Adv. Appl. Prob. **20,** 112–144 (1988)
Printed in N. Ireland

ROUTING IN CIRCUIT-SWITCHED NETWORKS: OPTIMIZATION, SHADOW PRICES AND DECENTRALIZATION

F. P. KELLY,* *University of Cambridge*

Abstract

How should calls be routed or capacity allocated in a circuit-switched communication network so as to optimize the performance of the network? This paper considers the question, using a simplified analytical model of a circuit-switched network. We show that there exist implicit shadow prices associated with each route and with each link of the network, and that the equations defining these prices have a local or decentralized character. We illustrate how these results can be used as the basis for a decentralized adaptive routing scheme, responsive to changes in the demands placed on the network.

NETWORK FLOW; ADAPTIVE ROUTING; DECENTRALIZATION

1. Introduction

Consider a network such as that illustrated in Figure 1. This might represent a telephone network or a circuit-switched computer communication network. There are finitely many links, labelled $k = 1, 2, \cdots, K$, and link k comprises C_k circuits. A subset $r \subset \{1, 2, \cdots, K\}$ identifies a route. Calls requesting route r arrive as a Poisson process of rate ν_r, and as r varies it indexes independent Poisson streams. A call requesting route r is blocked and lost if on any link $k \in r$ there are no free circuits. Otherwise the call is connected and simultaneously holds one circuit on each link $k \in r$ for the holding period of the call. The call holding period is independent of earlier arrival times and holding periods; holding periods of calls on route r are identically distributed with unit mean. Write R for the set of possible routes. In a typical network K might be measured in thousands and the set R might be much larger still.

A question of interest for such networks is how the routes used affect the performance of the network. For example, in the network of Figure 1 it may be that routes $\{1, 2\}$ and $\{4, 5, 6\}$ can both be used to carry calls between nodes α and γ. Suppose that we decide to increase $\nu_{\{4,5,6\}}$ and decrease $\nu_{\{1,2\}}$, holding the sum constant. How does this change affect the proportion of calls between α and γ that are lost, and, possibly as important, how does it affect the proportion of calls between β and γ, or between β and ζ, that are lost? In more complex networks there will be many substitute routes available and it would be useful to have a

Received 30 April 1986; revision received 18 November 1986.
* Postal address: Statistical Laboratory, 16 Mill Lane, Cambridge CB2 1SB, UK.

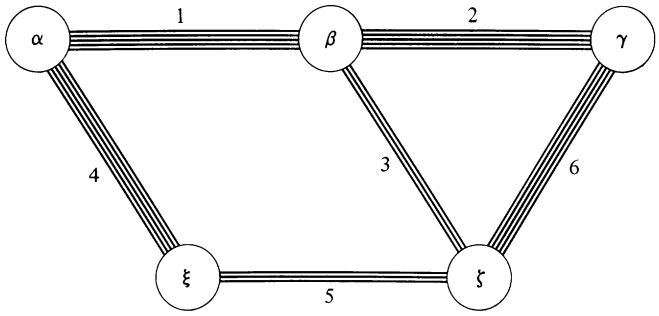

Figure 1. A circuit-switched network

simple criterion for comparing their efficiency. This criterion should take into account not just the loss probabilities along the routes compared, but also the knock-on effects upon the other routes in the network.

As described, a call has but one try to get through the network: a call requesting route r is lost if any link $k \in r$ has no free circuits. This we shall call *fixed routing*. In this paper we shall also consider schemes where, for example, a call between nodes α and γ which is blocked on the route $\{1, 2\}$ might then attempt the route $\{4, 5, 6\}$. Such schemes are special cases of what we shall term *alternative routing*. Again, questions arise as to which routes should be tried first, and how many routes should be tried before a call is discarded.

A related issue concerns the extent to which control can be decentralized. Over a period of time the form of the network or the demands placed on it may change, and routings may need to adapt accordingly. A single node could perhaps control this, receiving information from everywhere in the network and making all decisions about routing. But this approach has drawbacks, particularly if links or nodes may fail. Could control be distributed over the nodes of the network, with computations and decisions made locally? A distributed control scheme should be able to react rapidly to a local disturbance at the point of the disturbance, with slower adjustments in the rest of the network as effects propagate outwards.

To make progress with these questions we shall use a simplified analytical model, which we describe initially for a network with fixed routing. Let B_k, $k = 1, 2, \cdots, K$, solve the equations

$$B_k = E\Big(\sum_{r:k\in r} \nu_r \prod_{j\in r-\{k\}} (1-B_j),\, C_k\Big) \qquad k = 1, 2, \cdots, K \tag{1.1}$$

where for scalar ν and C

$$E(\nu, C) = \frac{\nu^C}{C!}\Big[\sum_{n=0}^{C} \frac{\nu^n}{n!}\Big]^{-1}. \tag{1.2}$$

Here $E(\nu, C)$ is Erlang's formula for the proportion of calls lost on a single link of capacity C circuits offered Poisson traffic at rate ν. Then an approximation for the

proportion of calls requesting route r that are lost is

$$L_r = 1 - \prod_{k \in r} (1 - B_k). \tag{1.3}$$

This approximation is a very natural one to consider, and has appeared frequently in the telecommunications literature (for an early example see [5]). The idea behind the approximation is that the stream of rate ν_r is thinned by a factor $1 - B_j$ at each link $j \in r - \{k\}$ before being offered to link k. If these thinnings could be assumed independent both from link to link and over all routes containing link k, then the traffic offered to link k would be Poisson with rate given by the first argument of Erlang's formula in Equations (1.1). By the Brouwer fixed-point theorem Equations (1.1) have a solution, and in [12] it is shown that this solution is unique. Let $B = (B_1, B_2, \cdots, B_K)$ and now write $\nu = (\nu_r, r \in R)$, $C = (C_1, C_2, \cdots, C_K)$. We shall call the solution $B = B(\nu; C)$ to Equations (1.1) the Erlang fixed point and refer to (1.1) and (1.3) as the Erlang fixed-point approximation. (For recent discussions of the approximation see [13], [25], [37].)

We are now able to describe the main results of this paper. To provide some overall measure of the performance of a network suppose that a call accepted on route r generates an expected revenue w_r. Under the Erlang fixed-point approximation the rate of return from the network will be

$$W(\nu; C) = \sum_r w_r \lambda_r \tag{1.4}$$

where $\lambda_r = \nu_r \prod_{k \in r} (1 - B_k)$ and $B = B(\nu; C)$ is the Erlang fixed point. Let $c = (c_1, c_2, \ldots, c_K)$ be the (unique) solution to the equations

$$c_k = \eta_k (1 - B_k)^{-1} \sum_{r: k \in r} \lambda_r \Big(w_r - \sum_{j \in r - \{k\}} c_j \Big) \tag{1.5}$$

where

$$\eta_k = E(\rho_k, C_k - 1) - E(\rho_k, C_k)$$

and

$$\rho_k = \sum_{r: k \in r} \nu_r \prod_{j \in r - \{k\}} (1 - B_j).$$

Thus ρ_k is simply the traffic offered to link k under the approximation procedure. Extend the definition (1.2) to non-integral values of scalar C by linear interpolation. At integer values of C_k define the derivative of $W(\nu; C)$ with respect to C_k to be the left derivative. Then in Section 2 we shall prove that

$$\frac{d}{d\nu_r} W(\nu; C) = (1 - L_r) \Big(w_r - \sum_{k \in r} c_k \Big) \tag{1.6}$$

and

$$\frac{d}{dC_k} W(\nu; C) = c_k. \tag{1.7}$$

Relation (1.6) shows that the effect of increasing the offered traffic on route r can be assessed from the following rule of thumb: an additional call offered to route r will be accepted with probability $1 - L_r$; if accepted it will earn w_r directly, but at a cost c_k for each link $k \in r$. The costs c measure the knock-on effects of accepting a call upon the other routes in the network. From (1.7) it follows that the costs c also have an interpretation as shadow prices, with c_k measuring the sensitivity of the rate of return to the capacity C_k of link k.

The local character of Equations (1.5) and (1.6) is striking. The right-hand side of Equation (1.6) involves costs c_k only for links k on the route r, while Equation (1.5) can be rewritten in the form

$$c_k = \eta_k(1 - B_k)^{-1} \sum_{r:k\in r} \lambda_r(c_k + s_r) \tag{1.8}$$

where

$$s_r = w_r - \sum_{j\in r} c_j. \tag{1.9}$$

Here s_r is the surplus value of a call on route r, and Equation (1.8) involves only surplus values s_r for routes r passing through link k. Under the approximation procedure the product $(1 - B_k)^{-1}\lambda_r$ appearing in Equation (1.8) is the traffic offered to link k from route r, and η_k is determined by ρ_k and C_k. In Sections 4 and 5 we illustrate how these observations can be used as the basis for a decentralized adaptive routing scheme, responsive to changes in the demands placed on the network.

