10-31-1993

Traffic management of a satellite communication network using stochastic optimization

Ambalavanar Arulambalam
New Jersey Institute of Technology

ABSTRACT

Traffic Management of a Satellite Communication Network Using Stochastic Optimization

by
Ambalavanar Arulambalam

The performance of nonhierachial circuit switched networks at moderate load conditions is improved when alternate routes are made available. However, alternate routes introduce instability under heavy and overloaded conditions, and under these load conditions network performance is found to deteriorate. To alleviate this problem, a control mechanism is used where, a fraction of the capacity of each link is reserved for direct routed calls.

In this work, a traffic management scheme is developed to enhance the performance of a mesh-connected, circuit-switched satellite communication network. The network load is measured and the network is continually adapted by reconfiguring the map to suit the current traffic conditions. The routing is performed dynamically. The reconfiguration of the network is done by properly allocating the capacity of each link and placing an optimal reservation on each link. The optimization is done by using two neural network-based optimization techniques: simulated annealing and mean field annealing. A comparative study is done between these two techniques. The results from the simulation study show that this method of traffic management performs better than the pure dynamic routing with a fixed configuration.

TRAFFIC MANAGEMENT OF A SATELLITE COMMUNICATION NETWORK USING STOCHASTIC OPTIMIZATION

by
Ambalavanar Arulambalam

A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Electrical Engineering

Department of Electrical and Computer Engineering

October 1993

APPROVAL PAGE

Traffic Management of a Satellite Communication Network Using Stochastic Optimization

Ambalavanar Arulambalam

Dr. Nirwan Ansari, Thesis Advisor (date)
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Zoran Siveski, Committee Member (date)
Assistant Professor of Electrical and Computer Engineering, NJIT

Dr. Edwin Hou, Committee Member (date)
Assistant Professor of Electrical and Computer Engineering, NJIT

BIOGRAPHICAL SKETCH

Author: Ambalavanar Arulambalam

Degree: Master of Science in Electrical Engineering

Date: October 1993

Undergraduate and Graduate Education:

- Master of Science in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 1993
- Bachelor of Science in Electrical Engineering,
  New Jersey Institute of Technology, 1992

Major: Electrical Engineering

This thesis is dedicated to
my family

ACKNOWLEDGMENT

The author wishes to express his sincere gratitude to his supervisor, Dr. Nirwan Ansari, for his guidance, friendship, and moral support throughout this research.

The author also extends his thanks to Drs. Zoran Siveski and Edwin Hou for serving as members of the thesis committee.

Special thanks to Santhalingam Balasekar for providing essential software and for his assistance with the simulations.

Lisa Fitton provided timely help and suggestions during the editing of the thesis.

None of this research would have been possible without the financial assistance of the author's uncle, Dr. T. Sivendran, M.D., who supported him in this and in countless other ways throughout his college career.

The author deeply appreciates the eternal love and continuous encouragement from his parents and his sister.

Finally, the author wishes to acknowledge his colleagues at the Center for Communications and Signal Processing Research at New Jersey Institute of Technology, especially Henry Hanek, Amit Shah and Zhiqiang Zhang, for their friendship and support throughout this research.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Satellite communications has gone through a period of tremendous growth, and has contributed in no small measure to the information explosion that has changed the world in the last quarter century. Satellites, from their geostationary orbit position, 22,300 miles over the equator, view over one-third of the earth and can instantly connect any pair of points within their coverage [1]. This property, together with their record of high reliability, makes them the most desirable multiple access diversity communication medium.

The decade from 1990 to 2000 will see the completion of a transition that began in the 1980s, of the world-wide, public telephone network from a mostly analog medium to an all digital one. Satellites have carried two-thirds of the world's trunk international communications for over two decades and have led in digitization of international links [2], but their dominance is now challenged by the introduction of terrestrial and transoceanic submarine fiber light guides (optical cable), which provide a virtually inexhaustible supply of digital transmission capacity between the points they connect [3]. However, this capacity is confined inside the light guide and is accessible only close to its gateways. Large areas of the United States, Canada, South America, Asia, the Middle East, and even Europe have thinly distributed, but large aggregate populations. Economic considerations may indefinitely defer the availability of high capacity fiber to them [4]. Nonetheless they need a wide range of digital services which satellites, if properly designed, can deliver at low cost.

The satellite communications industry is now creating a new generation of satellites and communications techniques that will serve those markets best suited to satellites. Numerous schemes have been proposed to improve the efficiency of satellite communications. Among others, related works include channel allocation and scheduling. Many studies on circuit switched telecommunication networks are useful here since a wide spectrum of applications in satellite networks are circuit switched networks.

For circuit switched networks, it is known that nonhierachial routing performs better than the hierachial static routing [5]. The dynamic non-hierachial routing (DNHR) method of AT&T [6] and an adaptive routing algorithm known as dynamic control routing (DCR) that is being considered by Telecomm Canada [7] are two of the most common examples of nonhierachial networks. Several schemes have been proposed and studied by many in the field of networking [8]–[11]. From these studies it is well known that nonhierachial dynamic routing performs better than the hierachial static routing. It is also shown that allowing alternate routes in nonhierachial networks results in an improved performance at moderate loads but suffers severely at over load conditions. Many control mechanisms have been proposed to overcome this stability problem. A study done by [11] suggests reservation of some portion of the capacity at high loads avoids the instability. In a study done by [12], optimal trunk reservation is found for a fully connected, completely symmetric network.

In this work a traffic management scheme is proposed to improve the efficiency of a circuit switched satellite communication network of geostationary orbital type. This work is an extension of work done by [13]. The proposed scheme incorporates the idea of dynamically adapting the networks as well as dynamically routing each arrival. The scheme allows the network to change according to the traffic conditions, and thus, improves the grade of service. The system is modeled such that it continuously organizes itself to minimize the cost for varying traffic conditions.

The proposed traffic management scheme is described in detail in Chapter 2. An analytical model for the scheme is presented in Chapter 3. Two neural network-based techniques, which are used in the management scheme, are explained in

Chapter 4 and Chapter 5. Chapter 6 provides the simulation results and their analysis. The conclusions and possible extensions of this work are presented in Chapter 7.

CHAPTER 2
THE TRAFFIC MANAGEMENT SCHEME

2.1 Network Model

The global satellite communication network of today consists of a number of geostationary satellites, each satellite covering a number of ground stations. This kind of satellite network can be modeled as a mesh connected type of communication network, as in Figure 2.1. Each node represents either a geostationary satellite or an earth station. The connections between the nodes shown by the arrows denote the links between stations. The links may have any number of circuits, but the total capacity of the network is fixed. Traffic is modeled as being generated by purely random (e.g., Poisson) sources characterized by two parameters: an average rate of message or call generation and an average length of message or call duration. The satellite communication system is modeled as an appropriate server system which provides a transmission service to the generated traffic or "customer."

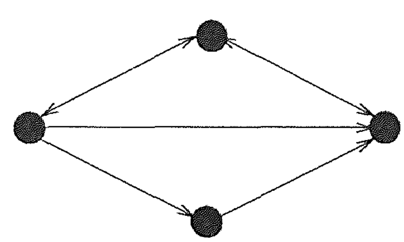

Figure 2.1 A Mesh-Connected Network with 4 Nodes and 7 Links

2.2 Traffic Management Scheme

The objective here is to design a network such that the overall block rate of the network is minimized and thus throughput is maximized. Since traffic conditions (i.e., the arrival rate) changes from time to time, a scheme which will dynamically route each call is proposed in this chapter. The proposed scheme can be explained with the help of the block diagram shown in Figure 2.2. The scheme is made up of four functional modules: map generator, router, controller and arbitrator. The function of each of the module is described in detail in the subsequent sections.

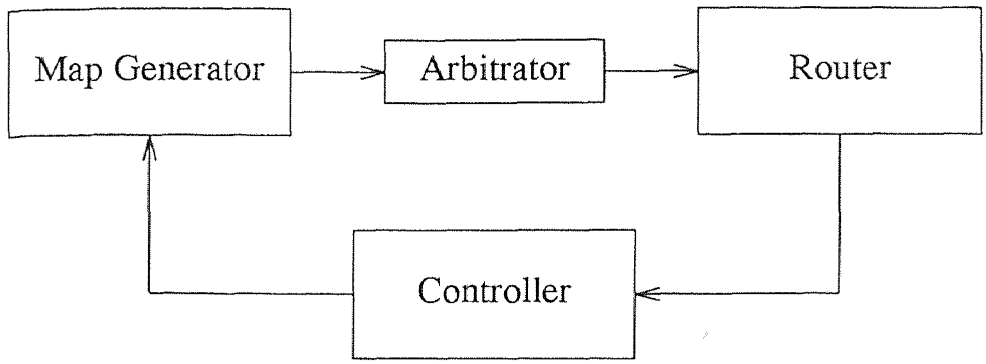

Figure 2.2 System Model

2.2.1 Map Generator

The function of the map generator is to generate a map of the best configuration for current traffic conditions. Maps differ from each other by two parameters namely $\vec{c}$ and $\vec{r}$. Vector $\vec{c}$ denote the link capacities of the network, and the elements of vector $\vec{r}$ denotes the number of circuits that can be used by the alternately routed calls. Therefore $c - r$ circuits in a particular link are reserved only for direct calls. The parameter, $c - r$, is referred to as the *reservation parameter*. Average arrival rates for each O-D pair, the total capacity of the network, and the current status of the network are given as inputs to this module. Based on this information, an optimization technique is used to find an optimal map which will minimize the total block rate of the network. For experimental purposes two kind of optimization techniques, simulated annealing and mean field annealing, are employed. Map generation by simulated annealing is discussed in Chapter 4 and map generation by mean field annealing is discussed in Chapter 5.

Apart from the reservation scheme, previous studies have shown that the proper assignment of link capacities improve the network performance. The optimal performance is obtained when the capacity assignment is such that all links saturate simultaneously [15]. In this ideal case, the flow-to-capacity ratio of all links shall be equal to the average flow-to-capacity ratio of the network. Also, it is known that circuit reservation for first-routed calls are effective as control mechanism against instabilities at high load conditions [11]. Based on these observations, it is clear that a map should be configured by selecting appropriate values for link capacities and the reservation parameter.

2.2.2 Router

The router performs the routing dynamically for every call arriving at the network, as follows.

1. If the direct link has any idle circuit, an arriving call is routed on the direct link.

2. If the direct link has no idle circuits then a randomly selected alternate route is tried. An alternately routed call will be blocked if either one or both links corresponding to that particular O-D pair is in the reserved state (i.e., at least r circuits in the link are busy).

3. If direct link routing and alternate routing fail, then the call is blocked and the call is lost from the network.

The routing module performs a simple routing function without much computation and thereby reduces the processing delay of each call. There are other schemes which try to dynamically compute the alternate route that minimizes the blocking rate. For instance, the least busy alternative (LBA) model developed by [12] computes a least busy alternate route. Here in this work the proper choice of map eliminates the need for computing the least busy alternate route. Also, such computations, as it will become evident in later chapters, will reduce the efficiency of the map generation process.

2.2.3 Controller

The controller's job is to keep track of the network's status and performance. The controller decides whether a new map is necessary based on the current network status, which is updated at regular intervals. The arrival rate of calls to each O-D pair and the load balance of the network are two of the more important parameters that the controller keeps track of.

Measurement of Load Imbalance

Let Δ denote the ratio of the network's total flow to the total capacity, and let δ_{ij} denote the ratio of flow to the capacity of the link (i,j). At each update a measure of the network's load imbalance, denoted by d, is computed from these ratios using the sum-square-error method as given below.

$$\Delta = F/C$$

$$\delta_{ij} = f_{ij}/c_{ij}$$

$$d = \frac{1}{C}\sum_{(i,j)}(\Delta - \delta_{ij})^2, \qquad (2.1)$$

where C and F are the total capacity and the total flow of the network, respectively. Therefore, the parameter d indicates the amount by which the network's current load balance deviates from that of the fully balanced network. This measure of disparity in the network load is taken as an indication of potential premature saturation of the network. Let d_t be defined as a threshold value for this imbalance, with respect to the network load balance. In the simulations done here the threshold value is defined as follows.

$$d_t = (0.1) \times \Delta. \qquad (2.2)$$

When d is larger than d_t, the network is considered to be in the inefficient state and the controller module calls the map generator module to come up with a better map for the current traffic condition. When d_t is small and close to zero then even a small deviation of network load balance from the ideal condition is not tolerated. In this situation map generator is called upon very frequently. But too frequent changes in the configuration may not be cost effective when considering other costs associated with changing the configuration of a satellite network. To avoid this, parameter d is updated after a fixed number of network updates. In the simulations here, network status is updated at the end of every time unit and the parameter d is computed after 10 updates. Up to 100 time units are used in measuring the traffic pattern.

2.2.4 Arbitrator

The function of the arbitrator is to decide whether changing the map will be beneficial to network performance, and thus, it is used as a cost saving measure. The routing of calls must be uninterrupted and the optimization has to be done in real time. Hence, there may be some instances where the map configured from the most recent network experience may not actually reflect the optimal performance for present traffic conditions if the traffic pattern changes too quickly. Therefore, the arbitrator's function is to change the map that is being generated if the change will result in better performance. The above situation can, in fact, be eliminated to a certain extent by properly choosing the duration for which the network is monitored for the purpose of measuring the network traffic condition, and by properly choosing the tolerance level (i.e., by reducing the threshold value δ_t).

CHAPTER 3

ANALYTICAL MODEL

A queuing model is employed here to analyze the network. Since the average blockrate is used as the cost, an expression for the average block rate must be developed. Before carrying out the analysis, the following assumptions are made.

- New arrivals to any O-D pair follow the Poisson process.
- All arrivals, including overflow calls, to any link form a Poisson process and are independent.
- Holding times of calls are exponentially distributed.
- Each link is represented by an $M/M/m/m$ queuing model, where m is the number of circuits in that link.
- The average holding time of calls is assumed to be one time unit.
- Link blocking probabilities are independent.
- Processing and propagation delays are negligible.

The last assumption helps us to study the network based only on the proposed scheme, where only the block rate is used as the cost. Furthermore, this is close to real situation in circuit switched networks due to long average holding time. Even though calls arriving at an alternate route is clearly not Poisson, this is found to be a reasonable assumption especially when each link derives traffic from many end users. Similarly, link blocking independence assumption is found to be reasonable, and this greatly reduces the complexity of the analysis. Detail discussion of the validity of these assumptions can be found in [7]. The validity of these assumptions in this work are briefly discussed when analyzing the simulation results.

3.1 Notations

Some of the commonly used notations are listed below.

(i,j)	Link from node i to node j.
$(i-j)$	Call originating from node i and destined for node j, also denotes the O-D pair from node i to node j.
λ_{i-j}	External (new) arrival rate of $(i-j)$.
Λ	The total rate of input traffic to the network.
γ_{ij}	Total arrival rate to (i,j).
f_{ij}	Flow in (i,j).
c_{ij}	Capacity, in number of circuits, of (i,j).
r_{ij}	Number of circuits in (i,j) that allow alternately routed calls.
m	A tandem node used in alternate route.
M_{i-j}	Set of tandem nodes that forms the alternate routes for $(i-j)$.
R^k_{i-j}	An alternate route for $(i-j)$ with node k as the tandem node.
B_{i-j}	Probability that call $(i-j)$ is blocked from the network.
B_{ij}	Probability that any call is blocked in (i,j).
B^R_{ij}	Probability that an alternately routed call is blocked in (i,j).
$\bar{B}$	Average network blocking probability.

For notational convenience, in the following text subscripts are dropped when referring to a particular link and subscripts are used when the entire network is considered.

