.

λ_i = the proportion of cycle that is effectively green for stage i.

w_{ij} = the saturation flow of stream j belonging to stage i.

n = the number of stages.

$i\sim j$ means stream j is in stage i.

$\max_{i\sim j}[x_j]$ = the maximum value of x_j in stage i.

Next, we give the traffic control policies we have implemented in our computer models.

Equisaturation policy

This policy in the static state is known as Webster's policy, but here we refrain from using this phrase, as its extension to the dynamic state was not suggested by Webster.

As formulated in CONTRAM, the equisaturation policy is such that the proportion of cycle that is effectively green, for stage i, is:

$$\lambda_i = \frac{\max_{i\sim j}[q_{ij}/w_{ij}]}{\sum_i \max_{i\sim j}[q_{ij}/w_{ij}]} \qquad (i = 1, \ldots, n)$$

With the λ_i's determined by the above equation, the green time of each stage becomes $\lambda_i c$, where c is the light cycle. If any of the green stages does not satisfy minimum green contraints, then the settings are adjusted approriately.

<u>A queuing version of policy P_0</u>

The version of P_0 policy which has been used to allocate the green light for each stage is of the form:

$$q_1/\lambda_1 = q_2/\lambda_2 \ldots = q_n/\lambda_n,$$

for a junction with just one stream in each stage.

For a junction where there is more than one stream in each stage, this version can be shown to be the solution of the program:

$$\max_{\lambda} \sum_{i} \sum_{i \sim j} q_{ij} \ln \lambda_i$$

subject to

$$\sum_{i} \lambda_i = 1 - L/c$$

$$\lambda_i \geq g_i^{min}/c$$

This problem is convex and can be solved using any feasible direction optimisation method for linearly constrained programs. Obviously, any solution with zero effective green time of a stage is not a solution to this problem, as the objective function would become infeasible, unless all q_{ij}'s are zero in the zero-valued effective green time of the stage. Also, since the maximum possible value of the objective function is 0, corresponding to a stage having λ_i=1 with no lost time assumed, then the problem is bounded from above. Hence the problem is well defined.

"Delay minimisation" policy

The "delay minimisation" policy in the dynamic state for isolated junctions has been investigated by Gazis (1964) under the heading "single intersection" networks, where it is formulated as an optimal control problem. The solution has been shown to correspond to a *bang-bang* solution in optimal control. That is the settings correspond to optimal switches, or, to the solution of a linear program whose variables are effective green settings, upon discretising the optimal control problem. Since the solution of a linear program is an extreme point, the full green of the cycle is allocated to just one stage as long as there is enough flow within this stage. Then, if there is insufficient flow in that stage, the green light is given to another stage. And so on. Consequently, the solution is a *bang-bang*, unless fuel consumption or a restriction on stopping time is introduced into the formulation. However, if fuel consumption and the restriction on stopping time are omitted, and only queuing delays are considered, there is a much simpler interpretation and an easy method for determining the signal settings. This is described in the next paragraph.

Minimisation of queuing delays, which correspond to the area confined between the cummulative flow and discharge curves in Figure 5.1, for all approaches of the light, can be viewed as setting the light so that it is fully utilised, or operating at the maximum throughput available. For example, if there are a number of stages and in each there is flow to discharge, then giving sufficient green light to discharge all the flow to the stage which can discharge at the highest rate, mininmises queuing delays. This procedure is specified exactly in the following steps:

1- Let $k_j = 1$ and $\lambda_j = 0$, for $j = 1, \ldots, n$; sum = 0; $f = 1$

2- Determine the stage i with the maximum value of $P_i = \sum_{j \in i} w_{ij}$ and which satisfies the conditions, $q_{ij} \geq 0$ and $k_j = 1$.

3- Let $\lambda_i = \max_{i \sim j}[\min[(q_{ij}.c)/(w_{ij}.(c-L)), f]]$.

4- If $\lambda_i > 0$, then let $sum_{new} = sum_{old} + \lambda_i$.

5- Let $f_{new} = f_{old} - \lambda_i$, and $k_i = 0$. If $f_{new} = 0$, then go to to step 6. Else, return to step 2.

6- For $l = 1, \ldots, n$, let $g_l = \lambda_l.(c-L)/sum$.

6.3 TEST NETWORKS AND RESULTS

Two networks were used. For each network and policy, we studied network performance as congestion level is increased. Ten congestion levels were considered. Only network performance under each of the three signal policies are of concern in this study - as in Chapter 3, computational time and number of iterations needed to arrive at the solution were not regarded as significant as the performance, though these might differ largely within and between the three control policies.

For the steady state, as in Chapter 3 the link cost function is supposed to be a combination of a constant travelling time and a delay term, due to Webster, if the link in question has at its down stream end a traffic light.

For the dynamic state, constant travelling time, in addition to queuing delays due to input flow exceeding the service rate of the link, as in Figure 5.1, are assumed for both networks.