To describe the material of Sections 4 and 5 it is helpful to think in terms of the following model of a distributed computation. Suppose there is a limited intelligence in the form of arithmetical processing ability available for each link k and for each route r. This intelligence may be located centrally or it may be distributed over the nodes of the network; for example the processing for route r might be carried out at the source node for calls on route r. Suppose also that there is the possibility of limited communication between the intelligences of link k and route r provided $k \in r$. Consider now Equations (1.5). In Section 4 we show that the right-hand side of Equations (1.5), regarded as a function of c, is a contraction mapping provided a certain light traffic condition is satisfied. In this case Equations (1.5) can be solved by repeated substitution, a computation which through Equations (1.8) and (1.9) can be distributed over the intelligences of the links and routes. Under the light traffic condition we are also able to show that Equations (1.5) have a certain stability property: any perturbation of the parameter $w_{r'}$ may cause a change in the entire vector $(s_r, r \in R)$ of surplus values, but the change in the component s_r diminishes

rapidly with the extent of the separation between routes r' and r. If the light traffic condition is not satisfied then the right-hand side of Equations (1.5) regarded as a function of c can still be used to define a contraction mapping, simply by taking a mixture of this function and the identity function. This corresponds to a damped version of repeated substitution, and leads again to a distributed computation. The extent of damping necessary depends upon the size of the network. Thus if the light traffic condition is not satisfied the individual intelligences may require some knowledge of the network beyond that locally available, in order sufficiently to damp the repeated substitution. We show that it is enough for the intelligences of links to know just one item of global information, namely K, the total number of links in the network.

In Section 5 we dispense with the fiction that the quantities η_k, B_k and λ_r appearing in Equations (1.5) are fixed and known. We suppose instead that the intelligences of links and routes are capable of sensing carried loads through them, and we show how these measurements of actual loads can be used in the distributed computations of Section 4. The resulting estimates of the derivatives (1.6) can then be used to implement a decentralized hill-climbing search procedure able gradually to vary routing patterns in response to changes in the demands placed on the network.

The relations (1.1), (1.5) and (1.6) could also be used directly as the basis for an efficient hill-climbing algorithm to maximize the function (1.4) over some region for the parameter ν. Hill-climbing algorithms have been used in conjunction with the approximation (1.1) for many years—for a recent example see [31]. But previous algorithms have involved solving for quite large sets of partial derivatives: this step in previous algorithms should be compared with our Equations (1.5) for (just) K unknowns. In Section 6 we consider the general question of whether hill-climbing is likely to find a global rather than just a local optimum.

The analogies with deterministic network flow suggest a number of other applications. For example, the shadow price interpretation of relation (1.7) could be used in algorithms to aid capacity expansion decisions—either with or without simultaneous optimization of routing. In a different direction, the general approach has implications for pricing policy [23] and for the apportionment of revenue between different sections of the network operation [38].

Section 7 describes how the methods of this paper can deal with alternative routing. The major additional consideration is that the surplus value s_r must now take into account that a call blocked on route r is not necessarily lost, but may be connected along an alternative route.

Section 3 contains an informal discussion of the basic cost relation (1.5). The straightforward relationship between its solution c and the derivatives (1.6) and (1.7) is a consequence of our use of the Erlang fixed-point approximation to define λ_r, $r \in R$, and W. Section 3 obtains the corresponding versions of Equations (1.6) and (1.7) when the starting point is taken to be the exact equilibrium distribution for

the stochastic process described at the start of this introduction. There remain a number of interesting questions concerning the relationship between these exact results, the solutions to Equations (1.5) and the quantities estimated in Section 5. Also of possible further interest are the various connections with work on phase transitions, on parallel computation, and on probabilistic hill-climbing algorithms noted in Sections 4, 5 and 6 respectively.

Some comments on the form of the objective function (1.4) are in order. To assess the efficiency of any scheme which can provide a good performance to certain routes at the expense of a poor performance on other routes, it is necessary to make comparisons between the value of different routes. In this paper we make these comparisons by means of the parameters w_r, $r \in R$. We refer to these parameters as expected revenues, but our methods apply equally when there are no revenue considerations, but instead penalties are incurred when a call is lost, with possibly different penalties for different routes. For example, minimizing the proportion of attempted calls that are lost corresponds to maximizing the rate of return from the network when $w_r = 1$, $r \in R$. The parameters w_r, $r \in R$, can be viewed simply as quantitative assessments of the cost to the network operator of losing calls on different routes. Of course, it may well be that the network operator is concerned with a more delicate problem than simply minimizing a weighted average of loss probabilities. For example, the operator may be concerned that no individual loss probability L_r is too far out of line with the overall loss probability for the network. In Section 5 we indicate how such concerns can be handled by our methods, essentially by increasing the nominal parameters w_r for traffic streams which are suffering a poor grade of service.

The scheme described in Section 5 is designed to be adaptive (or quasi-static) rather than dynamic; it attempts to respond to changes in network form or in arrival rates rather than to seek out and utilize capacity left idle for very short periods as a result of statistical fluctuations. Narendra et al. [28] have studied the application of learning automata to adaptive routing—for a recent review see [29]. It has been shown, for example, that learning automata can successfully select routes so as to equalize loss probabilities or loss rates over different routes. It has, however, been unclear how this is related to optimality, and whether the performance of the network could be improved by allowing communication between automata. These are the issues addressed by the approach of this paper. For recent work on dynamic routing see [8] and [30]. For other examples of work on routing in telephone networks see [2], [9] and [19].

There is a substantial literature on the optimization of routing in queueing networks, where many of the concerns are similar. For recent reviews see Mason [24] and Stidham [34], [35]. Gallager [7] has described a routing scheme for a queueing network, and makes a number of interesting general points on adaptive routing and decentralization. Comparing the analysis of Gallager [7] with that presented here we see that in many ways circuit-switched networks are more difficult

than product-form queueing networks (for example, the existence of knock-on effects and the possibility of long-range order) but as partial recompense they allow simpler and more direct analogies with deterministic network flow theory.

2. Main results for fixed routing

For ease of notation and exposition it will be convenient to allow a larger set of routes R than described in the introduction. Let R be an arbitrary finite set, and suppose that a call on route $r \in R$ requires A_{kr} circuits from link k, where $A_{kr} = 0$ or 1. We obtain the special case described in the introduction when each $r \in R$ is a subset of $\{1, 2, \cdots, K\}$, namely $r = \{k : A_{kr} = 1\}$. But we could, in Figure 1, label a call between node α and node γ using links 1 and 2 differently according to whether the call originated from node α or node γ. We could also allow distinct labels r, $r' \in R$ for calls which use exactly the same subset of links but have different values w_r, $w_{r'}$. Later we shall find it useful to include in the collection R a fictitious marker route j, for $j = 1, 2, \cdots, K$, with $A_{kj} = I[j = k]$, and with $\nu_j = w_j = 0$.

Throughout this section $B = (B_1, B_2, \ldots, B_K)$ is the unique [12] solution to the equations

$$B_k = E\left(\sum_r A_{kr}\nu_r \prod_{j \neq k} (1 - B_j)^{A_{jr}}, C_k\right), \quad k = 1, 2, \ldots, K$$

and

$$W(\nu; C) = \sum_r w_r \lambda_r$$

where

$$\lambda_r = \nu_r \prod_j (1 - B_j)^{A_{jr}}.$$

We shall write $B = B(\nu; C)$ when we wish to emphasise the functional dependence of B on the system parameters $\nu = (\nu_r, r \in R)$ and $C = (C_1, C_2, \ldots, C_K)$. We shall sometimes use the labels ν and C for scalars, for example as arguments of Erlang's formula $E(\nu, C)$, but shall indicate this usage explicitly. In summations, products and the definitions of matrices r ranges over R, and i, j, k or l range over $\{1, 2, \cdots, K\}$. To avoid trivialities assume $B_k > 0$ for $k = 1, 2, \cdots, K$.

Lemma 2.1. For scalar ν and C

$$\frac{d}{d\nu} E(\nu, C) = [1 - E(\nu, C)][E(\nu, C - 1) - E(\nu, C)].$$

Proof. This result follows directly from the definition (1.2) of Erlang's formula.

Recall that the route j is a fictitious marker route, passing only through link j and with $\nu_j = w_j = 0$.

Lemma 2.2. The derivative $(d/dv_j)B_k(v; C)$ is defined and continuous over the positive orthant $v \geqq 0$, for $j, k = 1, 2, \cdots, K$.

Proof. In [12], Section 5, it is shown that $B = B(v; C)$ is given by $B_k = 1 - \exp(-y_k)$ where $y = (y_k, k = 1, 2, \cdots, K) \in (0, \infty)^K$ minimizes a functional of the form

$$H_v(y) = \sum_r v_r \exp\left(-\sum_k y_k A_{kr}\right) + \sum_k u(y_k, C_k). \tag{2.1}$$

Here the function u is, in its first argument, twice continuously differentiable, increasing without bound and strictly convex. Observe that the functional $H_v(y)$ depends linearly on v. Hence the representation of y as the unique interior minimum of the form (2.1) establishes that $y = y(v)$ is a continuously differentiable function of v. Since B is a continuously differentiable function of y the result follows.