3.2 Calculation of The Block Rate

Denote the average network blocking probability as $\bar{B}$, which can be obtained by summing all the call blocking probabilities and normalizing the sum by the total arrivals to the network.

$$\bar{B} = \frac{1}{\Lambda} \sum_{(i-j)} \lambda_{i-j} B_{i-j}, \tag{3.1}$$

where Λ, the rate of the total input traffic to the network is given by

$$\Lambda = \sum_{(i-j)} \lambda_{i-j}. \tag{3.2}$$

Arrival rates λ_{i-j} are known quantities. Under the previously stated assumption of independent link blocking probabilities, B_{i-j} can be expressed in terms of B_{ij} and B_{ij}^R. A call $(i-j)$ is blocked from the network only when the direct link and the possible alternate routes are busy. The alternate routes are busy when either or both of the links constituting that route are busy. Hence, the call blocking probability B_{i-j} is:

$$B_{i-j} = B_{ij} \prod_{m \in M_{i-j}} \left[1 - (1 - B_{im}^R)(1 - B_{mj}^R)\right]. \tag{3.3}$$

3.3 Queuing Analysis

The link blocking probabilities B_{ij} and B_{ij}^R in Equation (3.3) are derived from the birth-death process of an $M/M/m/m$ queuing model [15]. Let λ and v denote the arrival rates of new and overflow calls, respectively. In this model a state is defined as the number of circuits that are occupied in a given link. With help of Figure 3.1 the state of a link can be described.

Figure 3.1 Representing the Reserved State

In the given link $c-r$ circuits are reserved for only the direct calls and r circuits can be used by either the direct calls or the alternately routed calls. If a new call arrives to the link or a call is serviced by the link then the state of a link changes from one to another. If the transition rate from state i_1 to state i_2 is denoted by $p(i_1, i_2)$, then

$$p(i, i+1) = \begin{cases} \lambda & (c > i \geq r) \\ \lambda + v & (r > i \geq 0) \end{cases}, \tag{3.4}$$

$$p(i, i-1) = i\mu \quad (c \geq i > 0), \tag{3.5}$$

where μ is the service rate. With $\mu = 1$ and defining

$$\gamma \doteq \lambda + v$$
$$\alpha \doteq \frac{\lambda}{\gamma}.$$

Let p_i denote the steady state probability of having i circuits occupied in a given link. The following steady state probabilities are obtained from the balance equations for a $M/M/m/m$ queuing model. Detailed derivation is found in [15].

$$p_i = \begin{cases} \frac{\gamma^i}{i!} p_0 & (0 \leq i \leq r) \\ \frac{\gamma^i (1-\alpha)^{i-r}}{i!} p_0 & (r < i \leq c) \end{cases}, \tag{3.6}$$

where p_0 is found using the flow conservation equation, $\sum_{i=0}^{c} p_i = 1$, and is given below.

$$p_0 = \left[\sum_{i=0}^{r} \frac{\gamma^i}{i!} + \sum_{i=r+1}^{c} \frac{\gamma^i (1-\alpha)^{i-r}}{i!} \right]^{-1} \tag{3.7}$$

Substituting Equation (3.7) into Equation (3.6), p_i can be solved in terms of γ, α and c. The link blocking probabilities B and B^R are then related to these parameters via Equation (3.8) and Equation (3.9) below.

$$B = p_c \tag{3.8}$$

$$B^R = \sum_{i=r+1}^{c} p_i \tag{3.9}$$

In order to solve for the link blocking probabilities B and B^R, one needs to relate the unknown link arrivals γ, and the parameter α to the known node-to-node arrivals λ, and the reservation parameter r of all the links in the network. This is done by finding the flow f_{ij} on each link with two different methods, as described in the following section.

3.4 Calculation of Flow

The flow on each link can be calculated using two different methods. Knowing the link blocking probabilities, the flow is found by first using γ and then by using λ and the overflow rate from other O-D pairs.

Method 1

For a given map (i.e., with c and r for each link specified) the flows f_{ij} of all links (i, j) can be computed from

$$f_{ij} = \gamma_{ij}(1 - B_{ij}). \tag{3.10}$$

Method 2

Alternatively, flows in each link can also be given in terms of the node-to-node traffic λ and the link blocking probabilities. Suppose, for link (i, j), the rate of accepted direct calls are denoted by u_{ij}, and the rate of accepted alternately routed calls are denoted by y_{ij}, then the flow on link (i, j) is

$$f_{ij} = u_{ij} + y_{ij}, \tag{3.11}$$

where

$$u_{ij} = \lambda_{i-j}(1 - B_{ij}). \tag{3.12}$$

With $f^k_{i'-j'}$ denoting the rate of accepted calls in link (i', j') on alternate route $R^k_{i'-j'}$, y_{ij} in Equation (3.11) is given by

$$y_{ij} = \left[\sum_{i':(i,j)\in R^i_{i'-j}} f^i_{i'-j} + \sum_{j':(i,j)\in R^j_{i-j'}} f^j_{i-j'} \right]. \tag{3.13}$$

Suppose $Q^k_{i'-j'}$ is used to denote the probability that call $(i' - j')$ is not blocked on the alternate route $R^k_{i'-j'}$ while it is blocked on all other alternate routes available for this O-D pair, then

$$f^k_{i'-j'} = \lambda_{i'-j'} B_{i'j'} Q^k_{i'-j'}, \tag{3.14}$$

where $\lambda_{i'-j'}B_{i'j'}$ is the rate of overflow of calls $(i'-j')$ from the direct link (i',j'). $Q^k_{i'-j'}$ can be given in terms of the link blocking probabilities and can be written as

$$Q^k_{i'-j'} = \left[\prod_{\substack{m \neq k \\ m \in M_{i'-j'}}} B(R^m_{i'-j'}) - \prod_{m \in M_{i'-j'}} B(R^m_{i'-j'}) \right]. \tag{3.15}$$

where $B(R^m_{i'-j'})$ is the blocking probability of the alternate route $R^m_{i'-j'}$ and this can be expressed in terms of the link blocking probabilities for alternately routed calls of the links that form this alternate route as given below.

$$B(R^m_{i'-j'}) = \left[1 - (1 - B^R_{i'm})(1 - B^R_{mj'})\right] \tag{3.16}$$

Using the initial estimate of the quantities α, link arrivals γ are obtained from external arrival rates λ. Then using Equation (3.4) through Equation (3.9) the link blocking probabilities are computed. These probabilities are used to compute the flow by using Equation (3.11) through Equation (3.16). For these flows the new link arrivals can be computed from Equation (3.10) and then a new set of α for each link from,

$$\alpha_{ij} = \frac{\lambda_{ij}}{\gamma_{ij}}. \tag{3.17}$$

The above steps are repeated until the flows found in successive iterations converge. Finally, from the resulting blocking probabilities, the desired quantity $\bar{B}$ is obtained from Equation (3.1) through Equation (3.3). This $\bar{B}$, the network block rate, is used as the cost function in determining the best map in the map generator module, as explained in the next two chapters.

CHAPTER 4
NEW MAP GENERATION USING SIMULATED ANNEALING

To improve the performance of the network, a map which will minimize the total block rate of the network must be chosen. As explained in Chapter 3, the block rate $\bar{B}$ depends on the capacity per link and on the number of circuits that can be used by alternately routed calls in a link. These two independent variables, c and r, make the solution space very large. Choosing the best map from this solution space is computationally time consuming. Therefore, a powerful optimization technique should be applied in order to find an optimal map. In this thesis, two neural network-based optimization techniques, *simulated annealing* and *mean field annealing*, are used. This chapter describes the application of simulated annealing in the map generation process. The next chapter describes how mean field annealing can be applied to speed up the map generation task.

4.1 Introduction to Simulated Annealing

First proposed by Kirkpatrik [17], simulated annealing is a stochastic optimization technique used to solve complex problems. Since then it has been applied in such diverse areas as computer-aided design of integrated circuits, image processing, code design, *etc.*

Generally, a combinatorial optimization problem consists of a set S of configurations or solutions and a cost function C, which determines the cost $C(s)$ for each configuration. An iterative improvement scheme known as local search could be performed to find the minimum cost. During a local search process an initial solution s_i is given and then a new solution s_n is proposed at random. If the new cost $C(s_n)$ is less than the current cost $C(s_c)$, which is same as $C(s_i)$, then the new configuration is accepted, and the new solution s_n becomes the current solution s_c. If the new cost is larger than the current cost then a new solution is proposed, again at random. This procedure continues until the minimum solution is found. Unfortunately, a local search may get stuck at the local minima and may escape the global minima. To alleviate the problem of getting trapped at local minima, simulated annealing occasionally allows "uphill moves" to solutions of higher cost according to the so-called Metropolis criterion [14].

The simulated annealing algorithm is based on the analogy between the simulation of the annealing of solids and the problem of solving large combinatorial optimization problems. For this reason the algorithm became known as "simulated annealing." The simulated annealing process consists of first "melting" the system being optimized at an effective high temperature, then lowering the temperature gradually until the system "freezes" and no further changes occur [17]. At each temperature the system is allowed to reach *thermal equilibrium*, characterized by a probability of being in a state s, $P(s)$. This probability is proportional to the Boltzman probability factor and can be written as:

$$P(s) \propto e^{-E(s)/kT}, \tag{4.1}$$

where $E(s)$ is the energy of the system at state s, k is the Boltzman constant, and T denotes the temperature at which thermal equilibrium is maintained. Now, apply a small perturbation to the the current state s_c, of the system and obtain a new state s_n. Denote p, the ratio between the probabilities of finding the system in two states s_c and s_n, as:

$$p = \frac{P(s_n)}{P(s_c)} = e^{-\frac{[E(s_n)-E(s_c)]}{kT}} \tag{4.2}$$

If the difference in energy, ΔE, between the current state and the slightly perturbed one is negative (i.e., the new state has lower energy than the current state), then the process is continued with the new state. If $\Delta E > 0$, then the probability of acceptance of the perturbed state is given by Equation (4.2). This acceptance rule for new states is referred to as the *Metropolis criterion*. Following this criterion, the system eventually evolves into a state of thermal equilibrium. Detailed discussions on the convergence of simulated annealing type algorithms and other similar algorithms can be found in [18]–[20].

4.2 Application of Simulated Annealing in Map Generation

In this section an optimal map generation process using simulated annealing is described. Before applying the algorithm, a cost function, the cooling schedule and a generation mechanism (or, equivalently, a *neighborhood structure*) must be defined. These are described below in detail.

4.2.1 Cost Function

In this work the total block rate of the network is chosen as the cost. An expression for finding the total block rate $\bar{B}$ is given in Equation (3.1). Since the block rate depends on the capacity of the link and the reservation parameter of each link, the energy function can be written as follows:

$$E(s) = \bar{B}(\vec{c}, \vec{r}), \tag{4.3}$$

where

$$\vec{c} = \{c_{ij} : (i,j) \in all\ links\}$$

and

$$\vec{r} = \{r_{ij} : (i,j) \in all\ links\}.$$

4.2.2 Cooling Schedule

Once the cost function is determined the following parameters should be specified:

1. the initial value of temperature T_0
2. the final value of temperature T_f (stopping criterion)
3. the number of transitions at each temperature

4. a rule for changing the current value of the control parameter T_j into the next one, T_{j+1}.

These particular parameters are referred to as the *cooling schedule* [21] and the method to choose these parameters are discussed below.

1. Initial Temperature

The initial value of temperature T_0 is determined in such a way that virtually all the transitions are accepted. If a value is too high then all the transitions will be accepted most of the time. If the value is too low then the solution may not be optimal. In order to avoid these problems a temperature must be chosen so that it is in the critical point where number of accepted maps is about to decrease significantly. This particular temperature is referred to as the *critical temperature.*

2. Stopping Criterion

A stopping criterion is usually based on the argument that execution of the algorithm can be terminated if the improvement in cost (that which would be expected in the case of continuing execution of the algorithm) is small [17]. Therefore, if the maps which are generated consecutively do not vary significantly in cost then the algorithm is terminated and the final map is considered the "best" map.

3. Number of Transitions at Each Temperature

The generation of new states is stopped and the next temperature level is tried when either the number of accepts reach a certain level (L_{accept}) or after generating a specific number (L_{trials}) of new states. Due to the variations in the experiments, different values of L_{accept} and L_{trials} are tried in the simulations.

4. Temperature Updating Rule

The difference between two temperatures and the value of the temperature relative to the difference in the possible range of costs of two states in the solution space are two parameters need to be given close attention in determining the rule for decreasing the temperature. Since the size of the network affects ΔE, the control parameter is related to the total capacity, C of the network. Also, in order to "fine tune" the solution near the optimal solution and in the beginning the rule of decreasing the temperature can be expressed as follows:

$$T = a\frac{C}{\ln(j)}, \tag{4.4}$$

where a is a constant and j is the iteration index which is incremented linearly to produce the desired temperature schedule. For experimental purposes and to get a proper annealing schedule, the constant a is varied.

4.2.3 Neighborhood Structure

A neighborhood structure is necessary when implementing the simulated annealing algorithm. Different neighborhoods are defined for each of the cases considered during the simulations and they are explained below.

Case 1: Varying Reservation Parameter of the Links

If a particular link has c circuits, then the number of circuits that can be reserved varies from zero to c. Therefore, when finding a neighbor, a random number in $[0,c]$ is chosen, and this number is assigned as the reservation parameter of that link.

Case 2: Varying Link Capacities Only

Here, one link is chosen at random, and one circuit is deducted from that link and assigned to another link which will be benifitted by this exchange. In order to be practical and to obtain reasonable results, a lower bound capacity and an upper bound capacity are assigned to each link. These conditions will avoid the state where some of the links having almost no circuits and some circuits having very large number of circuits. In addition, it will avoid unnecessary computations.

Case 3: Varying Both Link Capacities and Reservation Parameter

Number of combinations in this case are very large. In order to be practical and to avoid unnecessary calculations similar control measure described for Case 2 is used.

The same approach is taken here to trade the circuits, but in addition, a reservation parameter is also assigned in random selected between zero and the link capacity.

4.2.4 Application of the Algorithm

Once the cost function, the cooling schedule, and the neighborhood structure are defined, a combinatorial problem such as the "best" map generation problem can be solved by applying the simulated annealing algorithm, as shown in Figure 4.1. Note that the acceptance criterion is implemented by drawing random numbers from a uniform distribution on [0,1] and compared with $exp(-\frac{\Delta E}{T})$.

Thus, using the simulated annealing algorithm as described above, an optimal map could be found which improves the performance of the network. Simulation results are given in Chapter 6.