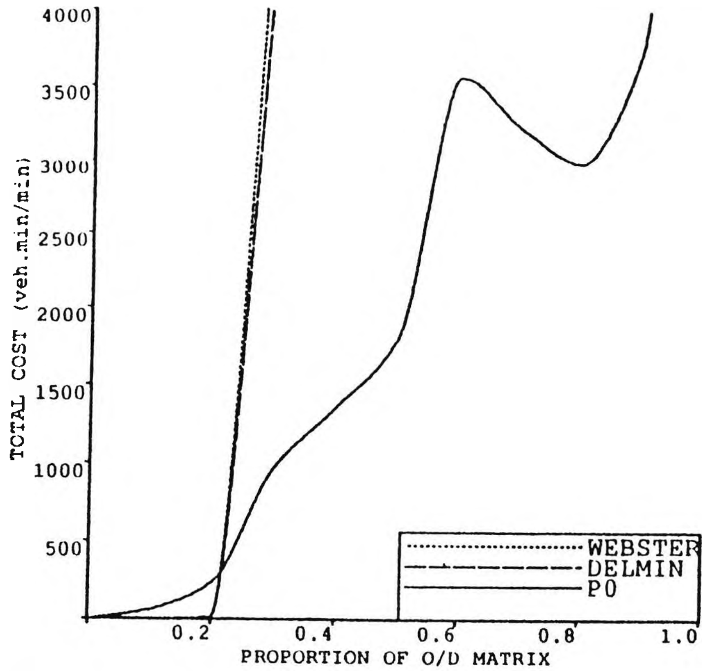
In the results given below, UE denotes results are obtained according to the user equilibrium routing strategy, and SO according to the approximate optimal routing strategy Chapter 5.

Time slice:	1	2	3	4	5	6	7	8	9
Demand (Veh/Hr):	1800	3600	7200	10800	14400	10800	7200	3600	1800

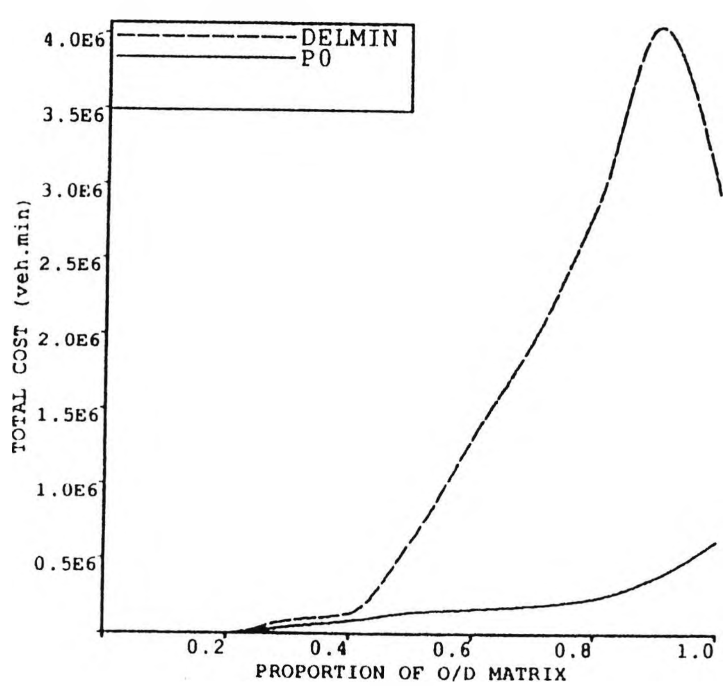
Table 6.1

Time slice:	1	2	3	4	5	6	7	8	9
Demand (Veh/Hr):	1800	3600	7200	10800	32400	10800	7200	3600	1800