Lemma 2.3.

$$\frac{d}{dv_j} W(v; C) = -\sum_k (1 - B_k)^{-1}\left(\sum_r A_{kr} w_r \lambda_r\right) \frac{d}{dv_j} B_k(v; C).$$

Proof. Define the form

$$W(v; B; C) = \sum_r w_r v_r \prod_j (1 - B_j)^{A_{jr}}. \tag{2.2}$$

This form is constant in its third argument, which has been included to avoid confusion between $W(.\,;.\,;.)$ and $W(.\,;.)$. Then

$$\begin{aligned}\frac{\partial}{\partial B_k} W(v; B; C) &= -\sum_r A_{kr} w_r v_r \prod_{j \neq k} (1 - B_j)^{A_{jr}} \\ &= -(1 - B_k)^{-1} \sum_r A_{kr} w_r \lambda_r.\end{aligned}$$

Also $(\partial/\partial v_j)W(v; B; C) = 0$, since $w_j = 0$. But

$$\frac{d}{dv_j} W(v; C) = \frac{d}{dv_j} W(v; B(v; C); C) = \left[\frac{\partial}{\partial v_j} + \sum_k \frac{d}{dv_j} B_k(v; C) \frac{\partial}{\partial B_k}\right] W(v; B; C)$$

giving the stated result.

Define

$$\rho_k = \sum_r A_{kr} v_r \prod_{j \neq k} (1 - B_j)^{A_{jr}}$$

and

$$\eta_k = E(\rho_k, C_k - 1) - E(\rho_k, C_k).$$

Lemma 2.4.

$$\eta_k^{-1}\frac{d}{d\nu_j}B_k(\nu;C) = I[j=k](1-B_k) - \sum_{l\neq k}(1-B_l)^{-1}\Big(\sum_r A_{lr}A_{kr}\lambda_r\Big)\frac{d}{d\nu_j}B_l(\nu;C).$$

Proof. Define the form

$$B_k(\nu;B_j, j\neq k;C_k) = E\Big(\sum_r A_{kr}\nu_r\prod_{j\neq k}(1-B_j)^{A_{jr}},\ C_k\Big). \tag{2.3}$$

Then for $l\neq k$ we have, using Lemma 2.1,

$$\begin{aligned}\frac{\partial}{\partial B_l}B_k(\nu;B_j, j\neq k;C_k) &= -(1-B_k)\eta_k\sum_r A_{lr}A_{kr}\nu_r\prod_{j\neq k,l}(1-B_j)^{A_{jr}}\\ &= -\eta_k(1-B_l)^{-1}\sum_r A_{lr}A_{kr}\lambda_r.\end{aligned} \tag{2.4}$$

Also

$$\frac{\partial}{\partial\nu_j}B_k(\nu;B_l, l\neq k;C_k) = I[j=k](1-B_k)\eta_k,$$

again using Lemma 2.1. But

$$\frac{d}{d\nu_j}B_k(\nu;C) = \Big[\frac{\partial}{\partial\nu_j} + \sum_{l\neq k}\frac{d}{d\nu_j}B_l(\nu;C)\frac{\partial}{\partial B_l}\Big]B_k(\nu;B_j, j\neq k;C_k)$$

giving the stated result.

Define the matrices

$$A = (A_{jr})_{j,r}$$
$$\frac{dB}{d\nu} = \Big[\frac{d}{d\nu_k}B_j(\nu;C)\Big]_{j,k},$$

and the diagonal matrices

$$\beta = \operatorname{diag}(1-B_j)_j, \qquad \eta = \operatorname{diag}(\eta_j)_j, \qquad \lambda = \operatorname{diag}(\lambda_r)_r.$$

Note that β and η are both of full rank. Define the row vectors

$$w = (w_r)_r$$
$$\frac{dW}{d\nu} = \Big[\frac{d}{d\nu_j}W(\nu;C)\Big]_j.$$

Write

$$\Lambda = \Big(I[j\neq k]\sum_r A_{jr}A_{kr}\lambda_r\Big)_{j,k},$$

the matrix obtained from $A\lambda A^T$ by deleting its diagonal elements. Then, from Lemmas 2.3 and 2.4,

$$\frac{dW}{d\nu} = -w\lambda A^T\beta^{-1}\frac{dB}{d\nu} \tag{2.5}$$

$$\frac{dB}{d\nu} = \eta\left(\beta - \Lambda\beta^{-1}\frac{dB}{d\nu}\right). \tag{2.6}$$

It is perhaps worth noting that the statement of Lemma 2.2 would follow from the Implicit Function Theorem [33], provided it could be shown that $I + \eta\Lambda\beta^{-1}$ is invertible for all $\nu \geqq 0$. We have found it simpler to prove Lemma 2.2 directly. From Lemma 2.2 and Equation (2.6) it follows that $(I + \eta\Lambda\beta^{-1})^{-1}$ exists, since $\eta\beta$ has full rank.

Define $c = (c_k, k = 1, 2, \cdots, K)$ by

$$c_k = -(1 - B_k)^{-1}\frac{d}{d\nu_k}W(\nu; C). \tag{2.7}$$

Theorem 2.5.

$$c_k = \eta_k(1 - B_k)^{-1}\sum_r A_{kr}\lambda_r\left(w_r - \sum_{j\neq k} A_{jr}c_j\right). \tag{2.8}$$

Proof. From Equations (2.5) and (2.6)

$$\frac{dB}{d\nu} = (I + \eta\Lambda\beta^{-1})^{-1}\eta\beta \tag{2.9}$$

and

$$\begin{aligned}\frac{dW}{d\nu} &= -w\lambda A^T\beta^{-1}(I + \eta\Lambda\beta^{-1})^{-1}\eta\beta \\ &= -w\lambda A^T\beta^{-1}[I - (I + \eta\Lambda\beta^{-1})^{-1}\eta\Lambda\beta^{-1}]\eta\beta.\end{aligned}$$

The diagonal matrices η and β commute, and so

$$\begin{aligned}\frac{dW}{d\nu}\eta^{-1} &= -w\lambda A^T + w\lambda A^T\beta^{-1}(I + \eta\Lambda\beta^{-1})\eta\Lambda \\ &= -w\lambda A^T + w\lambda A^T\beta^{-1}\frac{dB}{d\nu}\beta^{-1}\Lambda \\ &= -w\lambda A^T - \frac{dW}{d\nu}\beta^{-1}\Lambda.\end{aligned}$$

In component form,

$$\eta_k^{-1}\frac{d}{d\nu_k}W(\nu; C) = -\sum_r A_{kr}\lambda_r w_r - \sum_{j\neq k}\left(\sum_r A_{kr}\lambda_r A_{jr}\right)(1 - B_j)^{-1}\frac{d}{d\nu_j}W(\nu; C).$$

Equivalently

$$\eta_k^{-1}(1-B_k)c_k = \sum_r A_{kr}\lambda_r w_r - \sum_{j\neq k}\Big(\sum_r A_{kr}\lambda_r A_{jr}\Big)c_j$$
$$= \sum_r A_{kr}\lambda_r\Big(w_r - \sum_{j\neq k} A_{jr}c_j\Big),$$

giving the desired result.

Lemma 2.6.

$$\frac{d}{d\nu_r}B_k(\nu;C) = \sum_j A_{jr}\Big(\prod_{l\neq j}(1-B_l)^{A_{lr}}\Big)\frac{d}{d\nu_j}B_k(\nu;C).$$

Proof. Define the extended matrices

$$\frac{dB}{d\bar{\nu}} = \Big(\frac{d}{d\nu_r}B_k(\nu;C)\Big)_{k,r}$$
$$\bar{\beta} = \Big(A_{kr}\prod_j (1-B_j)^{A_{jr}}\Big)_{k,r}.$$

From the definition (2.3) and Lemma 2.1

$$\frac{d}{d\nu_r}B_k(\nu;B_j, j\neq k;C_k) = (1-B_k)\eta_k A_{kr}\prod_{j\neq k}(1-B_j)^{A_{jr}}$$
$$= \eta_k A_{kr}\prod_j (1-B_j)^{A_{jr}}.$$

But

$$\frac{d}{d\nu_r}B_k(\nu;C) = \Big[\frac{d}{d\nu_r} + \sum_{l\neq k}\frac{d}{d\nu_r}B_k(\nu;C)\frac{\partial}{\partial B_l}\Big]B_k(\nu;B_j, j\neq k;C_k),$$

and so, using (2.4),

$$\frac{dB}{d\bar{\nu}} = \eta\Big(\bar{\beta} - \Lambda\beta^{-1}\frac{dB}{d\bar{\nu}}\Big).$$

Thus

$$\frac{dB}{d\bar{\nu}} = (I+\eta\Lambda\beta^{-1})\eta\bar{\beta}$$
$$= \frac{dB}{d\nu}\beta^{-1}\bar{\beta},$$

from (2.9). In component form this gives the stated result.