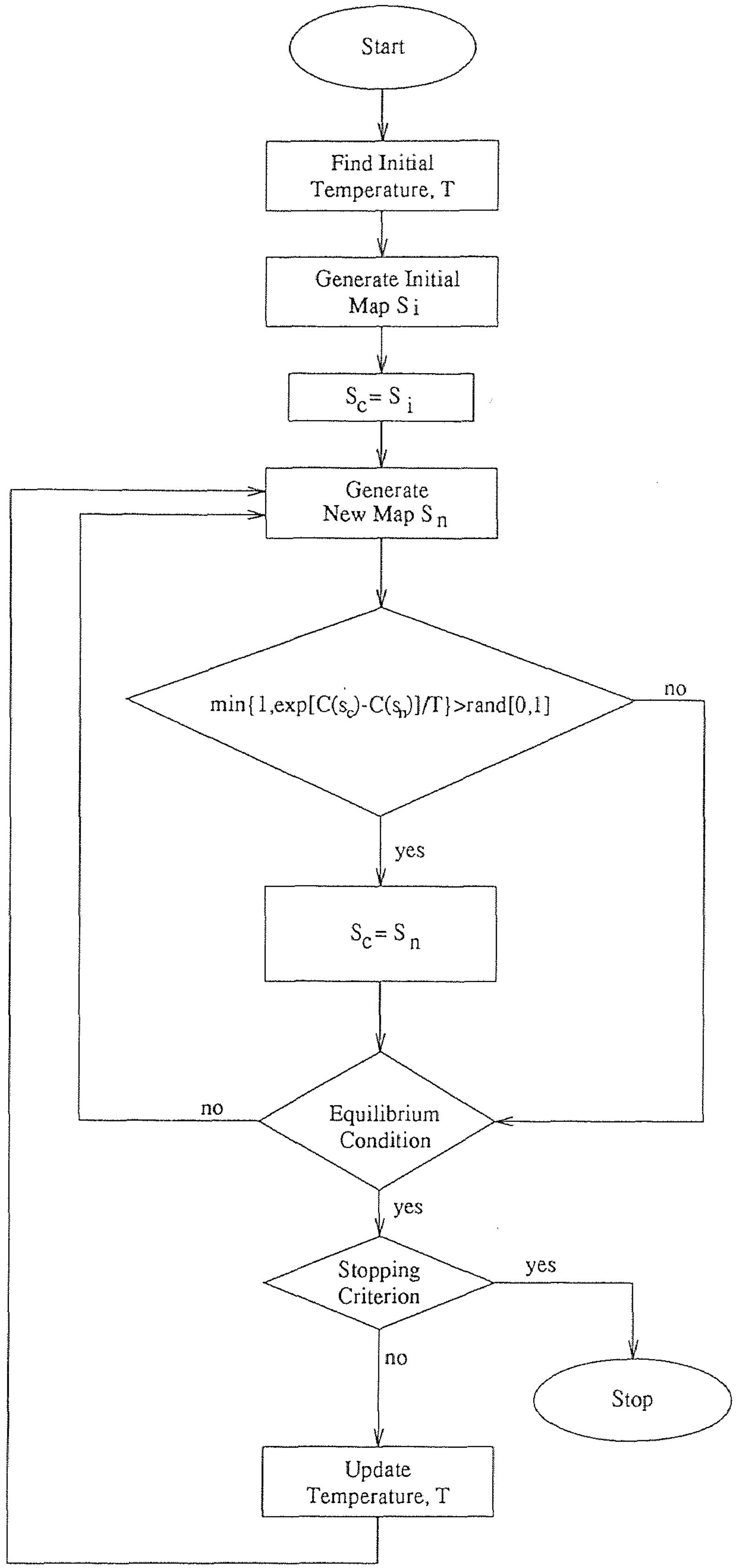

Figure 4.1 Simulated Annealing Algorithm

CHAPTER 5

NEW MAP GENERATION USING MEAN FIELD ANNEALING

Optimization by simulated annealing sometimes pose some problems. Choosing a cost function suitable for simulated annealing may be difficult for certain problems. Furthermore solving by simulated annealing may require very large computing time. These problems can be solved by using mean field annealing (MFA). By incorporating simulated annealing with Hopfield energy function the Mean Field Theory is derived [22]–[25]. In mean field annealing, two operations used in simulated annealing are still needed. These operations are a thermostatic operation which schedules the decrease of temperature and a random relaxation process which searches for the equilibrium solution at each temperature. However, the relaxation process in searching for the equilibrium solution has been replaced by searching for the mean value of the solution. Equilibrium can be reached faster by using the mean [25], and thus mean field annealing speeds up the computational process.

5.1 Introduction to Mean Field Annealing

This section gives a brief introduction to mean field annealing. Detailed description and the derivation of mean field equations can be found in [22]–[24]. The relaxation in both simulated annealing and mean field annealing follows a Boltzman distribution [17] which is given by

$$P(S) = \frac{1}{Z} e^{\frac{-E(S)}{T}}, \tag{5.1}$$

where S is any one of the possible configurations specified by the corresponding neuron set, $E(S)$ is the energy of the corresponding configuration, T is the parameter called temperature, and Z is the partition function given by

$$Z = \sum_{(S)} e^{\frac{-E(S)}{T}}. \tag{5.2}$$

In mean field theory, instead of concerning the neuron variables directly, consider the means of neurons by defining:

$$V_i = < S_i > . \tag{5.3}$$

Let the value of neuron S_i be "1" when the neuron S_i is on and let the value of neuron S_i be "0" when the neuron S_i is "off". Now Equation (5.3) becomes

$$\begin{aligned} V_i &= (1).P(S_i = 1) + (0).P(S_i = 0) \\ V_i &= P(S_i = 1), \end{aligned} \tag{5.4}$$

where $P(S_i = 1)$ and $P(S_i = 0)$ are the probabilities for $S_i = 1$ and $S_i = 0$, respectively, and V is the mean configuration corresponding to S. Thus, in terms of mean field variable Equation (5.1) becomes

$$P(V) = \frac{1}{Z} e^{\frac{-E(V)}{T}}. \tag{5.5}$$

Now define the local field,

$$h_i = -\frac{\partial E}{\partial S_i} \tag{5.6}$$

and

$$S_i = \begin{cases} 1 & \text{if } h_i > 0 \\ 0 & \text{if } h_i < 0 \end{cases} . \tag{5.7}$$

The probability of a particular neuron S_i to be "on" and "off" is given by the following two equations.

$$P(S_i = 1) = \frac{e^{h_i/T}}{e^{-h_i/T} + e^{h_i/T}} \tag{5.8}$$

$$P(S_i = 0) = \frac{e^{-h_i/T}}{e^{-h_i/T} + e^{h_i/T}} \tag{5.9}$$

Mean field theory approximation is to approximate the local field h_i by its thermal average (mean field). Therefore Equation (5.8) and Equation (5.9) becomes

$$P(S_i = 1) = \frac{e^{h_i^{MFT}/T}}{e^{-h_i^{MFT}/T} + e^{h_i^{MFT}/T}} \tag{5.10}$$

$$P(S_i = 0) = \frac{e^{-h_i^{MFT}/T}}{e^{-h_i^{MFT}/T} + e^{h_i^{MFT}/T}}, \tag{5.11}$$

where

$$h_i^{MFT} = < h_i > = < -\frac{\partial E}{\partial V_i} > . \tag{5.12}$$

Now, combining Equation (5.4), Equation (5.10), Equation (5.11) and Equation (5.12) an expression for V_i is obtained as follows,

$$V_i = \frac{e^{h_i^{MFT}/T}}{e^{-h_i^{MFT}/T} + e^{h_i^{MFT}/T}}, \tag{5.13}$$

or,

$$V_i = \frac{1}{2}[1 + \tanh(h_i^{MFT}/2T)]. \tag{5.14}$$

The relationship between V_i and $h_i^{MFT}/2T$ is depicted in Figure 5.1.

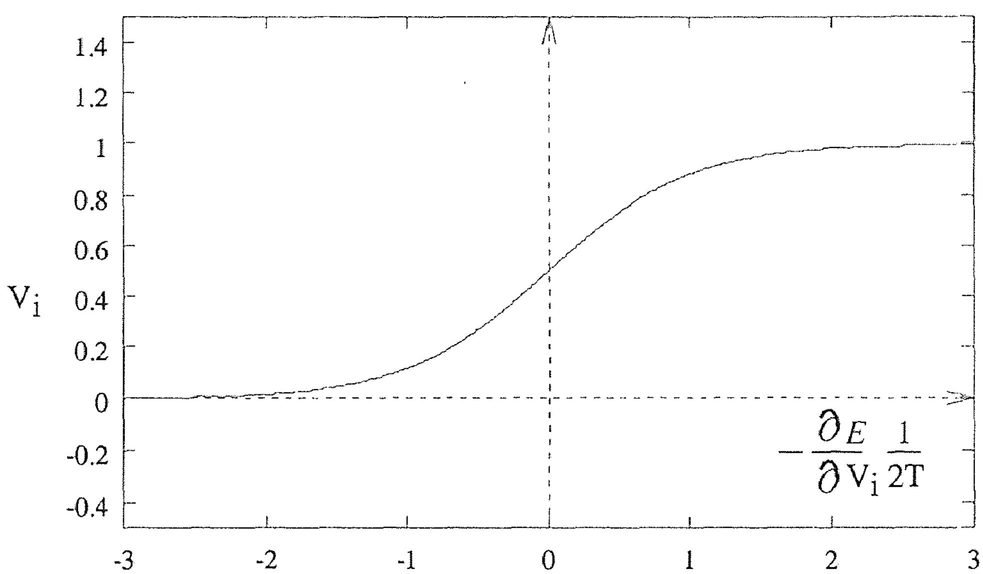

Figure 5.1 Depicting the Function of the MFT Equation

5.2 Application of Mean Field Annealing to Map Generation

In order to solve the problem by mean field annealing, the problem should be mapped into a neural network and then an energy function should be formulated. Since each map differ from each other in terms of the link capacities and the reservation parameter of each link, the energy function should be able to incorporate all possible combinations. For experimental purposes three different cases of map generation is considered. They are:

1. Reservation parameter of each link is varied while keeping the capacity of each link constant,
2. Capacity of each link is varied while keeping the reservation parameter constant,
3. Both capacity and the reservation parameter are varied for all links.

In all cases, the constraint is that the total network capacity is fixed to a constant value. In order to represent all these cases, an encoding method to denote the neurons is necessary. This encoding scheme is developed in the following section.

5.2.1 Neuron Encoding

A neuron is denoted by S_{ijr} or S_{ijc} or S_{ijcr} depending on the case considered in the analysis. This scheme can be defined mathematically as follows,

Case 1: Varying Reservation Parameter Only

$$S_{ijr} = \begin{cases} 1 & \text{if link } (i,j) \text{ has } r \text{ circuits available for alternate routes} \\ 0 & \text{otherwise} \end{cases} \tag{5.15}$$

Case 2: Varying Link Capacities Only

$$S_{ijc} = \begin{cases} 1 & \text{if link } (i,j) \text{ has } c \text{ circuits} \\ 0 & \text{otherwise} \end{cases} \tag{5.16}$$

Case 3: Varying Both Capacity and Reservation Parameter

$$S_{ijcr} = \begin{cases} 1 & \text{if link } (i,j) \text{ has } c \text{ circuits and} \\ & r \text{ of them are available for alternate routes} \\ 0 & \text{otherwise} \end{cases} \tag{5.17}$$

5.2.2 Associative Matrix N

Since the nodes in the networks may not be fully connected, some of the neurons are always fixed to zero. Most of the time the number of neurons, which are "off," are very large. Therefore during the computational process the calculations for those clamped neurons could be avoided, and thus speeding up the computational process. To implement this clamping technique into the neural network, an associative matrix **N** should be defined [26]. If there are N_N nodes, then there are N_N rows and N_N columns in matrix **N** as shown below

$$\mathbf{N} = \begin{bmatrix} n_{11} & n_{12} & \cdots & n_{1N_N} \\ n_{21} & n_{22} & \cdots & n_{2N_N} \\ \vdots & \vdots & \vdots & \vdots \\ n_{N_N1} & n_{N_N2} & \cdots & n_{N_NN_N} \end{bmatrix} \tag{5.18}$$

where n_{ij} takes on either 0 or 1 according to:

$$n_{ij} = \begin{cases} 1 & \text{if there is a link between node } i \text{ and node } j \\ 0 & \text{otherwise} \end{cases} \tag{5.19}$$

5.2.3 Formulation of Energy Function

In optimization problems, an objective function, which is to be optimized should be formulated. Since Hopfield energy function can be used for optimization problems, in this work the energy function will have the form of Hopfield energy function [27]. In constrained optimization problems, the energy function has two terms: the cost term and the constraint term. In the problem considered in thesis, the cost term is the total block rate of the network and the constraint term is the penalty added to the cost for violating the constraints. Therefore the energy function could be written as follows,

$$E = \alpha \times \text{"cost"} + \beta \times \text{"constraint1"} + \gamma \times \text{"constraint2"} + \cdots \tag{5.20}$$

where α, β, γ are the Lagrange parameters. For experimental purposes three different cases of optimization: the effect of changing the parameter r only, the effect of changing the number of circuits per link, c, and the effect of changing both reservation and the capacity of each link, are considered. Therefore three different energy functions should be developed.

Case 1: Varying the Reservation Parameter Only

In this case only the reservation parameter is changed. The capacity of each link is fixed at constant value. The "cost" term in Equation (5.20) is the total block rate of the network. Combining Equation (3.1) and Equation (3.3), an expression for the cost term E_0 is obtained:

$$E_{DB} = \sum_r B_{ijr} S_{ijr} \tag{5.21}$$

$$E_{AB} = \prod_{m \in M_{i-j}} \left[1 - (1 - \sum_r B^R_{imr} S_{imr})(1 - \sum_r B^R_{mjr} S_{mjr}) \right] \tag{5.22}$$

$$E_0 = \frac{1}{\Lambda} \sum_i \sum_j \lambda_{i-j} E_{DB} E_{AB} n_{ij}, \tag{5.23}$$

where

E_{DB}: the energy term corresponding to the direct blocking,

E_{AB}: the energy term corresponding to the alternate blocking,

B_{ijr}: the probability that a direct call is blocked in the link (i,j) with r circuits available for alternately routed calls,

B^R_{ijr}: the probability that an alternately routed call is blocked in the link (i,j) with r circuits available for alternately routed calls.

Constraint terms are defined as follows.

1. Each link is restricted to have only one particular reservation parameter. If more than one reservation parameter is assigned to a link then a penalty term is imposed.

$$E_1 = \sum_i \sum_j \sum_r \sum_{r' \neq r} S_{ijr} S_{ijr'} \tag{5.24}$$

2. The total number of neurons that are "on" must be equal to the number of links, N_L, in the network. Thus this constraint avoids the situation where all the neurons are "off."

$$E_2 = (\sum_i \sum_j \sum_r S_{ijr} - N_L)^2 \qquad (5.25)$$

The total energy is the sum of the cost and the constraints, and it can be written as:

$$E = \alpha \times E_0 + \beta \times E_1 + \gamma \times E_2, \qquad (5.26)$$

where α, β, γ, are the Lagrange parameters.

Case 2: Varying the Capacity Only

The case where varying only the capacity of each link is considered here. In this case the number of circuits that can be used for alternately routed calls are fixed to 20% of the capacity of each link. Similar to Case 1 the energy function can be developed as follows.

$$E_{DB} = \sum_c B_{ijc} S_{ijc}, \qquad (5.27)$$

$$E_{AB} = \prod_{m \in M_{i-j}} \left[1 - (1 - \sum_c B^R_{imc} S_{imc})(1 - \sum_c B^R_{mjc} S_{mjc}) \right], \qquad (5.28)$$

$$E_0 = \frac{1}{\Lambda} \sum_i \sum_j \lambda_{i-j} E_{DB} E_{AB} n_{ij}, \qquad (5.29)$$

where

E_{DB}: the energy term corresponding to the direct blocking,

E_{AB}: the energy term corresponding to the alternate blocking,

B_{ijc}: the probability that a direct call is blocked in the link (i, j) with c circuits,

B^R_{ijc}: the probability that an alternately routed call is blocked in the link (i, j) with c circuits.

Constraint terms are defined as follows.