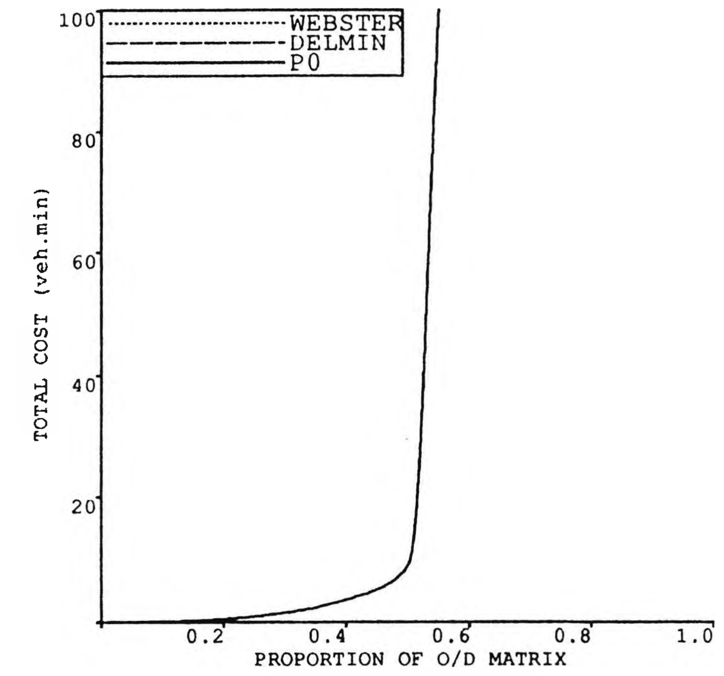
Table 6.2



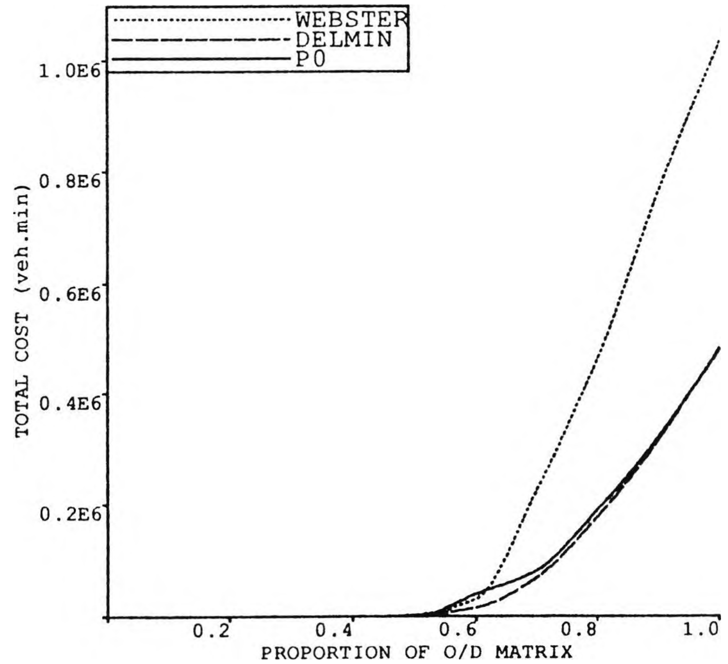
Graph 3.1



Graph 6.1



Graph 6.2



Graph 6.3

The description of the networks and their results are as follows.

Network 1

This is the network shown in Figure 3.3 of Chapter 3, but here the traffic lights are all A's instead of A's and B's in Figure 3.3. Again, junctions denoted as F were assumed to be flyovers or have large capacities.

Two cases were considered. Case 1 corresponds to a network where each signal-controlled link has a saturation flow of 1 veh/sec. This case is the same as Case 1 in Chapter 3 for the same network. Case 2 corresponds to a network where the saturation flow on each link at each traffic light along the central routes is six times greater than that of the saturation flow on the outer routes. For Case 2, the saturation flow of each link on outer route was taken as 1 veh/sec, and on the central routes was hence 6 veh/sec. The time-varying demand for the first case is given in Table 6.1, and in Table 6.2 for the second case. The steady state demand for the first case was 4 veh/sec from each origin, and 9 veh/sec for the second. All other properties of the network are the same as in Network 1 of Chapter 3.

Results of Network 1

The results are given in Graphs 6.1-6.3. Graph 6.1 and 6.3 are for the dynamic state results of Case 1 and 2, respectively. Graph 6.2 represents results of Case 2 for the dynamic state. Results of Case 2, dynamic state, are the same as in Graph 3.1 of Chapter 3, but included here for ease of reference.

Case 1, UE, steady state: This is Graph 3.1.

Case 1, UE, dynamic state: In Graph 6.1, the equisaturation policy and the queuing version of P_0 (only P_0 is shown on Graph 6.2), had similar performance, while "delay minimisation" surprisingly did not behave well. On Looking at the outputs of the computer runs concerning the allocation of green lights by "delay minimisation" it appeared that the central routes were given most of the green light, and the outer routes had only minimum green light. Webster's and the queuing version of P_0, on the other hand, kept the outer routes more open, which meant more throughput; hence, less costs.

Case 1, SO, dynamic state: Because "delay minimisation" did not behave well for Case 1, it was then incorporated into the approximate system optimum algorithm described in Chapter 5, for two levels of demand, 0.5 and 0.6 proportions of the total demand. Though costs went down at these two levels of demand, they did not however result in lesser costs than the costs due to applying the equisaturation policy together with a user equilibrium pattern of flow on the one hand, or due to P_0 together with a user equilibrium pattern of flow, on the other hand. For 0.5 and 0.6 proportions of the demand given in Table 6.1, the costs with the approximate system optimum and "delay minimisation" were for each level, respectively, 217716 and 494740 veh-min, whereas the user equilibrium with the equisatuaration policy had 137044 veh-min and 160537 veh-min, for the same levels of demand.

Case 2, UE, steady state: As shown in Graph 6.2, the policies had similar performance. This is because P_0 in this instance responds to traffic flows in the same way as Webster's policy and the delay minimisation policy.

Case 2, UE, dynamic state: The results are in Graph 6.3. Here P_0 and "delay minimisation" had more or less the same performance, while the equisaturation policy had a relatively poor performance.

Comments: The results obtained for this network are surprising in two ways. *Firstly*, comparing *UE, steady state* against *UE, dynamic state* it appears that there is a large difference or inconsistency between the behaviour of a policy in the steady state as opposed to its coorresponding in the dynamic state. In Graphs 3.1 and 6.1, Webster's policy and the delay minimisation policy each behaved badly as compared to P_0. In contrast, in Graph 6.1, only delay minimsation behaved badly, compared to the queuing version of P_0. Similarly, in Graph 6.2 and 6.3, Case 2, the behaviour of Webster's policy in the steady state did not carry over as before to the dynamic state (Equisaturation policy), and in this instance it was rather at odds with P_0, unlike in Case 1. *Secondly*, the results obtained in relation to alternating between an approximate dynamic system optimum and a locally delay minimisation policy confirm that a locally delay minimising policy is not generally globally optimal, as it is in the steady state, for convex formulations. Note that the approximate dynamic system optimum described in Chapter 5 is exact or optimal for this network.