Theorem 2.7.

$$\frac{d}{d\nu_r}W(\nu;C) = \Big(\prod_j (1-B_j)^{A_{jr}}\Big)\Big(w_r - \sum_j A_{jr}c_j\Big).$$

Proof. Using the form (2.2)

$$\frac{\partial}{\partial v_r} W(v; B; C) = w_r \prod_j (1 - B_j)^{A_{jr}}.$$

From this and Lemma 2.6,

$$\begin{aligned}
\frac{d}{dv_r} W(v; C) &= \frac{d}{dv_r} W(v; B(v; C); C) \\
&= \left[\frac{\partial}{\partial v_r} + \sum_k \frac{d}{dv_r} B_k(v; C) \frac{\partial}{\partial B_k}\right] W(v; B; C) \\
&= \left[\prod_j (1 - B_j)^{A_{jr}}\right]\left(w_r + \sum_j A_{jr}(1 - B_j)^{-1} \sum_k \frac{d}{dv_j} B_k(v; C) \frac{\partial}{\partial B_k} W(v; B; C)\right) \\
&= \left[\prod_j (1 - B_j)^{A_{jr}}\right]\left(w_r - \sum_j A_{jr}(1 - B_j)^{-1} \frac{d}{dv_j} W(v; C)\right),
\end{aligned}$$

and the result now follows from the definition (2.7) of c_j.

Recall that we extend the definition (1.2) of $E(v, C)$ to non-integral values of scalar C by linear interpolation. At integer values of C_j define the derivative of $W(v; C)$ or of $B_k(v; C)$ with respect to C_j to be the left derivative.

Lemma 2.8.

$$\eta_k^{-1} \frac{d}{dC_j} B_k(v; C) = -I[j = k] - \sum_{l \neq k} (1 - B_l)^{-1}\left(\sum_r A_{lr} A_{kr} \lambda_r\right) \frac{d}{dC_j} B_l(v; C).$$

Proof. This proof runs parallel to that of Lemma 2.4, but using

$$\frac{d}{dC_j} B_k(v; B_l, l \neq k; C_k) = -I[j = k]\eta_k.$$

Theorem 2.9.

$$\frac{d}{dC_j} W(v; C) = c_j.$$

Proof. Define the matrix

$$\frac{dB}{dC} = \left(\frac{d}{dC_k} B_j(v; C)\right)_{j,k}.$$

In matrix form Lemma 2.8 becomes

$$\frac{dB}{dC} = -\eta\left(I + \Lambda\beta^{-1} \frac{dB}{dC}\right).$$

Thus

$$\frac{dB}{dC} = -(I + \eta\Lambda\beta^{-1})^{-1}\eta$$

$$= -\frac{dB}{d\nu}\beta^{-1}$$

from Equation (2.9). Hence, using the form (2.2),

$$\frac{d}{dC_j}W(\nu; C) = \sum_k \frac{d}{dC_j}B_k(\nu; C)\frac{\partial}{\partial B_k}W(\nu; B; C)$$

$$= -\sum_k (1 - B_j)^{-1}\frac{d}{d\nu_j}B_k(\nu; C)\frac{\partial}{\partial B_k}W(\nu; B; C)$$

$$= -(1 - B_j)^{-1}\frac{d}{d\nu_j}W(\nu; C),$$

and so the result follows from the definition (2.7) of c_j.

Theorems 2.5, 2.7 and 2.9 comprise generalized forms of, respectively, the definition (1.5) and the assertions (1.6) and (1.7) of the introduction.

3. Discussion

In this section we present an informal argument leading to the basic cost relation (1.5). It is perhaps worth mentioning that the relation was first obtained through this argument. However, we have found it easier to establish the results through the approach of Section 2 than to present the argument rigorously. We also discuss the relationship between the results obtained in Section 2 from the Erlang fixed-point approximation and the results that can be obtained from the exact equilibrium distribution for the stochastic process described in the introduction.

Consider a single link of capacity scalar C offered Poisson traffic at rate ρ, and suppose the stochastic process describing the system is stationary. Now consider the effect upon the system of offering the link a single, additional call at time 0. The call will be accepted with probability $1 - E(\rho, C)$. Define the *disposition* of the other calls on the link to be the number of such calls and the elapsed holding times of these calls. Conditional on the additional call being accepted and on its holding time, the disposition of the other calls on the link has the same distribution at the time when the call begins and the time when it ends. Further, this distribution is just the stationary distribution for a link of capacity $C - 1$ offered Poisson traffic at rate ρ (a similar result is well known for closed queueing networks—see [32] or Theorem 3.12 (iii), [10]; for a general result covering both closed queueing networks and the present example see Theorem 9.5, [10]). Informally, the additional call leaves the system distributionally in the same state as it finds it, and while the additional call is

there the system behaves as a stationary single link with one less circuit. Thus if the additional call is accepted and its holding time is h, then the expected number of lost calls is increased by $\rho[E(\rho, C-1) - E(\rho, C)]h$. Now h has unit mean, and so acceptance of the additional call increases the expected number of lost calls by

$$\rho[E(\rho, C-1) - E(\rho, C)]. \tag{3.1}$$

Observe that this is just the increase in the expected number of lost calls per unit time if a single circuit is removed from the link.

Consider next the network described in the introduction. Let c_k be the increase in the expected value of lost calls as a result of removing a single circuit for unit time from link k. Make the approximation that link k is offered traffic on route r at rate $\lambda_r(1-B_k)^{-1}$ for r such that $k \in r$, and that these streams are Poisson and independent. Then the total traffic offered to link k is Poisson at rate ρ_k, and the expected value of calls rejected at link k as a result of removing a single circuit for unit time is

$$\sum_{r:k\in r} \lambda_r(1-B_k)^{-1}[E(\rho_k, C_k-1) - E(\rho_k, C_k)]w_r = \eta_k(1-B_k)^{-1} \sum_{r:k\in r} \lambda_r w_r. \tag{3.2}$$

But a call rejected at link k would, if it had been accepted, have used a single circuit on each link $j \in r - \{k\}$ as well as the single circuit on link k. Rejecting a call at link k leaves free the use of these other circuits elsewhere in the network. This suggests that c_k should be given by (3.2), but with w_r replaced by

$$w_r - \sum_{j\in r-\{k\}} c_j. \tag{3.3}$$

This gives

$$c_k = \eta_k(1-B_k)^{-1} \sum_{r:k\in r} \lambda_r\Big(w_r - \sum_{j\in r-\{k\}} c_j\Big),$$

which is just Equation (1.5).

If we set $w_r = 1$, $r \in R$, in expression (3.2), we obtain $\eta_k\rho_k$, the expected *number* of calls rejected at link k as a result of removing a single circuit for unit time. The expression $\delta_k = \eta_k\rho_k$ is sometimes called Erlang's improvement formula [3], and its use in capacity expansion decisions is known as Moe's principle [36]. The vector $\delta = (\delta_1, \delta_2, \cdots, \delta_K)$ will play an important role in the next section.

The argument leading to expression (3.1) can be made rigorous, and this expression is exact for a single link system. The arguments leading to expressions (3.2) and (3.3) involve approximations which correspond loosely to those implicit in the Erlang fixed-point procedure. In particular, the additive form of the final term in expression (3.3) is a consequence of the link independence assumption.

To shed light on this point we consider next the exact equilibrium distribution for the stochastic process described in the introduction. Let $n_r(t)$ be the number of calls

in progress at time t on route r, and let $n(t) = (n_r(t), r \in R)$. Then the stochastic process $(n(t), t \geqq 0)$ has a unique stationary distribution and under this distribution $\pi(n) = P\{n(t) = n\}$ is given by

$$\pi(n) = G(C)^{-1} \prod_r \frac{\nu_r^{n_r}}{n_r!} \qquad n \in S(C) \tag{3.4}$$

where $S(C) = \{n \in Z_+^R : An \leqq C\}$ and

$$G(C) = \sum_{n \in S(C)} \prod_r \frac{\nu_r^{n_r}}{n_r!}$$

(see, for example, [3], [10]). It follows that the proportion of calls on route r that are accepted is

$$1 - L_r^* = G(C)^{-1} G(C - Ae_r) \tag{3.5}$$

where $e_r \in S(C)$ is the unit vector corresponding to just one call in progress, on route r. Let $B_k^* = L_k^*$ and let $W^*(\nu; C) = E_\pi(\sum_r n_r w_r)$. Thus B^*, L^* and W^* are the analogues of B, L and W, but are calculated from the exact stationary distribution (3.4) rather than the Erlang fixed-point approximation. Define

$$c_r^* = W^*(\nu; C) - W^*(\nu; C - Ae_r). \tag{3.6}$$

Theorem 3.1.

$$\frac{d}{d\nu_r} W^*(\nu; C) = (1 - L_r^*)(w_r - c_r^*).$$

Proof.

$$W^*(\nu; C) = G(C)^{-1} \sum_{n \in S(C)} \left(\prod_r \frac{\nu_r^{n_r}}{n_r!} \right) \sum_{r'} n_{r'} w_{r'}.$$

Thus

$$\begin{aligned} \frac{d}{d\nu_r} W^*(\nu; C) &= G(C)^{-1} \sum_{n \in S(C - Ae_r)} \left(\prod_r \frac{\nu_r^{n_r}}{n_r!} \right) \sum_{r'} (n_{r'} + I[r = r']) w_{r'} \\ &\quad - G(C)^{-2} G(C - Ae_r) \sum_{n \in S(C)} \left(\prod_r \frac{\nu_r^{n_r}}{n_r!} \right) \sum_{r'} n_{r'} w_{r'} \\ &= \frac{G(C - Ae_r)}{G(C)} [W^*(\nu; C - Ae_r) + w_r - W^*(\nu; C)]. \end{aligned}$$

The result now follows from Equations (3.5) and (3.6).