1. Each link is restricted to have only one particular capacity. If more than one value of capacity is assigned to a link then a penalty term is imposed.

$$E_1 = \sum_i \sum_j \sum_c \sum_{c' \neq c} S_{ijc} S_{ijc'} \tag{5.30}$$

2. The total number of circuits in the network is fixed. If the allocated circuits do not equal the total available circuits then a penalty term will be added. This can be represented as follows,

$$E_2 = (\sum_i \sum_j \sum_c c S_{ijc} - C)^2. \tag{5.31}$$

3. Finally the total number of neurons that are "on" must be equal the number of links in the network.

$$E_3 = (\sum_i \sum_j \sum_c S_{ijc} - N_L)^2 \tag{5.32}$$

The total energy is the sum of the cost and the constraints and, it is written as:

$$E = \alpha \times E_0 + \beta \times E_1 + \gamma \times E_2 + \kappa \times E_3. \tag{5.33}$$

where α, β, γ and κ are the Lagrange parameters.

Case 3: Varying Both Capacity and Reservation

When changing the reservation parameter and the capacity of each link the energy function can be formulated as follows:

$$E_{DB} = \sum_c \sum_r B_{ijcr} S_{ijcr}, \tag{5.34}$$

$$E_{AB} = \prod_{m \in M_{i-j}} \left[1 - (1 - \sum_c \sum_r B^R_{imcr} S_{imcr})(1 - \sum_c \sum_r B^R_{mjcr} S_{mjcr}) \right], \tag{5.35}$$

$$E_0 = \frac{1}{\Lambda} \sum_i \sum_j \lambda_{i-j} E_{DB} E_{AB} n_{ij}, \tag{5.36}$$

where

E_{DB}: the energy term corresponding to the direct blocking,

E_{AB}: the energy term corresponding to the alternate blocking,

B_{ijcr}: the probability that a direct call is blocked in the link (i, j) with c circuits in which r circuits are available for alternately routed calls,

B^R_{ijcr}: the probability that an alternately routed call is blocked in the link (i, j) with c circuits in which r circuits are available for alternately routed calls.

Constraint terms are defined as follows.

1. Each link is restricted to have only one particular capacity. If more than one value of capacity is assigned to a link then a penalty term is imposed.

$$E_1 = \sum_i \sum_j \sum_c \sum_r \sum_{c' \neq c} S_{ijcr} S_{ijc'r} \tag{5.37}$$

2. Each link is restricted to have only one particular reservation parameter. If more than one reservation parameter is assigned to a link then a penalty term will be imposed.

$$E_2 = \sum_i \sum_j \sum_c \sum_r \sum_{r' \neq r} S_{ijcr} S_{ijcr'} \tag{5.38}$$

3. The total number of circuits in the network is fixed. If the allocated circuits do not equal the total available circuits then a penalty term will be added. This can be represented as follows,

$$E_3 = (\sum_i \sum_j \sum_c \sum_r c S_{ijcr} - C)^2 \tag{5.39}$$

4. The total number of neurons that are "on" must be equal to the number of links in the network. Thus this constraint avoids the situation where all the neurons are "off."

$$E_4 = (\sum_i \sum_j \sum_c \sum_r S_{ijcr} - N_L)^2 \tag{5.40}$$

The total energy is the sum of the cost and the constraints, and written as follows,

$$E = \alpha \times E_0 + \beta \times E_1 + \gamma \times E_2 + \kappa \times E_3 + \eta \times E_4. \tag{5.41}$$

where α, β, γ, κ and η are the Lagrange parameters.

5.2.4 Evaluation of Thermal Average

For all three cases, the energy functions should be evaluated in terms mean field variables. In the mean field domain, the energy function becomes the function of the mean field variables.

Case 1: Varying Reservation Parameter Only

$$E_{DB} = \sum_r B_{ijr} V_{ijr} \tag{5.42}$$

$$E_{AB} = \prod_{m \in M_{i-j}} \left[1 - (1 - \sum_r B^R_{imr} V_{imr})(1 - \sum_c \sum_r B^R_{mjr} V_{mjr}) \right] \tag{5.43}$$

$$E_0 = \frac{1}{\Lambda} \sum_i \sum_j \lambda_{i-j} E_{DB} E_{AB} n_{ij} \tag{5.44}$$

$$E_1 = \sum_i \sum_j \sum_r \sum_{r' \neq r} V_{ijr} V_{ijr'} \tag{5.45}$$

$$E_2 = (\sum_i \sum_j \sum_r V_{ijr} - N_L)^2 \tag{5.46}$$

Now thermal average of the local field, h^{MFT}_{ijr} can be evaluated as follows:

$$h^{MFT}_{ijr} = < h_{ijr} > = < -\frac{\partial E}{\partial V_{ijr}} > \tag{5.47}$$

In order to find $\frac{\partial E}{\partial V_{ijr}}$, derivatives of each component of the energy function are obtained as follows:

$$\frac{\partial E_0}{\partial V_{ijr}} = \frac{1}{\Lambda} \lambda_{i-j} B_{ijr} E_{AB} n_{ij} \tag{5.48}$$

$$\frac{\partial E_1}{\partial V_{ijr}} = \sum_{r' \neq r} V_{ijr'} \tag{5.49}$$

$$\frac{\partial E_2}{\partial V_{ijr}} = 2(\sum_i \sum_j \sum_r V_{ijr} - N_L). \tag{5.50}$$

Thus, $\frac{\partial E}{\partial V_{ijr}}$ is expressed as:

$$\frac{\partial E}{\partial V_{ijr}} = \alpha \frac{\partial E_0}{\partial V_{ijr}} + \beta \frac{\partial E_1}{\partial V_{ijr}} + \gamma \frac{\partial E_2}{\partial V_{ijr}}. \tag{5.51}$$

Case 2: Varying Link Capacities Only

$$E_{DB} = \sum_c B_{ijc} V_{ijc} \tag{5.52}$$

$$E_{AB} = \prod_{m \in M_{i-j}} \left[1 - (1 - \sum_c B^R_{imc} V_{imc})(1 - \sum_c B^R_{mjc} V_{mjc}) \right] \tag{5.53}$$

$$E_0 = \frac{1}{\Lambda} \sum_i \sum_j \lambda_{i-j} E_{DB} E_{AB} n_{ij} \tag{5.54}$$

$$E_1 = \sum_i \sum_j \sum_c \sum_{c' \neq c} V_{ijc} V_{ijc'} \tag{5.55}$$

$$E_2 = (\sum_i \sum_j \sum_c c V_{ijc} - C)^2 \tag{5.56}$$

$$E_3 = (\sum_i \sum_j \sum_c V_{ijc} - N_L)^2 \tag{5.57}$$

Now thermal average of the local field, h^{MFT}_{ijc} can be evaluated as follows:

$$h^{MFT}_{ijc} = \langle h_{ijc} \rangle = \langle -\frac{\partial E}{\partial V_{ijc}} \rangle \tag{5.58}$$

In order to find $\frac{\partial E}{\partial V_{ijc}}$, derivatives of each component of the energy function are obtained as follows,

$$\frac{\partial E_0}{\partial V_{ijc}} = \frac{1}{\Lambda} \lambda_{i-j} B_{ijc} E_{AB} n_{ij} \tag{5.59}$$

$$\frac{\partial E_1}{\partial V_{ijc}} = \sum_{c' \neq c} V_{ijc'} \tag{5.60}$$

$$\frac{\partial E_2}{\partial V_{ijc}} = 2(\sum_i \sum_j \sum_c c V_{ijc} - C)c \tag{5.61}$$

$$\frac{\partial E_3}{\partial V_{ijc}} = 2(\sum_i \sum_j \sum_c V_{ijc} - N_L) \tag{5.62}$$

Thus, $\frac{\partial E}{\partial V_{ijc}}$ is expressed as,

$$\frac{\partial E}{\partial V_{ijc}} = \alpha \frac{\partial E_0}{\partial V_{ijc}} + \beta \frac{\partial E_1}{\partial V_{ijc}} + \gamma \frac{\partial E_2}{\partial V_{ijc}} + \kappa \frac{\partial E_3}{\partial V_{ijc}} \tag{5.63}$$

Case 3: Varying Both Link Capacities and Reservation Parameter

$$E_{DB} = \sum_c \sum_r B_{ijcr} V_{ijcr} \tag{5.64}$$

$$E_{AB} = \prod_{m \in M_{i-j}} \left[1 - (1 - \sum_c \sum_r B^R_{imcr} V_{imcr})(1 - \sum_c \sum_r B^R_{mjcr} V_{mjcr}) \right] \tag{5.65}$$

$$E_0 = \frac{1}{\Lambda} \sum_i \sum_j \lambda_{i-j} E_{DB} E_{AB} n_{ij} \tag{5.66}$$

$$E_1 = \sum_i \sum_j \sum_c \sum_r \sum_{c' \neq c} V_{ijcr} V_{ijc'r} \tag{5.67}$$

$$E_2 = \sum_i \sum_j \sum_c \sum_r \sum_{r' \neq r} V_{ijcr} V_{ijcr'} \tag{5.68}$$

$$E_3 = (\sum_i \sum_j \sum_c \sum_r c V_{ijcr} - C)^2 \tag{5.69}$$

$$E_4 = (\sum_i \sum_j \sum_c \sum_r V_{ijcr} - N_L)^2 \tag{5.70}$$

Now thermal average of the local field, h^{MFT}_{ijcr} can be evaluated as follows:

$$h^{MFT}_{ijcr} = < h_{ijcr} > = < -\frac{\partial E}{\partial V_{ijcr}} > . \tag{5.71}$$

Again, derivatives of each component of the energy function are:

$$\frac{\partial E_0}{\partial V_{ijcr}} = \frac{1}{\Lambda} \lambda_{i-j} B_{ijcr} E_{AB} n_{ij} \tag{5.72}$$

$$\frac{\partial E_1}{\partial V_{ijcr}} = \sum_{c' \neq c} V_{ijc'r} \tag{5.73}$$

$$\frac{\partial E_2}{\partial V_{ijcr}} = \sum_{r' \neq r} V_{ijcr'} \tag{5.74}$$

$$\frac{\partial E_3}{\partial V_{ijcr}} = 2(\sum_i \sum_j \sum_c \sum_r c V_{ijcr} - C)c \tag{5.75}$$

$$\frac{\partial E_4}{\partial V_{ijcr}} = 2(\sum_i \sum_j \sum_c \sum_r V_{ijcr} - N_L) \tag{5.76}$$

Thus, $\frac{\partial E}{\partial V_{ijcr}}$ is expressed as

$$\frac{\partial E}{\partial V_{ijcr}} = \alpha \frac{\partial E_0}{\partial V_{ijcr}} + \beta \frac{\partial E_1}{\partial V_{ijcr}} + \gamma \frac{\partial E_2}{\partial V_{ijcr}} + \kappa \frac{\partial E_3}{\partial V_{ijcr}} + \eta \frac{\partial E_4}{\partial V_{ijcr}} \tag{5.77}$$

5.2.5 Evaluation of Lagrange Parameters

The selection of the Lagrange parameters are critical in the annealing process. These parameters must be selected carefully in order to guarantee convergence. If the parameters are selected poorly the process may not approach to a global minimum or divergence may occur. To avoid these problems an approximate method to find these Lagrange parameters is proposed for first two cases and similar procedure can be applied for the last case.

Case1: Varying the Reservation Parameter Only

In Equation (5.51) the parameter α governs the balance between "cost" and the "constraint" terms and the constants β and γ determine the importance of the constraints. Since only one value reservation parameter should be assigned to a link, energy term E_1 should be weighted heavier than the others Thus we have

$$\beta > \alpha, \gamma. \tag{5.78}$$

In order to find a relationship between α and γ assume that constraint (1) is satisfied, (i.e., $E_1 = 0$). This leads to the fact that $\frac{\partial E_1}{\partial V_{ijr}} = 0$. In Equation (5.51) $\frac{\partial E_0}{\partial V_{ijr}}$ is always positive. But $\frac{\partial E_2}{\partial V_{ijr}}$ may be positive or negative depending on whether the constraint on number of neurons "on" is satisfied. If the total number of neurons having "1" is more than the number of links then the term $\frac{\partial E_2}{\partial V_{ijr}}$ will be positive and if the total number of neurons having "1" is less than the number of links then the term $\frac{\partial E_2}{\partial V_{ijr}}$ will be negative.

When no specific reservation parameter is assigned to a link, a neuron must be turned "on." As shown in Figure 5.1 in order to turn "on" a neuron $\frac{\partial E}{\partial V_{ijr}} < 0$. This leads to the following:

$$\alpha \frac{\partial E_0}{\partial V_{ijr}} + \gamma \frac{\partial E_3}{\partial V_{ijr}} < 0 \tag{5.79}$$

$$\alpha\left(\frac{\lambda_{i-j}}{\Lambda} B_{ijr} E_{AB}\right) + 2\gamma\left(\sum_i \sum_j \sum_r V_{ijr} - N_L\right) < 0 \tag{5.80}$$

Since there are not enough neurons which are "on" than required,

$$(\sum_i \sum_j \sum_r V_{ijr} - N_L) < 0 \tag{5.81}$$

Thus, Equation (5.80) becomes:

$$\alpha(\frac{\lambda_{i-j}}{\Lambda} B_{ijr} E_{AB}) + 2\gamma(\sum_i \sum_j \sum_r V_{ijr} - N_L) < 0. \tag{5.82}$$

Furthermore maximum values of B_{ijr} and E_{AB} are "1" and the fraction $\frac{\lambda_{i-j}}{\Lambda}$ will never be more than "1." If only one neuron is needed to be turned "on" then $(\sum_i \sum_j \sum_r V_{ijr} - N_L) = -1$. Using these conditions the relationship between α and γ can be written as

$$\alpha - 2\gamma < 0. \tag{5.83}$$

Therefore,

$$\gamma > \frac{\alpha}{2}. \tag{5.84}$$

When more than one specific reservation parameter is assigned to a link, a neuron must be turned "off." In order to turn "off" a neuron $\frac{\partial E}{\partial V_{ijr}} > 0$. Then,

$$\alpha \frac{\partial E_0}{\partial V_{ijr}} + \gamma \frac{\partial E_3}{\partial V_{ijr}} > 0 \tag{5.85}$$

$$\alpha(\frac{\lambda_{i-j}}{\Lambda} B_{ijr} E_{AB}) + 2\gamma(\sum_i \sum_j \sum_r V_{ijr} - N_L) > 0 \tag{5.86}$$

Since there are more neurons which are "on" than required,

$$(\sum_i \sum_j \sum_r V_{ijr} - N_L) > 0 \tag{5.87}$$

Thus, Equation (5.86) becomes:

$$\alpha(\frac{\lambda_{i-j}}{\Lambda} B_{ijr} E_{AB}) + 2\gamma > 0. \tag{5.88}$$

Since both terms in the above equation are positive as long as α and γ are positive the condition is satisfied.

Incorporating all these relationships as a rule of thumb the following rule is obtained.

$$\beta > \gamma > \frac{\alpha}{2} \tag{5.89}$$

Case2: Varying the Link Capacities Only

Referring to Equation (5.63) a relationship between α, β, γ and κ should be developed. Since only one value of capacity should be assigned to a particular link, and the total capacity, C, of the network is fixed, energy terms E_1 and E_2 should be weighted heavier than the others. For convenience, let β and γ be equal. Thus we have

$$\beta = \gamma > \alpha, \kappa. \tag{5.90}$$

Now, a relationship between α and κ can be obtained using the procedure outlined in the previous case. Here, assume that constraints on the capacities are satisfied (i.e., one particular capacity per link and total capacity). Then energy terms E_1 and E_2 are equal to zero. In Equation (5.63) $\frac{\partial E_0}{\partial V_{ijc}}$ is always positive. But $\frac{\partial E_3}{\partial V_{ijc}}$ may be positive or negative depending on whether the constraint on number of neurons "on" is satisfied. If the total number of neurons having "1" is more than the number of links then the term $\frac{\partial E_3}{\partial V_{ijc}}$ will be positive and if the total number of neurons having "1" is less than the number of links then the term $\frac{\partial E_3}{\partial V_{ijc}}$ will be negative.