Network 2

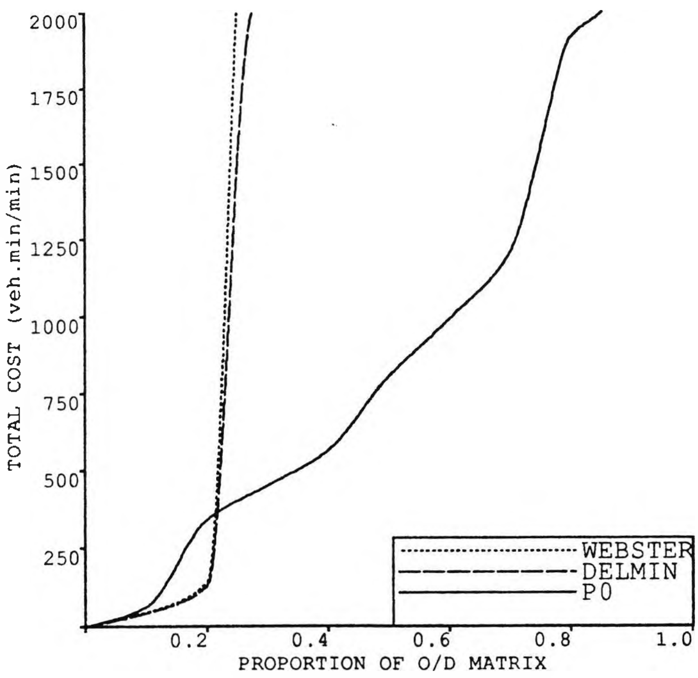
This network is shown in Figure 3.5 of Chapter 3. The capacity of each link is given in Table 6.3. All other properties of this network, such as demand and link travel time, are given in Chapter 3.

Results of Network 2

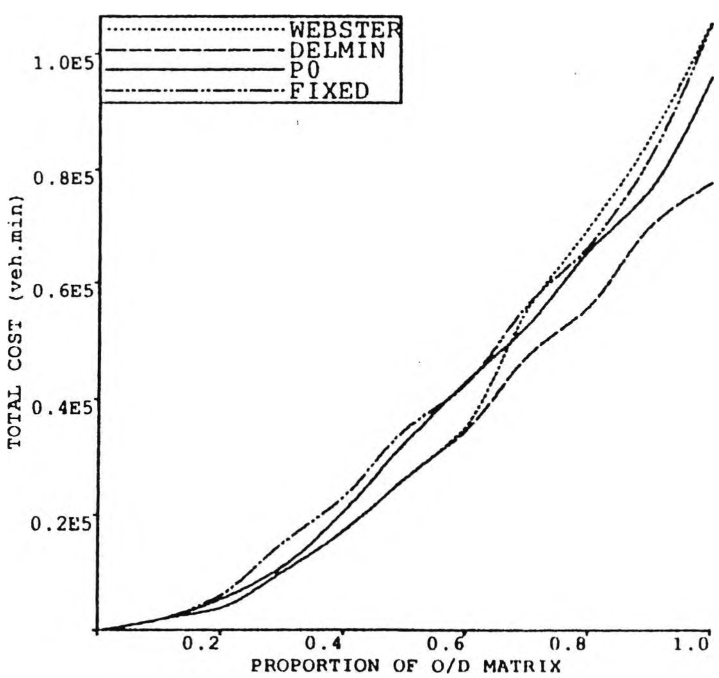
UE, steady state: This is Graph 3.7.

Link	Link Capacity (Veh/Hr)	Link	Link Capacity (Veh/Hr)
1 - 10	3150	2 - 11	3680
3 - 14	3156	4 - 16	3150
5 - 17	2900	6 - 18	1975
7 - 19	2900	8 - 22	1975
9 - 10	3680	10 - 1	10000
10 - 9	10000	10 - 11	500
10 - 22	500	11 - 2	10000
11 - 10	3680	11 - 12	3550
12 - 11	4150	12 - 13	500
12 - 22	3150	13 - 12	3175
13 - 14	10068	13 - 21	2550
14 - 13	3550	14 - 15	4150
14 - 23	2550	15 - 14	7240
15 - 16	3550	15 - 17	3150
16 - 4	10000	16 - 15	3550
16 - 17	500	14 - 3	10000
17 - 5	10000	17 - 15	2550
17 - 16	375	17 - 18	500
17 - 23	2900	18 - 6	10000
18 - 17	500	18 - 19	500
19 - 7	10000	19 - 18	500
19 - 20	500	19 - 23	2550
20 - 21	550	20 - 23	500
21 - 13	2900	21 - 20	2550
21 - 22	500	22 - 8	10000
22 - 10	500	22 - 12	3150
22 - 21	500	23 - 14	2550
23 - 17	2500	23 - 19	3150
23 - 20	1975	20 - 19	500

Table 6.3. Assumed capacity of links of Network 2.