Define

$$c_r = \sum_j A_{jr} c_j. \tag{3.7}$$

Then from Theorem 2.7

$$\frac{d}{dv_r} W(v; C) = (1 - L_r)(w_r - c_r). \tag{3.8}$$

Theorem 3.1 shows that an analogue of this result can be obtained directly from the exact stationary distribution (3.4). In contrast to the additive relation (3.7) between the costs c_r, $r \in R$, appearing in Equation (3.8), from the exact stationary distribution (3.4) only an approximately additive relation between c_r^*, $r \in R$, can be deduced:

$$\begin{aligned} c_r^* &= W^*(v; C) - W^*(v; C - Ae_r) \\ &\approx \sum_j A_{jr}[W^*(v; C) - W^*(v; C - Ae_j)] \\ &= \sum_j A_{jr} c_j^*. \end{aligned}$$

In this paper we are mainly interested in consequences of the additive relation (3.7), and we have found the simplest way to exhibit these consequences is to start out from the Erlang fixed-point approximation. It would be interesting to know under what circumstances the collection $(c_r, r \in R)$ is a good approximation to the collection $(c_r^*, r \in R)$. It is known that under some limiting regimes the Erlang fixed-point approximation emerges as an asymptotic result ([12], [37], [39]). However there are network structures, for example a number of small-capacity links arranged one after another in a line ([15], [40]), where the approximation is inappropriate, and where the relationship between c_r^*, $r \in R$, may be far from additive.

4. Local iterative solutions of the cost relation

The basic cost relation (2.8) is linear in the costs c, and so there are a large number of methods available for finding a solution. For example, the most direct method would involve explicitly inverting a matrix of dimension K. In this section we describe some simple iterative methods which have the property that the calculations involved can be carried out in a distributed fashion, using only local information.

As motivation for the discussion of this and the next section, suppose there is a limited intelligence in the form of arithmetical processing ability available for each link k and for each route r. This intelligence may be located centrally or it may be distributed over the nodes of the network; for example the processing for route r might be carried out at the source node for calls on route r. We will require the possibility of limited communication between the intelligences of link k and route r provided $A_{kr} = 1$. We suppose in this section that the values of ρ_k, η_k, B_k are fixed

and known to the intelligence of link k, while the value of λ_r is fixed and it, together with w_r, is known to the intelligence of route r.

A very natural method for solving Equations (2.8) would be repeated substitution. Define a linear mapping $f:\mathbb{R}^K \to \mathbb{R}^K$ by $f = (f_1, f_2, \cdots, f_K)$,

$$f_k(x) = \eta_k(1 - B_k)^{-1} \sum_r A_{kr}\lambda_r\Big(w_r - \sum_{j\neq k} A_{jr}x_j\Big).$$

Then repeated substitution corresponds to calculating the sequence $f^m(x)$, $m = 1, 2, \cdots$. This calculation is essentially local, since the definition of $f_k(x)$ involves just η_k, B_k, x_k and the terms λ_r, $w_r - \sum_j A_{jr}x_j$ for routes r with $A_{kr} = 1$. We shall see that repeated substitution is successful provided the traffic within the network is not too great. For $a \in (0, 1)$ define $f_{(a)}:\mathbb{R}^K \to \mathbb{R}^K$ by $f_{(a)}(x) = (1-a)x + af(x)$. Thus $f_{(a)}(.)$ is simply a damped version of $f(.)$. We shall see that sufficiently damped repeated substitution is successful whatever the traffic conditions.

Define a norm on $\mathbb{R}^K$ by $\|c\|_A = \max_{r,k}\{A_{kr}\sum_{j\neq k} A_{jr}|c_j|\}$ and a norm on $\mathbb{R}^R$ by $\|w\|_\infty = \max_r\{|w_r|\}$. Recall that $\delta = (\delta_1, \delta_2, \cdots, \delta_K)$ where $\delta_k = \eta_k\rho_k$.

Theorem 4.1.

(i) The Equations (2.8) have a unique solution c.

(ii) Suppose that $\|\delta\|_A = d < 1$. Then the mapping $f:\mathbb{R}^K \to \mathbb{R}^K$ is a contraction under the norm $\|.\|_A$, and so the sequence $f^m(x)$, $m = 1, 2, \cdots$, converges to c for any $x \in \mathbb{R}^K$. Further, if c' is the unique solution to Equations (2.8) when w is replaced by w',

$$\|c - c'\|_A \leqq \frac{d}{1-d}\|w - w'\|_\infty. \tag{4.1}$$

(iii) If $a < K^{-1}$ then the sequence $f^m_{(a)}(x)$, $m = 1, 2, \cdots$, converges to c for any $x \in \mathbb{R}^K$.

Proof. Rewrite Equations (2.8) in the equivalent form

$$c_k = \eta_k(1 - \eta_k\rho_k)^{-1}(1 - B_k)^{-1} \sum_r A_{kr}\lambda_r\Big(w_r - \sum_j A_{jr}c_j\Big).$$

Let $g_k(c)$ denote the right-hand side of this equation, and write $g = (g_1, g_2, \cdots, g_K)$. Then in matrix form $g(c) = (w - cA)\lambda A^{\mathrm{T}}\gamma$ where γ is the positive diagonal matrix

$$\gamma = \operatorname{diag}(\eta_j(1 - \eta_j\rho_j)^{-1}(1 - B_j)^{-1})_j.$$

Define the positive diagonal matrices $\gamma^{\frac{1}{2}}$, $\gamma^{-\frac{1}{2}}$ componentwise. The relation $c = g(c)$ is equivalent to $c(I + A\lambda A^{\mathrm{T}}\gamma) = w\lambda A^{\mathrm{T}}\gamma$ which in turn is equivalent to

$$c\gamma^{-\frac{1}{2}}(I + \gamma^{\frac{1}{2}}A\lambda A^{\mathrm{T}}\gamma^{\frac{1}{2}}) = w\lambda A^{\mathrm{T}}\gamma^{\frac{1}{2}}.$$

But $(I + \gamma^{\frac{1}{2}}A\lambda A^{\mathrm{T}}\gamma^{\frac{1}{2}})$ is positive definite and hence invertible. Thus the relation

$c = g(c)$ has a unique solution

$$c = w\lambda A^{\mathrm{T}}\gamma^{\frac{1}{2}}(I + \gamma^{\frac{1}{2}}A\lambda A^{\mathrm{T}}\gamma^{\frac{1}{2}})^{-1}\gamma^{\frac{1}{2}}, \tag{4.2}$$

and this is also the unique solution to Equations (2.8).

To prove part (ii), choose $x, x' \in \mathbb{R}^K$. Then

$$f_k(x) - f_k(x') = -\eta_k(1 - B_k)^{-1} \sum_r A_{kr}\lambda_r \sum_{j \neq k} A_{jr}(x_j - x_j').$$

Hence

$$\begin{aligned} |f_k(x) - f_k(x')| &\leqq \eta_k(1 - B_k)^{-1} \sum_r A_{kr}\lambda_r \sum_{j \neq k} A_{jr} |x_j - x_j'| \\ &\leqq \eta_k(1 - B_k)^{-1} \sum_r A_{kr}\lambda_r \|x - x'\|_A \\ &= \eta_k\rho_k \|x - x'\|_A. \end{aligned}$$

Thus

$$\begin{aligned} \|f(x) - f(x')\|_A &\leqq \max_{r,k} \left\{ A_{kr} \sum_{j \neq k} A_{jr}\eta_j\rho_j \right\} \|x - x'\|_A \\ &= \|\delta\|_A \|x - x'\|_A, \end{aligned}$$

and so f is a contraction if $\|\delta\|_A < 1$. Next suppose that c solves Equations (2.8), and c' solves them when w is replaced by w'. Then

$$|c_k - c_k'| \leqq \eta_k\rho_k(\|w - w'\|_\infty + \|c - c'\|_A)$$

and so

$$\|c - c'\|_A \leqq \max_{r,k} \left\{ A_{kr} \sum_{j \neq k} A_{jr}\eta_j\rho_j \right\} (\|w - w'\|_\infty + \|c - c'\|_A).$$

Thus if $\|\delta\|_A = d < 1$ then

$$\|c - c'\|_A \leqq \frac{d}{1 - d} \|w - w'\|_\infty.$$

To prove part (iii) it is enough to establish that the sequence $f_{(a)}^m(x)$, $m = 1, 2, \cdots$, converges, since the limit vector must solve Equations (2.8) by the continuity of $f_{(a)}(.)$. Let ρ be the diagonal matrix $\rho = \mathrm{diag}\,(\rho_j)_j$. Then

$$f(x) = w\lambda A^{\mathrm{T}}\beta^{-1}\eta - xA\lambda A^{\mathrm{T}}\beta^{-1}\eta + x\rho\eta$$

and so

$$f_{(a)}(x) = aw\lambda A^{\mathrm{T}}\beta^{-1}\eta + x[I - a(I - \rho\eta) - aA\lambda A^{\mathrm{T}}\beta^{-1}\eta].$$