When no specific capacity is assigned to a link, a neuron must be turned "on". As shown in Figure 5.1 in order to turn "on" a neuron $\frac{\partial E}{\partial V_{ijc}} < 0$. This leads to the following:

$$\alpha \frac{\partial E_0}{\partial V_{ijc}} + \kappa \frac{\partial E_3}{\partial V_{ijc}} < 0 \tag{5.91}$$

$$\alpha\left(\frac{\lambda_{i-j}}{\Lambda} B_{ijc} E_{AB}\right) + 2\kappa\left(\sum_i \sum_j \sum_c V_{ijc} - N_L\right) < 0 \tag{5.92}$$

Since there are not enough neurons which are "on" than required,

$$\left(\sum_i \sum_j \sum_c V_{ijc} - N_L\right) < 0 \tag{5.93}$$

Thus, Equation (5.92) becomes:

$$\alpha(\frac{\lambda_{i-j}}{\Lambda}B_{ijc}E_{AB}) + 2\kappa(\sum_i\sum_j\sum_c V_{ijc} - N_L) < 0. \tag{5.94}$$

Furthermore maximum values of B_{ijc} and E_{AB} are "1" and the fraction $\frac{\lambda_{i-j}}{\Lambda}$ will never be more than "1." If only one neuron is needed to be turned "on" then $(\sum_i\sum_j\sum_c V_{ijc} - N_L) = -1$. Using these conditions the relationship between α and κ is obtained as

$$\alpha - 2\kappa < 0 \tag{5.95}$$

Therefore,

$$\kappa > \frac{\alpha}{2} \tag{5.96}$$

When more than one specific capacity is assigned to a link, a neuron must be turned "off." In order to turn "off" a neuron $\frac{\partial E}{\partial V_{ijc}} > 0$. Then,

$$\alpha\frac{\partial E_0}{\partial V_{ijc}} + \kappa\frac{\partial E_3}{\partial V_{ijc}} > 0 \tag{5.97}$$

$$\alpha(\frac{\lambda_{i-j}}{\Lambda}B_{ijc}E_{AB}) + 2\kappa(\sum_i\sum_j\sum_c V_{ijc} - N_L) > 0 \tag{5.98}$$

Since there more neurons which are "on" than required,

$$(\sum_i\sum_j\sum_c V_{ijc} - N_L) > 0 \tag{5.99}$$

Thus, Equation (5.98) becomes:

$$\alpha(\frac{\lambda_{i-j}}{\Lambda}B_{ijc}E_{AB}) + 2\kappa > 0. \tag{5.100}$$

Since both terms in the above equation are positive as long as α and κ are positive the condition is satisfied.

Incorporating all these relationships as a rule of thumb the following rule is obtained.

$$\beta = \gamma > \kappa > \frac{\alpha}{2} \tag{5.101}$$

During the simulations these parameters may be tuned accordingly in order to secure good results.

5.2.6 Cooling Schedule

Similar to simulated annealing, a cooling schedule must be specified. Cooling schedule includes the initial temperature, the stopping criterion, the time spent at each temperature and the temperature updating rule. These are described below in detail.

1. Initial Temperature

Initial temperature is found by finding a critical point where the energy decreases significantly. Finding this temperature is important to obtain the best map and to avoid unnecessary computations.

2. Stopping Criterion

The annealing process is stopped when the following saturation conditions are met.

1. All neuron values are within the range [0.0,0.1] or within the range [0.9,1.0] without any exceptions;

2. When the following criterion is met:

   Case 1:

$$\frac{\sum_i \sum_j \sum_r (V_{ijr})^2}{N} > 0.95, \tag{5.102}$$

   Case 2:

$$\frac{\sum_i \sum_j \sum_c (V_{ijc})^2}{N} > 0.95, \tag{5.103}$$

   Case 3:

$$\frac{\sum_i \sum_j \sum_c \sum_r (V_{ijcr})^2}{N} > 0.95, \tag{5.104}$$

   where N is the number of neurons that have values within the range [0.9,1.0].

3. Time spent at each Temperature

At each temperature the mean field equations are iterated until the following convergence criterion is met.

Case 1:

$$\sum_i \sum_j \sum_r \mid V_{ijr}(t+1) - V_{ijr}(t) \mid < 0.001 N_{on} \tag{5.105}$$

Case 2:

$$\sum_i \sum_j \sum_c \mid V_{ijc}(t+1) - V_{ijc}(t) \mid < 0.001 N_{on} \tag{5.106}$$

Case 3:

$$\sum_i \sum_j \sum_c \sum_r \mid V_{ijcr}(t+1) - V_{ijcr}(t) \mid < 0.001 N_{on} \tag{5.107}$$

where N_{on} is the number of non-zero neuron elements.

4. **Temperature Updating Rule**

The updating rule of the temperature is same as the updating rule used in the simulated annealing method. It is given here again for completeness,

$$T = a\frac{C}{\ln(j)} \tag{5.108}$$

where a is a constant and j is the iteration index.

5.2.7 Mean Field Annealing Algorithm

After formulating the energy function, finding a suitable cooling schedule and finding all the necessary parameters, the mean field annealing algorithm as shown in Figure 5.2 can be implemented. The main features of the algorithm are summarized below:

1. Initialize the neurons with random numbers as follows:

$$V_{ijr} = rand[0,1]n_{ij}, \tag{5.109}$$

$$V_{ijc} = rand[0,1]n_{ij}, \tag{5.110}$$

$$V_{ijcr} = rand[0,1]n_{ij}; \tag{5.111}$$

2. Anneal the network until the network is saturated according to the saturation criterion defined before.

3. At each temperature, iterate the MFT equations given below until the convergence criterion is satisfied.

Case 1:

$$V_{ijr} = \frac{n_{ij}}{2}[1 + \tanh \frac{h_{ijr}^{MFT}}{2T} n_{ij}] \tag{5.112}$$

Case 2:

$$V_{ijc} = \frac{n_{ij}}{2}[1 + \tanh \frac{h_{ijc}^{MFT}}{2T} n_{ij}] \tag{5.113}$$

Case 3:

$$V_{ijcr} = \frac{n_{ij}}{2}[1 + \tanh \frac{h_{ijcr}^{MFT}}{2T} n_{ij}] \tag{5.114}$$

Using the mean field annealing algorithm the network is simulated and the simulation results for various cases are presented in the next chapter.

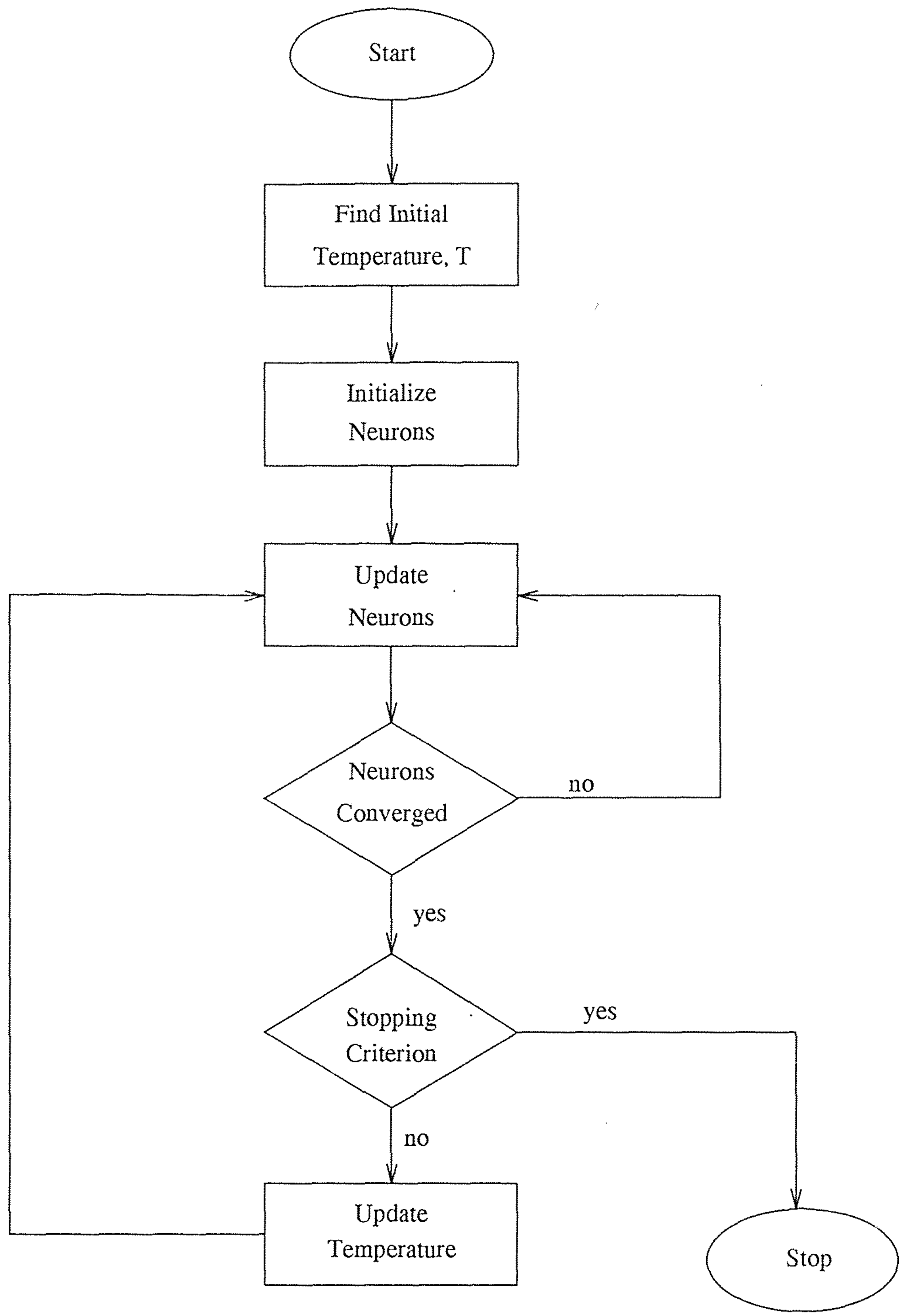

Figure 5.2 Mean Field Annealing Algorithm

CHAPTER 6

NETWORK SIMULATION AND DISCUSSION OF RESULTS

The proposed traffic management scheme is simulated and the results from different experiments are presented in this chapter. A reasonably sized network is used in the simulation and shown in Figure 6.1. From Figure 6.1 the following properties about the network are apparent:

- The network is mesh connected
- The network has 11 nodes and 47 links
- The link capacities may vary from link to link
- The total capacity of the network is fixed
- Each O-D pair has a direct link and alternate routes varying from zero to four

The following elements of the simulation are pre-defined:

- The connection matrix of the network as shown in Figure 6.2, i.e., the connection between nodes of the network (In Figure 6.2, a "1" indicates a link from i to j)
- The O-D pairs and the possible alternate routes for each O-D pair
- The arrival rates to each O-D pair

The total number of circuits for the network is varied in different experiments. These are stated more specifically in the simulation results.

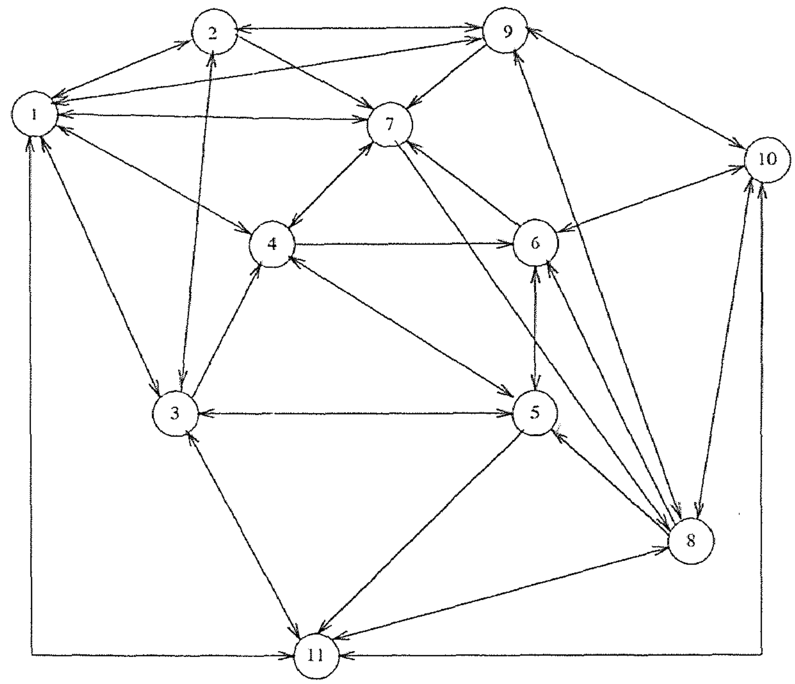

Figure 6.1 Network Used in Simulation

Node j

Node i	1	2	3	4	5	6	7	8	9	10	11
1	0	1	1	1	0	0	1	0	1	0	1
2	1	0	1	0	0	0	1	0	1	0	0
3	1	1	0	1	1	0	0	0	0	0	1
4	1	0	0	0	1	1	1	0	0	0	0
5	0	0	1	1	0	1	0	0	0	0	1
6	0	0	0	0	1	0	1	0	0	1	0
7	1	0	0	1	0	0	0	1	0	0	0
8	0	0	0	0	1	1	0	0	1	1	1
9	1	1	0	0	0	0	1	1	0	1	0
10	0	0	0	0	0	1	0	1	1	0	1
11	1	0	1	0	0	0	0	1	0	1	0

Figure 6.2 Connections Between Nodes of the Network

6.1 Network Simulations and Discussions

Various types of experiments are performed on the network to study the proposed scheme. Some of the experiments are listed below. The results and the discussions will follow in the next section.

- Network performance with and without alternate routes
- Network performance with a reservation scheme
- Performance of the network after varying only the reservation parameter using simulated annealing and mean field annealing
- Performance of the network after varying only the link capacities using simulated annealing and mean field annealing
- Performance of the network after varying the reservation parameter and the link capacities using simulated annealing and mean field annealing

All measurements (i.e., throughput, arrival rate) in the plots shown in this chapter are normalized to the total capacity C of the network. All measurements of arrival rate and throughput are measured in *number of calls per one time unit.*

6.1.1 Simulation 1: Network Performance with and without Alternate Routes

This simulation is done to show that allowing alternate routes in a nonhierarchical network results in better performance of the network at moderate load conditions rather than at overload conditions. The total capacity of the network is 940 circuits. The arrival rates into the O-D pairs and the capacity of each link are made equal so that the effect of alternate routes can be studied alone. The network was first simulated with no alternate routes allowed, and then calls to any O-D pair were allowed to use the predefined alternate routes if the direct link was found busy, (i.e., there were no idle circuits on the direct link). Figures 6.3 and 6.4 show the performance of the network with 20 circuits per link at moderate load and overload, respectively. The following results are observed from the simulation:

- The network throughput improves when alternate routes are made available at moderate load
- With near low load (0.55 calls per time unit) and near high load (around 0.80 calls per time unit) the gain in throughput becomes very small.
- At overload conditions the network shows poor performance when alternate routes are made available.

From this simulation it is clear that allowing alternate routes at moderate loads is advantageous because the calls that cannot go through the direct link may go through the alternate routes. At overload conditions not only is the advantage of alternate routes lost, but also the alternate routes deteriorate the performance and the network becomes more and more unstable. This behavior is due to the fact that when the offered load goes into the heavy load region, many calls get blocked on direct links and are subsequently routed via an alternate route. Every alternately routed call occupies two links in the network, thus further increasing the probability of future calls being blocked from the network, and as a consequence, the network soon becomes saturated.