Graph 3.7



Graph 6.4

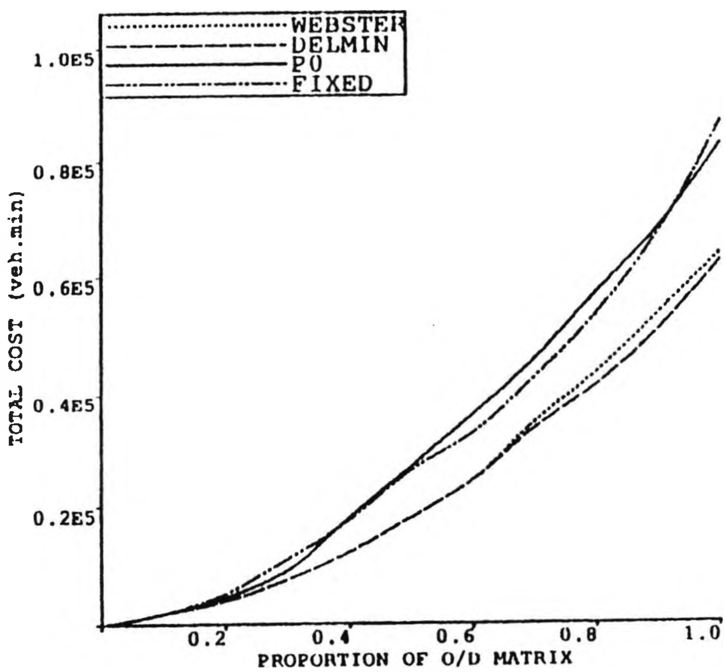
<u>UE, dynamic state</u>: Graph 6.4 shows "delay minimisation" as the favourable policy to adopt for this network, for the levels of demand 0.6-1.0. The Equisaturation policy may also be used for all levels of demand that are not greater than 0.6 instead of "delay minimisation". But, after the 0.6 level of demand, the Equisaturation policy tends to become poor in comparison to either P_0 or the "delay minimisation" policy.

<u>SO, dynamic state</u>: The results are given in Graph 6.5. The only significant remark that could be said about these results is that the delay minimisation policy when used in conjunction with the approximately system optimum, given in Chapter 5, as a routing strategy, it produced the best network performance, as compared to P_0 or the Equisaturation policy. Further, this is also true when compared to either the delay minimisation, P_0 or the Equisaturation policy, but in conjunction with the user equilibrium routing strategy.

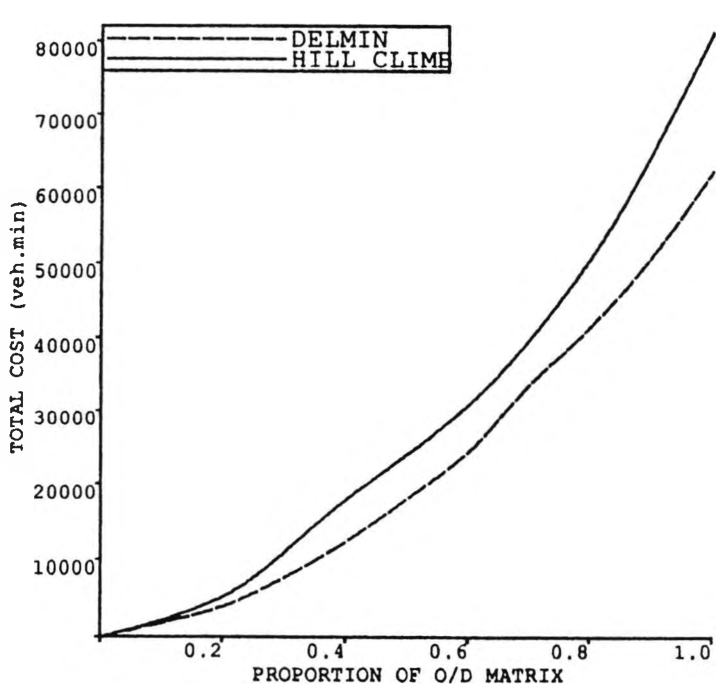
<u>Comments</u>: Again, as far as the performance of a policy in the steady state, compared to its corresponding in the dynamic state, the notably nice behaviour of P_0 in the steady did not have the same impact in the dynamic state.

6.4 CONCLUDING REMARKS CONCERNING THE THREE CONTROL POLICIES

Confining the argument to the dynamic state only, indeed, it is difficult to draw any conclusions as to when to favour the use of a signal control policy against another for a general network. The variability between the policies in the above networks is obvious – in Network 1, *Case 1, UE, dynamic state*, P_0 behaved well in contrast with "delay minimisation", but not in Network 2, *UE, dynamic state*. On the other hand, in Network 1



Graph 6.5



Graph 6.6

delay minimisation was better than the Equisaturation policy in *Case 1, UE, dynamic state*, but not in *Case 2, UE, dynamic state*

If this variabilty was to suggest a general approach, it would suggest that one would have to study each network, which is being modelled, separately to determine the policy to adopt and implement the one which results in a better system performance. What is more is that the policy employed has to be revised routinely, as a different policy might become more favourable as the level of demand increases. This is clear in our results of Network 2, where delay minimisation and the Equisaturation policy in the dynamic state had the same network performance up to 0.6 level of demand, but not after this level.

Though, P_0 seems to be better for bypasses, as it penalises the excessive use of a town center and, consequently, diverts drivers to higher capacity roads.

Another suggestion may be to consider the option of solving for the dynamic optimal control problem under the assumption of a user equilibrium routing strategy. This, in principle, should result in the best system performance, unless there is more than one solution and the solution determined was not any better than a solution obtained by alternating between assignment and control, as in this paper. Even if there is only one solution, it may be hard anyway to determine the optimal settings for two difficulties. 1) because, as mentioned in Chapter 2, no algorithm yet exists for the dynamic optimal control problem under that assumption, and 2) origin-destination demand profiles are neither easily obtained nor are they always available. Perhaps the second difficulty may now be defused with the introduction of automatic electronic monitoring devices, such as smart cards, that could be used to survey the origin-destination zone particular of each vehicle. However, the first difficulty remains unresolved. In fact, it is this difficulty which has triggered earlier studies on control and assignment in the steady state, and this first study in a dynamic context.