The sequence $f_{(a)}^m(x)$, $m = 1, 2, \cdots$, will converge provided the eigenvalues of the

symmetric matrix

$$I - a(I - \rho\eta) - a\eta^{\frac{1}{2}}\beta^{-\frac{1}{2}}A\lambda A^{\mathrm{T}}\beta^{-\frac{1}{2}}\eta^{\frac{1}{2}} = I - a(I - \rho\eta) - a(\lambda^{\frac{1}{2}}A^{\mathrm{T}}\beta^{-\frac{1}{2}}\eta^{\frac{1}{2}})^{\mathrm{T}}(\lambda^{\frac{1}{2}}A^{\mathrm{T}}\beta^{-\frac{1}{2}}\eta^{\frac{1}{2}})$$

lie in the interval $(-1, 1)$. This is ensured if $a < K^{-1}$, since, using the Euclidean norm $\|.\|$ and the Cauchy–Schwarz inequality,

$$\begin{aligned}
\|\lambda^{\frac{1}{2}}A^{\mathrm{T}}\beta^{-\frac{1}{2}}\eta^{\frac{1}{2}}x\| &= \left[\sum_r \left(\sum_j \lambda_r^{\frac{1}{2}}A_{jr}\beta_j^{-\frac{1}{2}}\eta_j^{\frac{1}{2}}x_j\right)^2\right]^{\frac{1}{2}} \\
&\leqq \left[\sum_r \left(\sum_j \lambda_r A_{jr}\beta_j^{-1}\eta_j\right)\|x\|^2\right]^{\frac{1}{2}} \\
&= \left(\sum_j \eta_j\rho_j\right)^{\frac{1}{2}}\|x\| \\
&\leqq K^{\frac{1}{2}}\|x\|.
\end{aligned}$$

Remarks. If R consists of subsets of links then the condition $\|\delta\|_A < 1$ becomes

$$\max_r \max_{k\in r}\left\{\sum_{j\in r-\{k\}} \eta_j\rho_j\right\} < 1.$$

The product $\eta_j\rho_j$ increases to 1 as ρ_j, the traffic offered to link j, increases. The condition $\|\delta\|_A < 1$ might therefore be termed a light traffic condition. It may well be violated in networks with long routes or heavily loaded links. In these circumstances an attempt to solve Equations (2.8) by repeated substitution may fail: it may for example produce a sequence which oscillates away from the solution. Part (iii) of the theorem shows that a sufficient damping of the function f can guarantee convergence. Note, however, that the condition ensuring sufficient damping, that $a < K^{-1}$, involves some non-local knowledge—namely K, the total number of links in the network.

Part (ii) of the theorem shows that if $\|\delta\|_A < 1$ then any changes in the revenue vector w have a limited effect on the cost vector c. For example the inequality (4.1) implies that

$$\sum_j A_{jr}c_j - \sum_j A_{jr}c_j' \leqq \frac{2d}{1-d}\|w - w'\|_\infty \qquad \forall r \in R,$$

and so uniformly bounded changes in w, the revenues, cause uniformly bounded changes in cA, the costs along routes, no matter how large the network. Provided $\|\delta\|_A < 1$ the number of links in the network, K, is not involved.

To investigate this point further it is helpful to imagine a network with a very large number of links and nodes, but where routes are short, and each route overlaps with only a small number of other routes. Observe that Equations (2.8) can be written in the form $c = w\lambda A^{\mathrm{T}}\beta^{-1}\eta - c\Gamma$ where $\Gamma = (A\lambda A^{\mathrm{T}}\beta^{-1} - \rho)\eta$. Thus c has

the representation

$$c = w\lambda A^{\mathrm{T}}\beta^{-1}\eta\Big\{\sum_{n=0}^{\infty}(-\Gamma)^n\Big\} \tag{4.3}$$

provided the summation converges, a condition which is implied by the condition $\|\delta\|_A < 1$. Observe that $\Gamma_{jk} = 0$ if $\sum_r A_{jr}A_{kr} = 0$, that is if there is no route through both links j and k. Similarly, $(\Gamma^n)_{jk} = 0$ if it is not possible to reach link k from link j by a concatenation of n overlapping routes. Thus the higher powers of $(-\Gamma)$ in the representation (4.3) provide the linkage through which changes in the parameter $w_{r'}$ can affect values of c_k for links k widely separated from r'. If the summation in (4.3) does not converge then there is the possibility that revenues may influence costs over arbitrarily great distances. The discussion is complicated by the alternating nature of the series (4.3), and other representations may be more useful. For example, from (4.2) we can write

$$c = w\lambda A^{\mathrm{T}}\gamma^{\frac{1}{2}}\Big\{\sum_{n=0}^{\infty}(-\gamma^{\frac{1}{2}}A\lambda A^{\mathrm{T}}\gamma^{\frac{1}{2}})^n\Big\}\gamma^{\frac{1}{2}}$$

provided the summation converges. Again the (j, k) entry of $(\gamma^{\frac{1}{2}}A\lambda A^{\mathrm{T}}\gamma^{\frac{1}{2}})^n$ is 0 if it is not possible to reach k from j by a concatenation of n overlapping routes.

It is known that long-range order is possible in circuit-switched networks, and forms of phase transition have been deduced from the exact equilibrium distribution (3.4) for certain networks of highly specialized form ([11], [14] consider a regular tree, and [20] considers a two-dimensional lattice). It is notable that for quite general networks the above discussion reduces questions concerning long-range effects to an investigation of a set of linear equations, (2.8). In this connection it is worth pointing out that the relationship between the Erlang fixed-point approximation and the underlying stochastic process described in the introduction precisely parallels the relationship between the mean field or Bragg–Williams approximation of statistical mechanics [4] and the stochastic Ising model [17].

5. A decentralized adaptive routing scheme

In this section we outline how a local estimation of carried loads can be combined with an iterative method of Section 4 to produce a decentralized adaptive routing scheme. As in Section 4 we suppose some limited intelligence is available for each link k and for each route r. Here we shall additionally suppose that limited communication is possible between the intelligences of routes which can be substituted for one another. Also we shall require that the intelligences are capable of sensing carried loads, updating moving averages, and computing certain functions.

Under the Erlang fixed-point approximation the carried traffic on route r is λ_r and the carried traffic through link k is $\rho_k(1 - B_k) = \sum_r A_{kr}\lambda_r$. Recall that $\delta_k = \rho_k\eta_k$.

Thus Equations (2.8) can be written in the form

$$c_k = \delta_k \sum_r A_{kr} \frac{\text{carried traffic on route } r}{\text{carried traffic through link } k} (c_k + s_r) \tag{5.1}$$

$$s_r = w_r - \sum_k A_{kr} c_k. \tag{5.2}$$

The scheme described in this section replaces the model-derived quantities λ_r, ρ_k, B_k with estimates based upon measurements of actual carried loads.

Specifically, suppose there are available measures $X_r(n)$ and $Y_k(n)$ of the carried load on route r and link k respectively over the interval $((n-1)\tau, n\tau)$. These measures might be instantaneous carried loads or mean values found by averaging over the interval. They should bear the relation

$$Y_k(n) = \sum_r A_{kr} X_r(n).$$

From these measures smoothed estimates $x_r(n)$, $y_k(n)$ of the mean carried traffics can be computed by, for example, the simple moving-average iterations:

$$x_r(n+1) = (1-b)x_r(n) + bX_r(n) \tag{5.3}$$

$$y_k(n+1) = (1-b)y_k(n) + bY_k(n). \tag{5.4}$$

Here the constant $b \in (0, 1)$, its precise value reflecting a balance between accuracy of estimation and speed of response. If the traffic offered to the network is stationary then the variance of the estimates $x_r(n)$, $y_k(n)$ about the true mean carried traffics can be made arbitrarily small by choosing a value of b arbitrarily close to 0. However, larger values of b will allow the estimates $x_r(n)$, $y_k(n)$ to respond more quickly if there are changes in the form of the network or in the pattern of traffic offered to it.

Next we describe how the estimate $y_k = y_k(n)$ of the mean carried traffic can be used to construct an estimate of the parameter δ_k appearing in Equation (5.1). Define a function $F_k : [0, C_k] \to [0, 1]$ as follows: set

$$F_k(y_k) = \rho_k[E(\rho_k, C_k - 1) - E(\rho_k, C_k)]$$

where ρ_k satisfies $y_k = \rho_k[1 - E(\rho_k, C_k)]$. Imagine an isolated link of capacity C_k offered Poisson traffic at rate ρ_k. The rate ρ_k is chosen to correspond to the mean carried load y_k, and $F_k(y_k)$ gives the increase in the expected number of calls lost per unit time if a single circuit is removed from the link (cf. expression (3.1)). Put another way, for this model of an isolated link, $F_k(y_k)$ estimates the parameter δ_k. As y_k increases from 0 to C_k the function $F_k(y_k)$ increases from 0 to 1. An example of F_k for a link of capacity $C_k = 100$ is given in Figure 2. The function F_k will be needed by the intelligence of link k in the recursive form of Equation (5.1). The function could be readily tabulated and made available to the intelligence of link k in the form of a look-up table.