6.1.2 Simulation 2: Network Performance with a Reservation Scheme

In order to alleviate instability and to improve the performance of the network with alternate routes under heavy and overloaded conditions, the reservation scheme, where a fraction of circuits in the links are reserved only for direct calls, is applied to the network. The network is simulated with some portion of the capacity of each link reserved for direct calls. This is done with a network which has 20 circuits per link and the reservation parameters used are 5%, 10% and 20%. The results from this

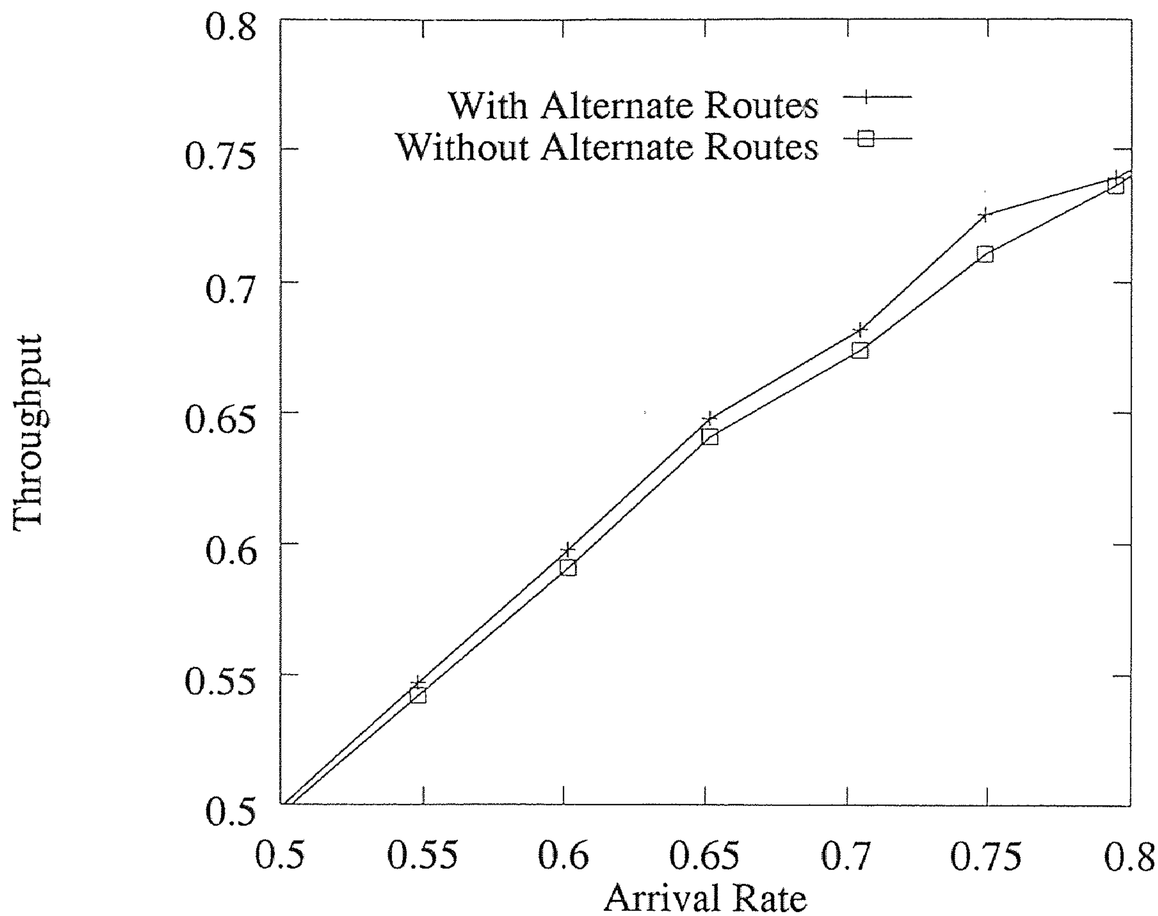

Figure 6.3 Routing of Calls with and without Alternate Routes under Moderate Load Conditions

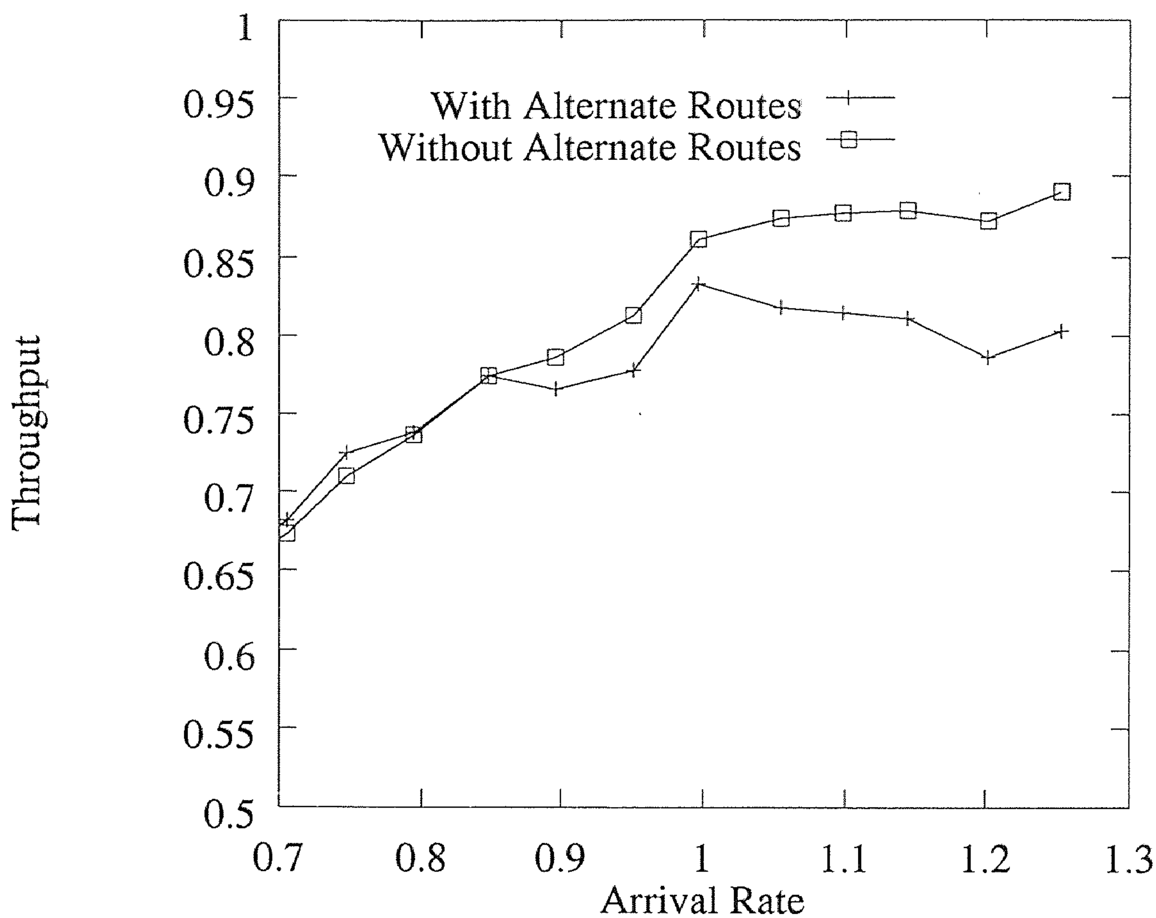

Figure 6.4 Routing of Calls with and without Alternate Routes under Overload Conditions

simulation are plotted in Figure 6.5. From these results the following observations are made:

- When reservation is applied there is significant improvement of throughput at overload conditions
- As the reservation parameter is increased performance increases, but the significance of the improvement becomes negligible

From the observations it is evident that near overload conditions make it necessary to impose some reservations in order to overcome the instability that alternate routing causes. Figure 6.6 shows that by allowing alternate routes along with a reservation scheme improves the performance at moderate loads, and avoids the instability at overloads and performs better than a network without any alternate routes.

In this simulation the link capacities were not varied, but if the capacity and the reservation parameter of each link could be adjusted, then a better solution than one for fixed capacity can be found. Adjusting the capacity of each link is treated later in this chapter.

6.1.3 Simulation 3: Network Performance after Annealing: Varying Reservation Parameter Only

Unlike the previous simulation, where all the links were assigned the same reservation parameter, in this simulation the reservation parameter of each link is allowed to vary while keeping the capacity of each link fixed. Both the simulated annealing and mean field annealing methods are tried in optimizing the network performance.

When optimization by simulated annealing is performed, initially, 20% of the link capacities are reserved for direct calls. Annealing is done for different arrival rates, and the results from the simulation are plotted in Figure 6.7. The time spent at each temperature during annealing is 2350 trials. The generation of new states is stopped and the next temperature level is tried when either the number of accepts at

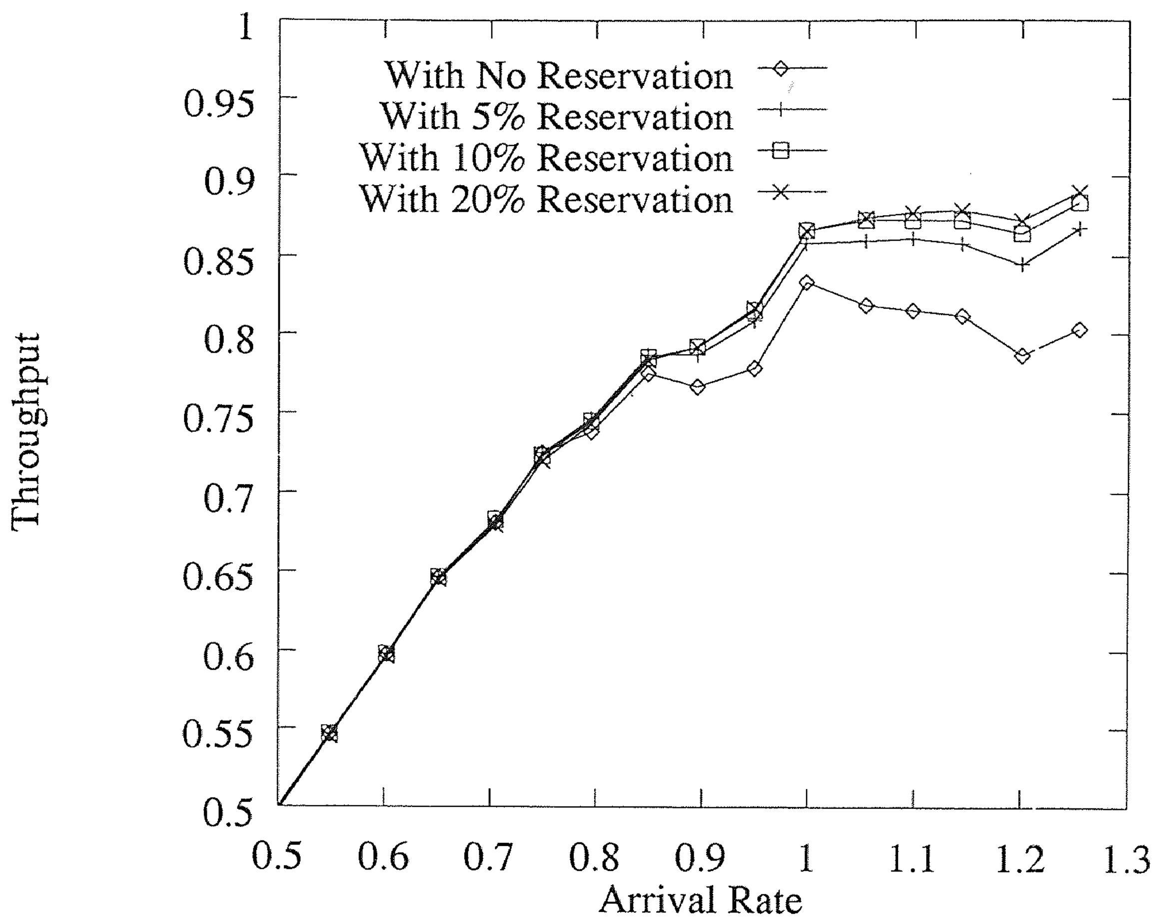

Figure 6.5 Network Performance with Different Amount of Reservation Parameters

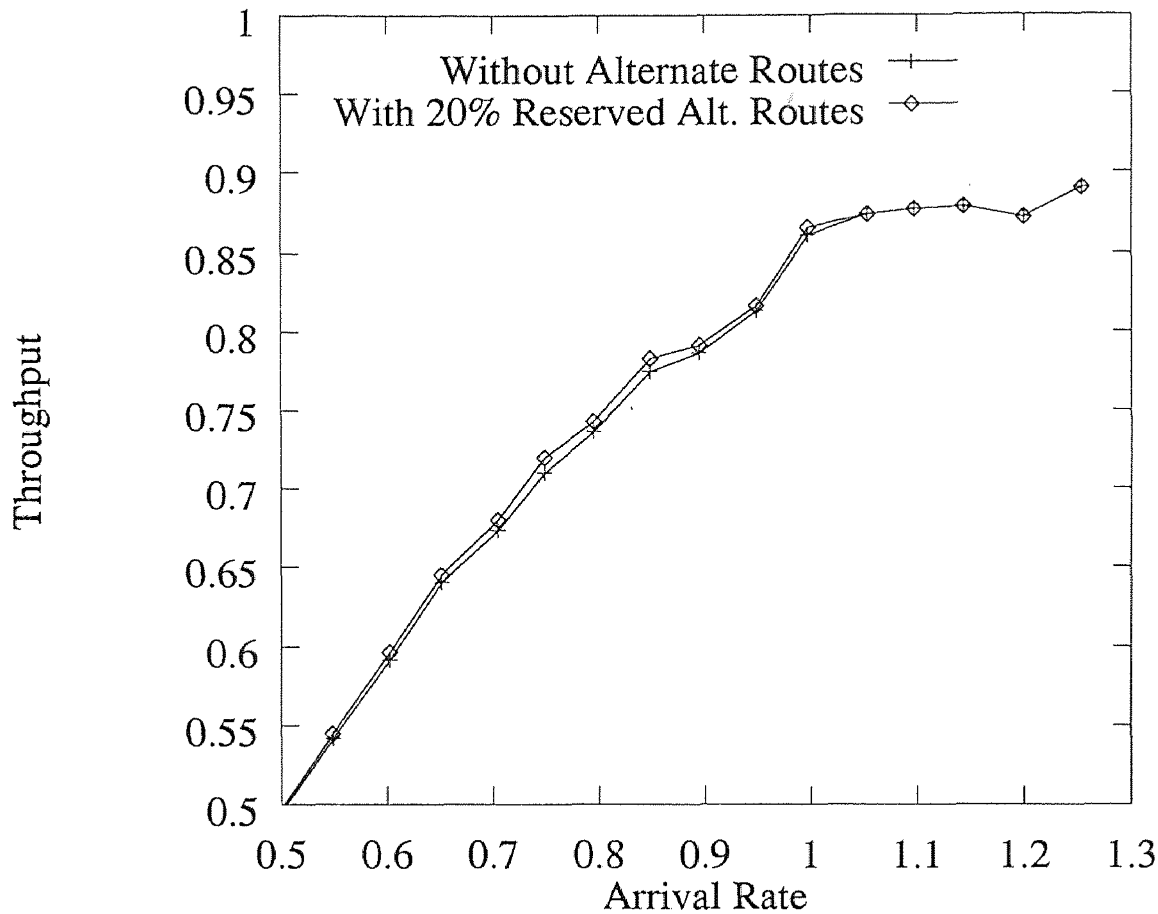

Figure 6.6 Network Performance without Alternate Routes and with Alternate Routes under a Reservation Scheme

a temperature level reaches 470 or the total trials (generation of new states) reaches 2350. Table 6.1 gives the number of accepts at each temperature cycle. Values here are obtained when annealing is done for the normalized arrival rate of 0.8 calls.