6.5 LOCAL DELAY MINIMISATION WHILE ACCOUNTING FOR FIFO

Algorithms which optimise green times at each traffic light in isolation may not only by and large yield non-optimal signal settings, but may also destroy the first-in, first-out queuing discipline in a multi-commodity network, which a dynamic assignment process, such as the method described in Chapter 5 and CONTRAM, tends to satisfy.

In what follows we describe a method which accounts for first-in, first-out and monitors the total costs, while varying the signal settings of each traffic light independently.

The method described here uses a hill climbing optimisation method with fixed routes obtained from either an equilibrium assignment step or from the approximate system optimal given in Chapter 5.

On each variation of the settings of some traffic light, an iterative procedure to satisfy FIFO is employed to reassign the flow along the fixed routes found optimal in the assignment step.

Having satisfied FIFO, the total costs are evaluated and, if decreasing, the signal settings of the traffic light whose settings are varied are implemented. Obviously, if the variation resulted in an increase in costs, a move in the opposite direction is attempted.

Formally, the method may be outlined as in the following steps:

1- Solve an assignment program and fix the time-varying routes for each origin-destination pair.
2- Using a hill climbing optimisation method, for each traffic light, vary its settings and implement an iterative procedure to reassign the flow along the fixed routes until the arrival time of each vehicle or packet of flow, has settled down. If the variation resulted in less costs, then implement. Otherwise, move the settings in the opposite direction.
3- If the total costs obtained in step 1 and 2 do not vary greatly, then terminate. Otherwise, return to step 1.

Although this procedure monitors costs and satisfies FIFO, it does not for a general network determine an optimal solution, unless perhaps each route joining each origin-destination pair passes through no more than a single bottleneck (a traffic light in this setting), as in Smith and Ghali (1990) and Smith (1991). The inability of the method to produce an optimal solution could be readily seen if the network of Figure 6.1, which is that of D'Ans and Gazis (1976), is considered in the context of the above method. For this network, there is no route choice, and the throughput when full green is given to link (1,7) and (7,8) at junction 7 and 8 respectively, is assumed to be greater than the total throughput when link (2,7) and (3,8) receive full green instead. Assuming initially that the signalised junction 7 and 8 were set so that the throughput across in the horizontal direction was less than that in the vertical direction, by allocating more green to the arms in the vertical direction, then varying appropriately the settings of the signalised junction 7 does not increase the throughput in the horizontal direction since this is still influenced by the settings at junction 8, unless both lights are varied simultaneously.

In view of the fact that locally delay minimising at each signal

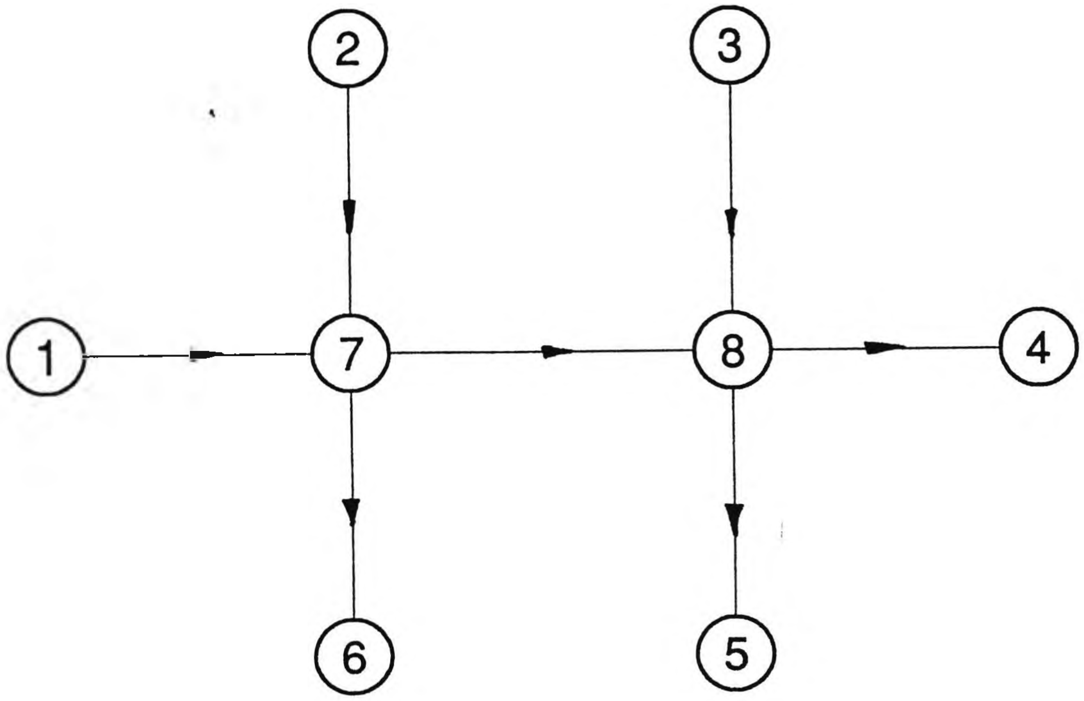


Figure 6.1

separately while accounting for first-in, first-out does not on the whole give the optimum solution, even when the routes are supposed fixed, various heuristic schemes might be suggested. One such scheme may be to alter the settings of a traffic light and allow the others or the downstream junctions to respond to the new output of the varied traffic light, upstream. Another scheme would rely on the judgment of the traffic engineer to define the traffic signals which could be made responsive and those whose settings could be hill climbed. Both suggestions might or might not reduce congestion, but the optimal settings are likely to remain undetected. Therefore, a more encompassing approach is needed.