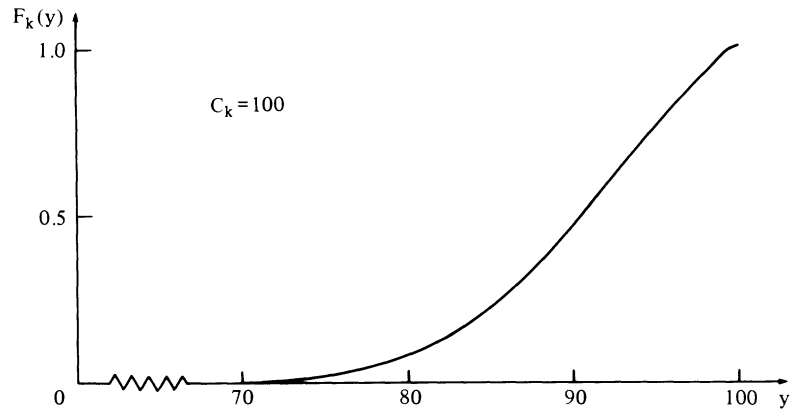

Figure 2. Estimation of δ

Use $s_r(n)$ to denote an estimate of s_r, the surplus value of a call on route r, and use $c_k(n)$ to denote an estimate of c_k, the implied cost of using a circuit on link k. Then these estimates can be updated by the following recursive versions of Equations (5.1) and (5.2):

$$c_k(n+1) = (1-a)c_k(n) + aF_k(y_k(n)) \sum_r A_{kr} \frac{x_r(n)}{y_k(n)} (c_k(n) + s_r(n)) \tag{5.5}$$

$$s_r(n+1) = w_r - \sum_k A_{kr} c_k(n) \tag{5.6}$$

where $a,\ b \in (0, 1)$. Observe that the computation of $s_r(n+1)$ involves only values of $c_k(n)$ on links k which route r passes through, and the computation of $c_k(n+1)$ involves only values of $x_r(n)$, $s_r(n)$ on routes r which pass through link k. The recursion (5.5) for $c_k(n+1)$ corresponds to the damped version $f_{(a)}$ of Equations (2.8) discussed in Section 4. Convergence of the iteration for c, as discussed for $f_{(a)}$ in Section 4, is the main consideration affecting the choice of the constant a. It is not essential that the time interval for the update of the recursions (5.5) and (5.6) be the same as that used for the moving-average estimates (5.3) and (5.4). In particular a shorter time interval for the recursions (5.5) and (5.6) with the same value for the constant a would provide a faster response to drifting estimates $x_r(n)$, $y_k(n)$.

Suppose, then, that implied costs c_k and the associated surplus values s_r have been estimated using the scheme (5.3)–(5.6). Suppose also that the loss probability on route r, L_r, has also been estimated, perhaps by a similar simple moving-average scheme. Consider as an example the network of Figure 1. A call from node α to γ will generate a net expected revenue of $(1 - L_r)s_r$ if offered to route r. If this quantity is negative for any route then that route should not be used: more revenue will be lost elsewhere in the network than can be generated by accepting calls on that route. With this proviso the traffic from node α to node γ should be shared out between the possible routes (possibly by cycling between these routes) so as to

reflect the net expected revenues $(1-L_r)s_r$. If one route has a higher value of $(1-L_r)s_r$ than the others a greater share of the traffic should be directed to this route. The adjustment should be made gradually, since the effect of increasing the traffic offered to a route will be to push up the loss probability of the route and the implied costs c_k along that route, and hence reduce the product $(1-L_r)s_r$. The adjustments could be made automatically or the implied costs c_k, surplus values s_r and loss probabilities L_r could be displayed in a network management centre to inform human operators and enable them to make appropriate routing decisions. It may well be that grade of service as well as revenue generation is important. This could be handled by increasing the nominal parameters w_r for routes handling traffic which is suffering a high loss probability, and allowing the system to adjust. This procedure would help to make explicit the trade-offs between revenue generation and grade of service.

The statistical properties of moving-average estimates such as (5.3) and (5.4) are fairly well understood, and this together with the work of Section 4 allows some degree of theoretical insight into the recursions (5.3)–(5.6). For small enough values of a and b and assuming a stationary carried load of $\tilde{\lambda}_r$ on route r, $r \in R$ the recursions (5.3)–(5.6) are actually solving the relation

$$\tilde{c}_k = F_k\Big(\sum_r A_{kr}\tilde{\lambda}_r\Big) \sum_r \frac{A_{kr}\tilde{\lambda}_r}{\sum_{r'} A_{kr'}\tilde{\lambda}_{r'}} \Big(w_r - \sum_{j \neq k} A_{jr}\tilde{c}_j\Big) \tag{5.7}$$

(in particular the results of Section 4 apply, with δ_k replaced by $\tilde{\delta}_k = F_k(\sum_r A_{kr}\tilde{\lambda}_r)$). It would be interesting to know how the solution $(\tilde{c}_k, k = 1, 2, \cdots, K)$ to Equations (5.7) compares with the solution $(c_k, k = 1, 2, \cdots, K)$ to Equations (2.8) and the collection $(c_r^*, r \in R)$ defined in Section 3.

In view of its important role in the recursions (5.3)–(5.6) and in relation (5.7) it is perhaps worth making some further remarks on our use of the function F. First, observe that our construction of this function places a considerable emphasis on the presumed Poisson-like properties of the traffic offered to link k; this seems not altogether unreasonable in view of the framework within which the results of Section 2 were derived. Second, note that from a series of *independent* observations $Y(1), Y(2), \cdots, Y(N)$ of a random variable Y with the truncated Poisson distribution

$$P\{Y = y\} = \frac{\rho^y}{y!}\Big[\sum_{m=0}^{C} \frac{\rho^m}{m!}\Big]^{-1}, \qquad y = 0, 1, \cdots, C \tag{5.8}$$

a sufficient statistic for the scalar parameter ρ, or its transform $\delta = \rho[E(\rho, C-1) - E(\rho, C)]$, is $T = \sum_{n=1}^{N} Y(n)$. Efficient estimation of δ, in this context, would be based upon T. The recursion (5.4) and the term $F_k(y_k)$ in recursion (5.5) are attempts to use this insight in an adaptive scheme.

Despite the above remarks it should be emphasised that the recursions (5.3)–(5.6)

are but one method of estimating Equations (5.1) and (5.2) or Equation (2.8). For an example of a rather different method, suppose that the circuits of link k are numbered $m = 1, 2, \cdots, C_k$ and that when a call is set up through link k it is allocated the idle circuit with the lowest number. Let $O_k = 0$ or 1 according as circuit number C_k is idle or occupied. Then, under the model of an isolated link,

$$E(O_k) = \rho_k[E(\rho_k, C_k - 1) - E(\rho_k, C_k)] = \delta_k.$$

Replace $F_k(y_k(n))$ in recursion (5.5) with the occupancy measure O_k, and replace the ratio $x_r(n)/y_k(n)$ by an instantaneous measurement of the proportion of traffic through link k on route r. Then the resulting recursion, together with (5.6), could be used to estimate Equations (5.1) and (5.2). Observe that this scheme places more weight on observations collected when the link is near to capacity than does the earlier scheme. There are many other variations which could be suggested, using different forms of measurement, more or less computation, rearrangements of Equation (2.8), and so on. An outstanding question for any method is how accuracy of estimation should be balanced against speed of response to changing traffic conditions.

An issue that has attracted attention in the theory of parallel computation is whether asynchronous, or chaotic, algorithms can be used to solve systems of equations, and there exists theoretical and empirical evidence that such algorithms can perform well ([1], [16], [21]). As presented the scheme (5.3)–(5.6) is synchronous, but if coordination between intelligences is difficult it might well be advantageous to implement an asynchronous version.

6. The shape of the function W

In this section we consider informally the global, rather than local, optimization of the function $W(.\,; C)$. There is no simple result here: the function is rarely concave and may not even be unimodal. However, discussion of a limiting regime and calculations for a specific example give some insight into the general shape of W.

In what region can the vector v be varied? A reasonable assumption is that v can be varied subject to the constraints $v \geqq 0$, $Dv \leqq v$. Here D might be a 0–1 matrix, with the units in a row corresponding to routes which could be substituted for one another.

The limiting regime we consider is that studied in detail in [12]. Replace $(v_r, r \in R)$ and $(C_k, k = 1, 2, \cdots, K)$ by $(Nv_r, r \in R)$ and $(NC_k, k = 1, 2, \cdots, K)$ respectively. Similarly replace v by Nv. Write $\lambda_r(N)$ for the carried traffic on route r in this system. Then [12]

$$\lambda_r(N) = N\lambda_r + o(N), \quad \text{as} \quad N \to \infty$$

where

$$\lambda_r = v_r \prod_j (1 - B_j)^{A_{jr}} \qquad r \in R \tag{6.1}$$

and $(B_1, B_2, \ldots, B_K)$ is any solution to

$$\text{(6.2)} \qquad \begin{cases} \sum_r A_{jr} \nu_r \prod_k (1 - B_k)^{A_{kr}} = C_j & \text{if} \quad B_j > 0 \\ \leqq C_j & \text{if} \quad B_j = 0 \\ B_1, B_2, \cdots, B_K \in [0, 1). \end{cases}$$

Thus

$$W = \sum_r w_r \lambda_r(N)$$

$$= \Big(\sum_r w_r \lambda_r\Big) N + o(N) \quad \text{as} \quad N \to \infty$$

and so to first order our task is to

$$\text{(6.3)} \qquad \begin{aligned} &\text{maximize } w\lambda \\ &\text{subject to (6.1), (6.2), } \nu \geqq 0, D\nu \leqq v. \end{aligned}$$

We may now assume that $\nu = \lambda$, and that $B_k = 0$ for $k = 1, 2, \cdots, K$, for we may continuously transform ν to the value λ along the path

$$\text{(6.4)} \qquad \nu_r(t) = \nu_r \prod_j (1 - B_j)^{A_{jr} t} \quad t \in [0, 1]$$

without affecting the objective function $w\lambda$. Hence (6.3) reduces to

$$\text{(6.5)} \qquad \begin{aligned} &\text{maximize } w\lambda \\ &\text{subject to } A\lambda \leqq C, \lambda \geqq 0, D\lambda \leqq v. \end{aligned}$$

But this is simply a linear program.