For different arrival rates optimization is also done by mean field annealing. The results from this method are also plotted in Figure 6.8. Figure 6.9 shows the performance comparison between simulated annealing and mean field annealing. In order to show the advantage of using mean field annealing, CPU time measurements are taken and shown in Table 6.3. The parameters used in the simulations are given in Table 6.4 and Table 6.5. From these results the following observations can be made:

- The measured throughput with different arrival rates shows significant improvement from the previous simulations where no annealing is done
- The simulation results using two techniques produce results which are very close to each other
- From the CPU time measurements, it is found that optimization by mean field annealing is much faster than simulated annealing

Table 6.1 Annealing Schedule: Varying Reservation Parameter

Temperature Cycle	Number of Accepts
1	470
2	108
3	116
4	14
5	15
6	2
7	0
8	0
9	0

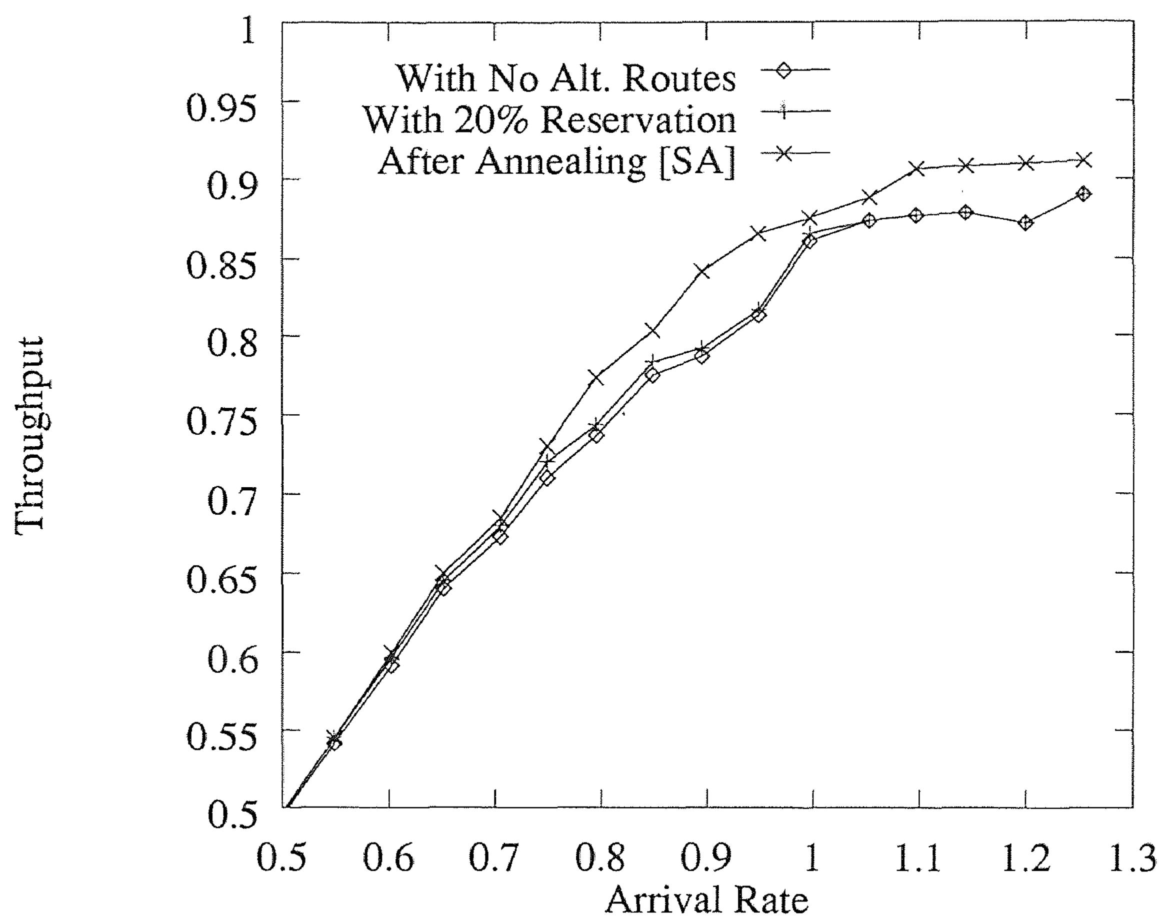

Figure 6.7 Network Performance after Simulated Annealing: Varying the Reservation Parameter Only

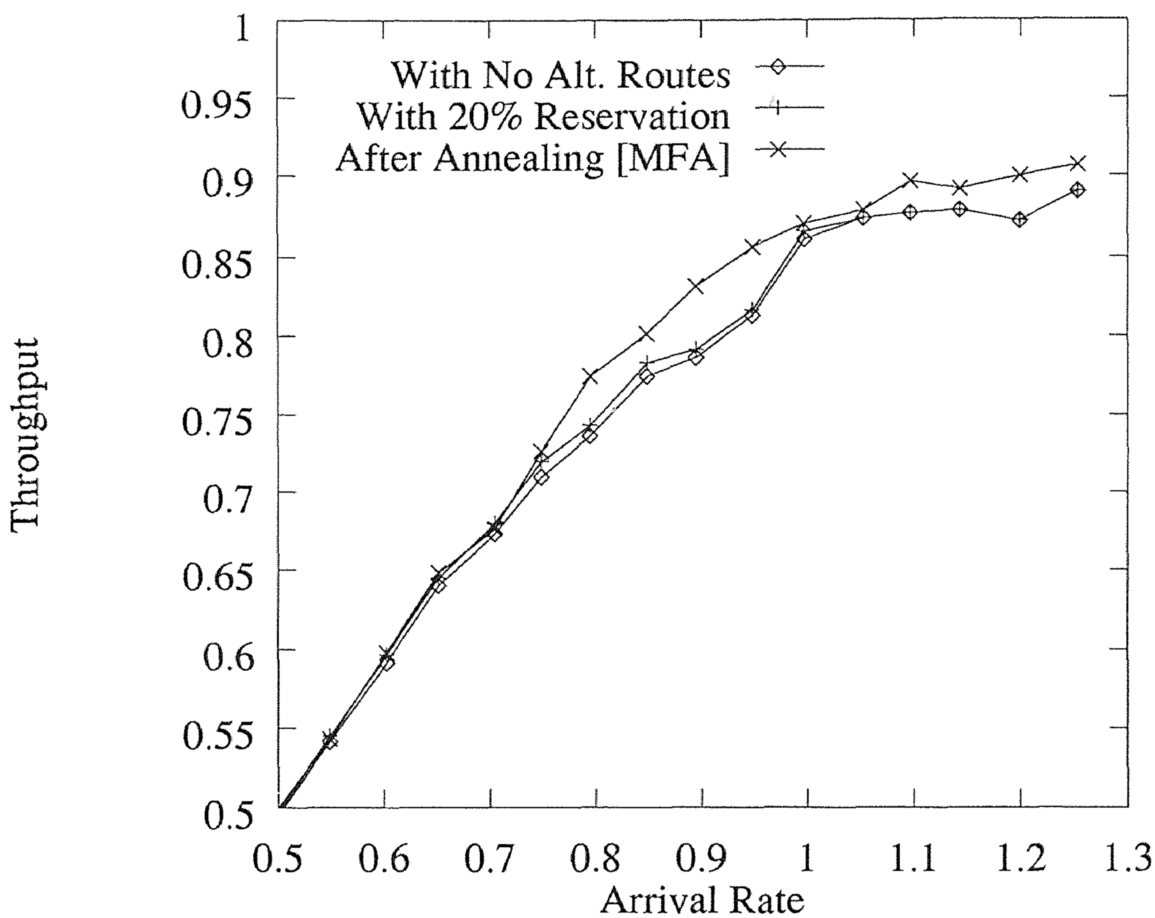

Figure 6.8 Network Performance after Mean Field Annealing: Varying the Reservation Parameter Only

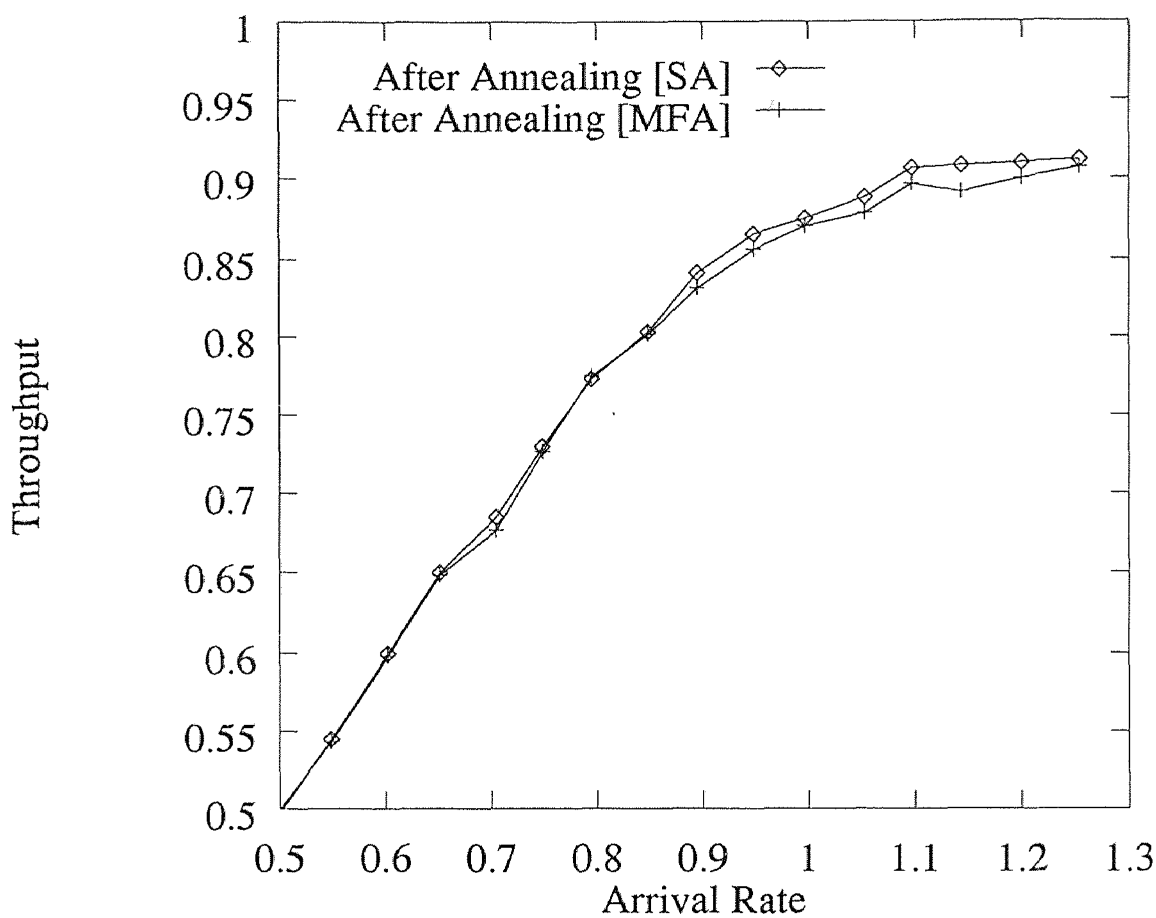

Figure 6.9 Comparison of Simulated Annealing and Mean Field Annealing: Varying the Reservation Parameter Only

6.1.4 Simulation 4: Network Performance after Annealing: Varying the Link Capacities Only

In this simulation, instead of varying the reservation parameter of each link, the capacity of each link is varied. The constraint here is that the total capacity of the network is fixed. The network is simulated with 20% of the link capacity reserved for direct calls only. For various arrival rates the simulations are done using simulated annealing and mean field annealing optimization techniques. The results obtained from these simulations are presented in Figure 6.10 – Figure 6.12. Table 6.2 shows the annealing schedule obtained from simulated annealing. Similar to Simulation 3, CPU time measurements are taken and tabulated in Table 6.3. The parameters used in the simulations are given in Table 6.4 and Table 6.5.

Table 6.2 Annealing Schedule: Varying Link Capacity

Temperature Cycle	Number of Accepts
1	940
2	940
3	940
4	831
5	376
6	111
7	33
8	0
9	0
10	0

Table 6.3 CPU Times of Annealing Methods

Varying Parameter	SA (Time in min.)	MFA (Time in min.)
Reservation	264.5	12.3
Capacity	412.4	16.7

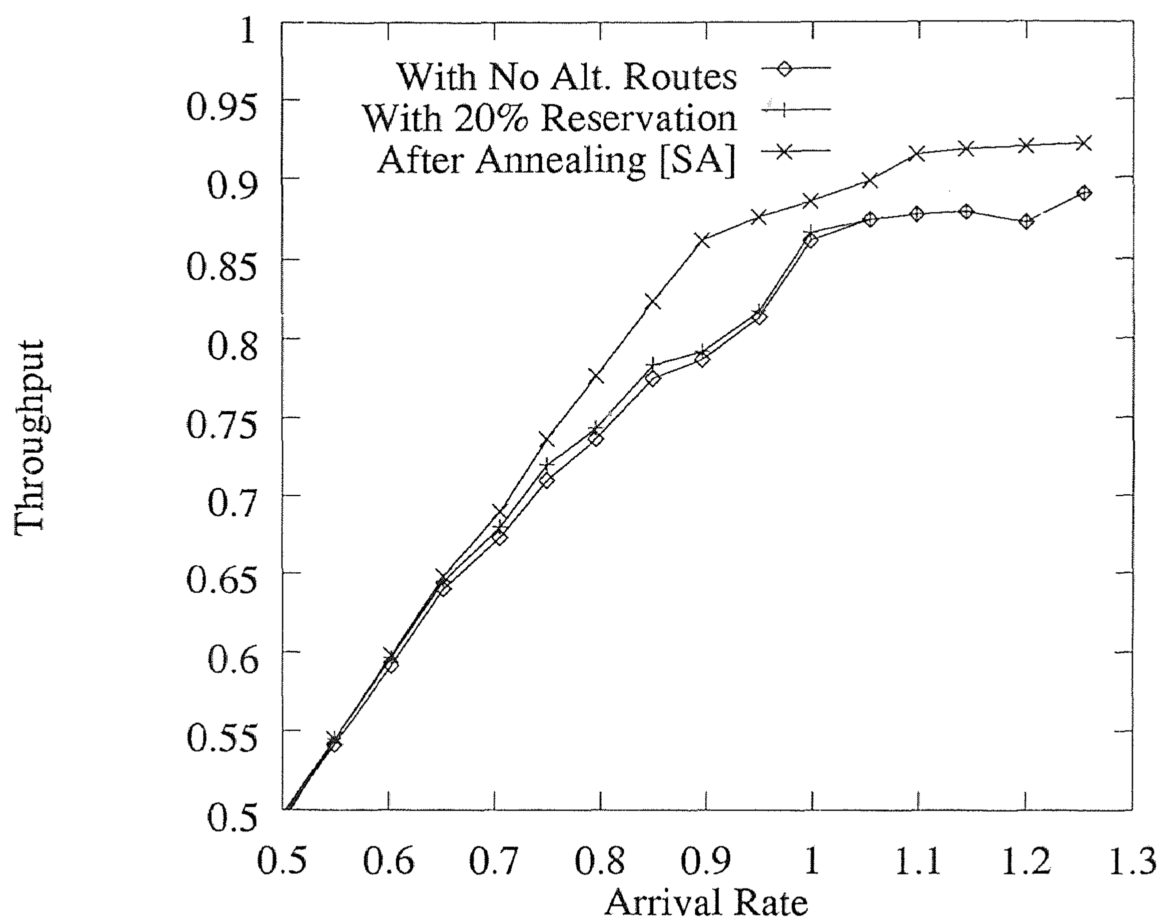

Figure 6.10 Network Performance after Simulated Annealing: Varying the Link Capacities Only