In spite of that the method suggested in the above three steps is non-optimal, it was implemented in CONTRAM and applied to the network shown in Figure 6.1. The results shown in Graph 6.6 show that delay minimisation of Section 6.2, together with the approximately system optimum of Chapter 2, outperformed the method described in step 1-3.

REFERENCES

Abdulaal, M. and L.J. LeBlanc, "Continuous Equilibrium Network Design Models", *Transpn. Res.* **13B**(1), 19-32, (1979).

Allsop, R.E., "Delay Minimising Settings for Fixed-Time Traffic Signals at a Single Road Junction", *J. Instit. Math. Appl.*, **8**(2), PP. 164-185, (1971).

Allsop, R.E., "Some Possibilities for Using Traffic Control to Influence Trip Distribution and Route Choice", Transportation and Traffic Theory, Proc. of Sixth International Symposium on Transportation and Traffic Theory, edited by D.J. Buckley, PP. 345-375, Elsevier, New York, (1974).

Allsop, R.E. and J.A. Charlesworth, "Traffic in a Signal-Controlled Road Network: an Example of Different Siganl Timinings Inducing Different Routings", *Traffic Engineering and Control*, **18**(5), 262-264, (1977).

Bazaraa, M. S. and J. J. Jarvis, "*Linear Programming and Network Flows*", John Wiley & Sons, New York, (1977)

Beckmann, M., C. McGuire and C.B. Winsten, "*Studies in the Economic of Transportation*", Yale University Press, New Haven, Connecticut, (1956).

Ben-Ayed, O., D.E. Boyce and C.E. Blair III, "A General Bilevel Linear Programming Formulation to the Network Design Problem", *Transpn. Res.*, **22B**(4), 311-318, (1988).

Carey, M., "Optimal Time-Varying Flows on Congested Networks", *Operations Research*, **35**(1), 58-69, (1987).

Carey, M. and A. Srinivasan, "Solving a Class of Network Models for Dynamic Flow Control", Working Paper 88-13, School of Urban and Public Affairs, Carnegie Mellon University, (1987).

D'Ans, G.C. and D.C. Gazis, "Optimal Control of Oversaturated Store and Forward Transportation Networks", *Transportation Science*, **10**, 1-19, (1976).

Dafermos, S.C. and F.T. Sparrow, "The Traffic Assignment Problem for a General Network", *J. Res. Nat. Bureau of Standards- B Mathematical Sciences*, **73B**(2), (1969).

Dafermos, S.C., "Traffic Equilbrium and Variational Inequilities", *Transpn. Sci.*, **14**, PP. 42-54, (1980).

Daganzo, C.F., "On the Traffic Assignment Problem with Flow Dependent Costs- I", *Transpn. Res.*, **11**, PP. 433-437, (1977).

Dantzig, G.B. *et al.* "Formulating and Solving the Network Design Problem by Decomposition", *Transpn. Res.* **13B**, 5-18, (1979).

Dijkstra, E.W., "A Note on Two Problems in Connexion with Graphs", *Numerical Mathematics* **1** , PP. 269-271, (1959).

Gazis, D.C., "Optimal Control of a System of Oversaturated Intersections", *Operations Research*. **12**, 815-831, (1964).

Gershwin, S. B. and H-N. Tan, "Hybrid Optimisation: Optimal Static Control Constrained by Drivers' route choice behaviour", Massachusetts Institute of Technology, Laboratory for Information and Decision System Report *LIDS-P-870*, (1978).

Ghali, M.O., "A Note on The Dynamic User Equilibrium Traffic Assignment Problem", *Transportation Research-B*, (to be published).

Goldfarb, D., "Extension of Davidon's Variable Metric Method to Maximisation Under Linear Inequality and Equality Constraints", *SIAM J. Appl. Math.*, **17**(4), 739-764, (1969).

M D Hall, D Van Vliet and L G Willumsen, "SATURN - a simulation-assignment model for the evaluation of traffic management schemes", Traffic Engineering and Control, **21**, 168-176 (1980).

Hearn, D.W., S. Lawphongpanich and J.A. Ventura, "Restricted Simplicial Decomposition: Computation and Extensions", *Mathematical Programming Study*, **31**, 99-118, (1987).

Hendrickson, C. and G. Kocur, "Schedule Delay and Departure Time Decisions in a Deterministic Model", *Transportation Science*, **15**(1), (1981).

Heydecker, B.G., "Some Theoretical Difficulties in Using Equilibrium Assignment in Signal-Controlled Road Networks", proceedings of the PTRC Summer Annual Meeting, Warwick University, P198, 1-12, July (1980).

Heydecker, B.G., "Some Consequences of Detailed Junction Modelling in Road Traffic Assignment", *Transportation Science*. **17**(3), 263-281, (1983).