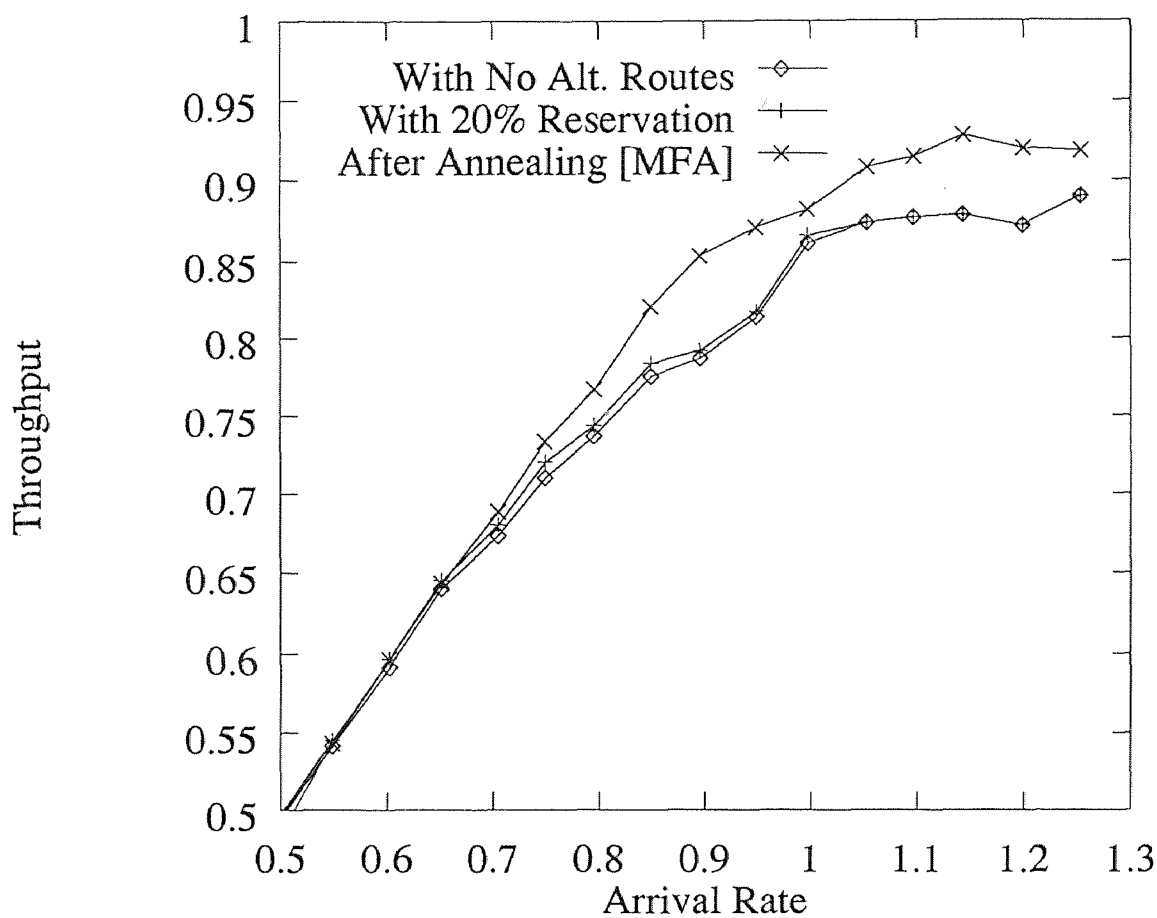

Figure 6.11 Network Performance after Mean Field Annealing: Varying the Link Capacities Only

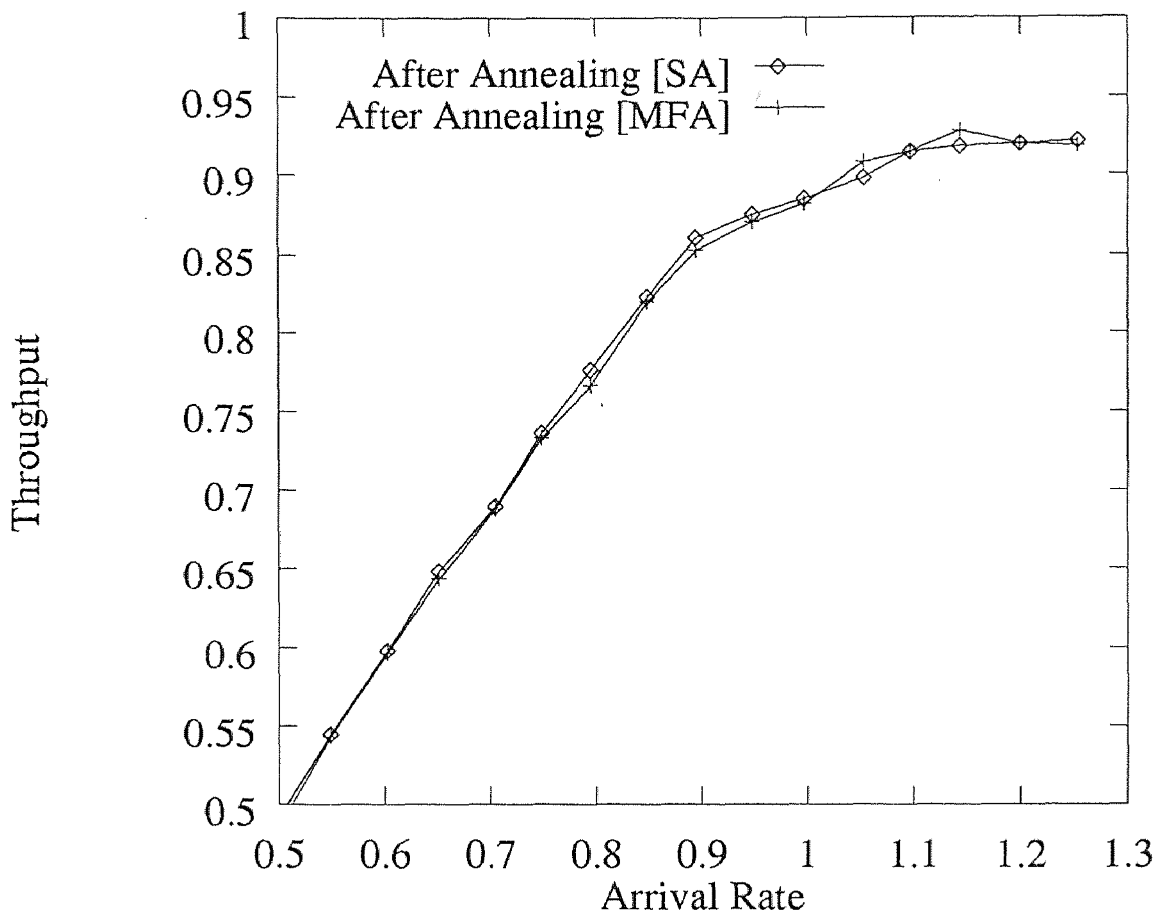

Figure 6.12 Comparison of Simulated Annealing and Mean Field Annealing: Varying the Link Capacities Only

Table 6.4 Parameter Values Used in Simulated Annealing

Parameter	Varying Reservation	Varying Capacity
a	0.002	0.004
$i_{initial}$	25	35
increment value of i	10	10

Table 6.5 Parameter Values Used in Mean Field Annealing

Parameter	Varying Reservation	Varying Capacity
a	0.002	0.004
$i_{initial}$	25	35
increment value of i	10	10
α	0.5	0.5
β	2.0	2.0
γ	0.3	2.0
κ	–	0.4

6.1.5 Simulation 5: Network Performance after Annealing: Varying Both Capacity and Reservation Parameter of Each Link

In this simulation, the number of circuits per link and the reservation parameter were allowed to vary simultaneously and independent of each other. The annealing was performed starting with all links having 20 circuits and 20% of the circuits reserved for direct calls. During the simulated annealing process, even after spending much more time at each temperature, the algorithm did not converge to a good solution. This indicates that the annealing needs to be done for a longer time. Also, due to the large solution space, the number of neurons needed for the mean field annealing algorithm is tremendous. This may introduce divergence and inaccurate solutions. Due to these problems results for this case were not obtained. Some control measures should be taken in order to reduce the solution space. Therefore, this method of annealing is not recommended. It is very efficient to vary the capacity and keep the reservation parameter around 20%.

CHAPTER 7

CONCLUSIONS

From the simulation results it can be verified that having alternate routes improves the performance of a satellite network at moderate load conditions. However, when having alternate routes, the network becomes unstable as the offered load is increased to a heavy load region and after a critical point the performance deteriorates. In order to overcome this undesirable effect, a control scheme is implemented where some portion of the link capacity is reserved for routing direct calls only. The results obtained here show that this is an effective control mechanism in avoiding instability under overloaded traffic conditions. Furthermore, this implementation of the reservation scheme improves the throughput.

Largely due to the ease of changing the link capacities in a satellite communication network (in contrast to terrestrial networks), the network was able to adapt to the current traffic pattern while routing each call dynamically. Usage of simulated annealing and mean field annealing "fine tunes" the network configuration and thus improves the performance. The problem with the simulated annealing algorithm is that the annealing process is very time consuming. Mean field annealing reduces processing time significantly and produces a configuration which improves performance.

One of the important features of the proposed traffic management scheme is the flexibility with which the factors that affect the network performance can be included. This is done effectively by properly defining the cost function for both simulated annealing and mean field annealing algorithms. An added feature of the scheme for improving the throughput of the network is that only the external arrival rates were used, and this was easily measured from the network. From the simulation results it is found that varying both the capacity and the reservation parameter of each link leads to many problems, such as long processing time and divergence. In the mean field annealing algorithm, the number of neurons needed for this case is huge; therefore, the parameter selection is very important, otherwise the algorithm may never converge. This was evident in the simulations for this thesis, and no suitable solution was found. It is more efficient to change the capacity while keeping the reservation parameter constant.

The selection of appropriate parameters for simulated annealing and mean field annealing is essential. If the selection is poor the algorithms may diverge and may fail to produce an optimal solution. In this thesis an approximate method to find the parameters for mean field annealing is analyzed. However, more rigorous and efficient methods should be developed in the future.

Further studies may be done on improving the routing module in the proposed traffic management scheme. Also, a delay analysis could be done for this scheme, since delay in a network is an important feature aside from the throughput. Other types of communication networks (e.g., cellular communication networks), where the facilities can be changed without increasing the cost significantly may be tried for optimizing performance using the scheme proposed in this work.

REFERENCES

1. S. J. Campanella, "Small low cost earth stations, A major trend." *IEEE Spectrum*, vol. 24, no. 1, pp. 42–43, January, 1987.

2. R. Stanyard, *World Satellite Survey*, Eyre & Spottiswoode, London and Margate, 1987.

3. C. M. Melliar-Smith, "Photonic and electronic device technology," *AT&T Tech. Journal*, vol. 68, Jan./Feb. 1989.

4. J. M. Casas, *Opportunities for Satellite Sys. in the European ISDN*, 1988.

5. M. Schwartz, *Telecommunication Networks: Protocols, Modelling and Analysis*. Addison–Wesley Publishing Co., 1987.

6. G. R. Ash, "Design and Control of Networks with Dynamic Non-hierarchical Routing," *IEEE Communications Magazine*, vol. 28, no. 10, pp. 34–40, 1990.

7. A. Gerard and M. Bell, "Blocking Evaluation for Networks with Residual Capacity Adaptive Routing," *IEEE Transactions on Communications*, vol.37, no. 12, pp. 1372–1380, 1989.

8. A. Gersht and A. Shulman, "Optimal Routing in Circuit Switched Communication Networks," *IEEE Transactions on Communications*, vol. 37, no. 11, pp. 1203–1211, 1989.

9. J. S. Kaufman, "Blocking in Shared Resource Environment,"*IEEE Transactions on Communications*, vol. 29, no. 10, pp. 1474–1481, 1981.

10. J. M. Aein and O. S. Kosonych, "Satellite Capacity Allocation," *Proceedings of IEEE*, vol. 65, no. 3, pp. 332–342, 1977.

11. J. M. Akinpela, "Over-Load Performance of Engineering Networks with Non-Hierarchical and Hierarchical Routing," *AT&T Bell Labs Tech J.*, vol. 63, no. 7, pp. 1261–1281, 1984.

12. D. Mitra, R. J. Gibbens and B. D. Huang, "State-Dependent Routing on Symmetric Loss Networks with Trunk Reservations I," *AT&T Bell Laboratories*, Murray Hill, NJ.

13. S. Balasekar and N. Ansari, "Adaptive Map Configuration and Dynamic Routing to Optimize The Performance of a Satellite Communication Network," to be presented at 1993 *IEEE GLOBECOM*, Houston, Tx, Nov.29–Dec.2, 1993.

14. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller and E. Teller, "Equations of State Calculations by Fast Computing Machines," *J. of Chemical Physics*, no. 21, pp. 1087-1091, 1953.

15. L. Kleinrock, *Queuing Systems*, vol. 1, John Wiley & Sons, NY, 1975.

16. L. Kleinrock, *Queuing Systems*, vol. 2, John Wiley & Sons, NY, 1976.

17. S. Kirkpatrick, C.D. Gelatt Jr. and M.P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, pp 671–680, 1983.

18. S. Gemen and D. Gemen, " Stochastic Relaxation, Gibbs Distributions and Baysian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721–741, 1984.

19. B. Gidas, "Non-stationary Markov Chains and Convergence of the Annealing Algorithm," *J. of Statistical Physics*, no. 39, pp. 73-131, 1985.

20. S. B. Gelfand, "Analysis of Simulated Annealing Type Algorithms," *PhD Thesis*, Massachusetts Institute of Technology, 1987.

21. P. J. M. Van Laarhoven and E. H. L. Arts, *Simulated Annealing: Theory and Applications*, Dordrecht, Holland: Reidel Publishing Company, 1987.

22. C. Peterson, "A Mean Field Theory Learning Algorithm for Neural Networks," *Complex System*, Vol. 1, pp. 995–1019, 1987.

23. C. Peterson, "Neural Networks and NP-Complete Optimization Problem; a Performances Study on the Graph Bisection Problem," *Complex System*, Vol. 2, pp. 59–89, 1988.

24. C. Peterson and E. Hartman, "Explorations of the Mean Field Theory Learning Algorithm," *Neural Network*, Vol. 2, pp. 475–494, 1989.

25. G. Bilbro and R. Mann, et al, "Optimization by Mean Field Annealing," *Advances in Neural Information Processing System 1*, ed. D. S. Touvetzky, pp. 91–98, 1989.

26. N. Ansari, E. S. H. Hou and Y. Yu, "A New Method to Optimize the Satellite Broadcasting Schedules Using The Mean Field Annealing of a Hopfield Neural Network," to appear in *IEEE Transactions on Neural Networks*.

27. J. J. Hopfield and D. W. Tank, "Neural Computation of Decisions in Optimization Problems," *Biological Cybernetics*, Vol. 52, pp. 141–152, 1985.