Heydecker, B.G., "The Equilibrium Network Design Problem: a Critical Review", paper presented to NATO Advanced Workshop, Capri, October (1986).

Heydecker, B.G. and T. K, Khoo, "The Equilibrium Network Design Problem", (unpublished) paper presented at the 21st annual meeting of Universities Transport Study Groups, Hatfield, London, (1990).

Hooke, R. and T.A. Jeeves, "Direct Search Solution of Numerical and Statistical Problem", *J. Assn. Comp. Mach.* **8**, 212-229, (1961).

Leblanc, L.J., E.K. Morlock, and W.P. Pierskalla, "An Accurate and Efficient Approach to Equilibrium Traffic Assignment on Congested Networks". *Transportation Research Record* **491**, Interactive Graphics and Transportation Systems Planning, PP. 12-33, (1974).

Leonard, D.R., J.B. Tough and P.J. Bagueley, "CONTRAM, a Traffic Assignment Program for Predicting Flows and Queues in Peak-Periods" TRRL Report 841, Crowthorne, (1978).

Luenberger, B.G., "*Linear and Nonlinear Programming*", Addison-Wesley, Reading, Massachusetts, (1984).

Mahmassani, H. and R. Herman, "Dynamic User Equilibrium Departure Time and Route Choice on Idealised Arterials", *Transportation Science*, **18**(4), 362-384, (1984).

Marcotte, P., "Network Optimisation with Control Parameters", *Transpn. Sci*, **17**(2), 181-197,(1983).

Marcotte, P., "A Note on a Bilevel Programming Algorithm by Leblanc and Boyce", *Transpn. Res.*, **22B**(3), 233-237, (1988).

Merchant, D.K. and G.L. Nemhauser, "A Model and an Algorithm for the Dynamic Traffic Assignment Problem", *Transportation Science*, **12**(3), 183-199, (1978).

Pierre, D.A. and M.J. Lowe, "*Mathematical Programming via Augmented Lagrangians*". Addison-Wesley, Reading, Massachusetts, (1975).

Poorzahedy, H. and M.A. Turnquist, "Approximate Algorithms for the Discrete Network Design Problem", *Transpn. Res.*, **13B**(1), 45-55, (1982).

Potts, R.B. and R.M. Oliver, "*Flows in Transportation Networks*". Academic Press, New York, (1972).

Smith, M.J., "Traffic Control and Route-Choice: a Simple Example", *Transpn. Res.*, **13B**, 289-294, (1979a).

Smith, M.J., "Existence, Uniqueness and Stability of Traffic Equilibria", *Transportation Research* **13B**, PP. 295-304, (1979b).

Smith, M.J., "A Descent Algorithm for Solving a Variety of Monotine Equilbrium Problems", *Transpn. Sci.*, **18**, 385-494, (1984).

Smith, M.J., "The Existance of a Time-Dependent Equilbrium Distribution of Arrival at a Single Bottleneck". *Transpn. Sci.* **18**(4), PP. 385-394, (1984).

Smith, M.J., "Traffic Signals in Assignment", *Transportation Research*, **19B**(2), 155-160, (1985).

Smith, M.J., "Multi-Commodity Dynamic System Optimal Traffic Assignment", *Operations Research*, (to be published).

Smith, M.J. and M.O. Ghali, "The Interaction Between Traffic Flow and Traffic Control in Congested Urban Road Networks", paper presented at the joint Italian/USA Seminar on Congested Urban Networks: Traffic Control and Dynamic Equilibrium, Capri, June (1989). To be published in *Transportation Research-B* also.

Smith, M.J. and M.O. Ghali, "Dynamic Traffic Assignment and Dynamic Traffic Control", Proceedings of the Eleventh International Symposium on Transportation and Traffic Theory", 223-263, Yokohama, Japan, (1990).

Steenbrink, P., "*Optimisation of Transport Networks*", John Wiley and Sons, New York, (1974).

Tan, H-N., S.B Gershwin and M. Athans, "Hybrid Optimization in Urban Traffic Networks", Final Report DOT-TSC-RSPA-97-7, MIT, Cambridge, (1979).

Thompson, W.A. and H.J. Payne, "Traffic Assignment on Transportation Networks with Capacity Constraints and Queueing". Paper presented at the 47th National ORSA/TIMS North American Meeting, (1975).

Vincent, R.A., A.I. Mitchell and D.I. Robertson, "User Guide to TRANSYT Version 8", TRRL Report 888, Crowthorne, (1980).

Wardrop, J.G., "Some Theoretical Aspects of Road Traffic Research", *Proc. Instit. Civil Eng. Part II*, **1**(2), PP. 325-378, (1952).

Webster, F.V. and B.M. Cobbe, "Traffic Signals", Road Research Technical Paper, no. 56, HMSO, London, (1966).

Wie, B-W., T.L. Friesz and R.L. Tobin, "Dynamic User Optimal Assignment: a Control Theoretic Formulation", paper presented at the US/Italy joint seminar on Urban Trafffic Networks. Capri, Italy, June 1989.

Yagar, S., "Dynamic Traffic Assignment by Individual Path Minimisation and Queueing". *Transportation Research*, **5**, 179 - 196, (1971).

Zawack, D.J. and G.L. Thompson, "A Dynamic Space-Time Network Flow Model for City Congestion", *Transportation Science*, **21**, 153-162, (1987).