The path (6.4) from a general feasible solution ν for problem (6.3) to a feasible solution with $\nu = \lambda$ leaves the first-order objective function $w\lambda$ constant. The higher-order terms will then be important, and they may decrease along the path. Next we indicate that between any two feasible solutions ν, ν' for problem (6.3) giving distinct values $w\lambda < w\lambda'$ for the objective function there is a smooth path along which the objective function strictly increases. Write

$$\lambda'_r = \nu'_r \prod_j (1 - B'_j)^{A_{jr}}, \qquad r \in R$$

for the components of the vector λ' corresponding to ν'. For simplicity assume the vectors $(B_1, B_2, \cdots, B_K)$ and $(B'_1, B'_2, \cdots, B'_K)$ have the same support. Then

$$\nu_r(t) = (\lambda'_r t + \lambda_r(1 - t)) \prod_j (1 - B_j)^{-A_{jr} t^x} (1 - B_j)^{-A_{jr}(1-t)^x}, \qquad t \in (0, 1)$$

gives a parametrization of a smooth path from $\nu(0) = \nu$ to $\nu(1) = \nu'$ along which $w\lambda(t)$ increases linearly with t. By choosing the constant x large enough we can

ensure the path never violates the constraint $Dv \leqq v$, and so remains within the feasible region.

We now summarize the discussion of this section. Under the limiting regime described the function W takes the linear form $W = wv$ on the region $v \geqq 0$, $Av \leqq C$, $Dv \leqq v$. Outside this region W takes a constant value along paths of the form (6.4). The global maxima of W lie along the paths of this form which connect to points in the region $v \leqq 0$, $Av \leqq C$, $Dv \leqq v$ where wv is maximized. There are no local maxima of W other than these global maxima.

The limiting regime certainly loses a great deal of the detail of the function $W(v; C)$, but does expose an underlying structure (6.5) which is essentially a deterministic multi-commodity network flow problem. Higher order perturbations of this linear structure may produce convex or concave local features, and these features may become important when the slope of the linear structure is not pronounced. We illustrate these points by a numerical investigation of a very simple network.

Our example is the symmetric star network illustrated in Figure 3. In this network there are K links each of capacity C. A call may use one circuit from link k, such calls arriving at rate v_1 for $k = 1, 2, \cdots, K$; or a call may use one circuit from each of links j and k, such calls arriving at rate $v_2(K-1)^{-1}$ for $j \neq k$, $j, k = 1, 2, \cdots, K$. Equations (1.1) reduce to a single equation

$$B = E(v_1 + v_2(1-B), C) \tag{6.6}$$

for the common link blocking probability B. Let w_1 and w_2 be the worth of calls over one and two links respectively. Then the function $W(v; C)$ reduces to

$$W = K[w_1 v_1(1-B) + \tfrac{1}{2} w_2 v_2 (1-B)^2]. \tag{6.7}$$

Figure 4 illustrates the function W when link capacity is $C = 10$ and $v_1 + v_2 = 10$. Observe that in Figure 4(a) the cross-sections are almost linear: in fact those for $w_2 = 0$ and 1 are concave, while those for $w_2 = 2$, 3 and 4 are convex. Convexity is

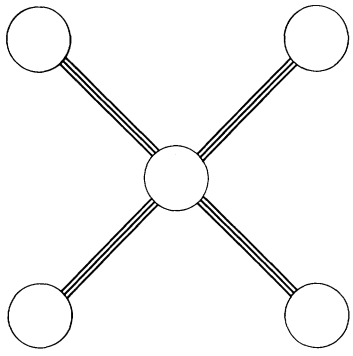

Figure 3. Star network

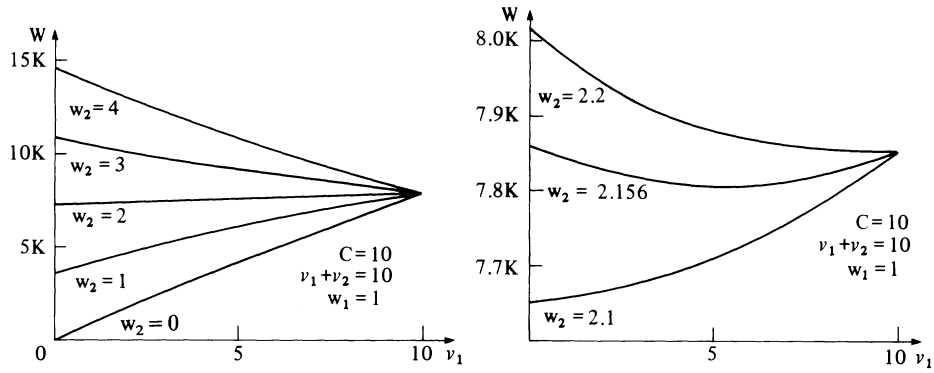

Figure 4. Cross-sections of W

clearer on the magnified W scale of Figure 4(b); and we observe that for a small range of w_2 values the cross-section is bimodal.

It is known ([37], [39]) that as $K \to \infty$ the solution B to the Erlang fixed-point Equation (6.6) and the ratio W/K obtained from (6.7) become exact (i.e. the discrepancies $B - B^*$, $(W - W^*)/K$ approach 0 as $K \to \infty$, where B^* and W^* are as defined in Section 3). Hence the features illustrated in Figure 4 and discussed above are *not* artefacts of the Erlang fixed-point approximation.

It is interesting to conjecture on the behaviour of the routing scheme of Section 5 in the event that the function W^* has local maxima other than the global optimum. Work on the equilibrium behaviour of probabilistic hill-climbing algorithms ([22], [26]) is relevant and suggestive. The stochastic fluctuations inherent in the scheme of Section 5 may allow it to escape from the region around a non-optimal local maximum, and if the global maximum is sufficiently greater than other maxima then the equilibrium distribution for the scheme should assign a relatively small mass to the regions around the other maxima.

7. Alternative routing

In this section we outline how the methods of this paper can deal with alternative routing, where a call which is blocked on a route may be allowed to try again on another route. The natural extension of the Erlang fixed-point Equations (1.1) to the case of alternative routing is well known in the literature (see, for example, [18]). We shall find it convenient to consider a more general case still, where the arrival rate for a route is allowed to depend arbitrarily upon which links are full. Let $b = (b_1, b_2, \cdots, b_K)$ denote the blocking configuration of the links: $b_k = 0$ or 1 according as link k has free circuits or not. Write:

$$p(b, B) = \prod_{k=1}^{K} B_k^{b_k}(1 - B_k)^{1-b_k}. \tag{7.1}$$

Thus $p(b, B)$ is the probability of blocking configuration b under the assumption that links $1, 2, \cdots, K$ block independently, link k blocking with probability B_k. Write $\lambda_r(b)$ for the traffic offered to route r when the blocking configuration is b. Insist that

$$\lambda_r(b) = 0 \quad \text{if} \quad \prod_k (1 - b_k)^{A_{kr}} = 0.$$

This will ensure that $\lambda_r(b)$ is also the rate of accepted traffic on route r when the blocking configuration is b. The total carried traffic on route r is thus

$$\lambda_r = \sum_b p(b, B)\lambda_r(b). \tag{7.2}$$

Write λ for the entire collection $(\lambda_r(b), \quad b \in \{0, 1\}^K, r \in R)$. Define $B = (B_1, B_2, \cdots, B_K)$ as a solution to the equations

$$B_k = E\Big(\sum_r A_{kr}(1 - B_k)^{-1} \sum_b p(b, B)\lambda_r(b), C_k\Big), \qquad k = 1, 2, \cdots, K. \tag{7.3}$$

Observe that we recover the model of Section 2 if we set

$$\lambda_r(b) = \nu_r \prod_k (1 - b_k)^{A_{kr}}.$$

However, here we regard λ as the free parameter, with B determined as a function of λ and C. By the Brouwer fixed-point theorem, Equations (7.3) have a solution $B = B(\lambda; C)$. The solution may not be unique [27], in which case we allow $B(\lambda; C)$ to be any one of the multiple solutions. The Implicit Function Theorem [33] will permit $B(\lambda; C)$ to be chosen as a locally differentiable function of λ except at a rather special set of points (cf. [12], Section 5).

Write

$$W(\lambda; C) = \sum_r w_r\lambda_r \tag{7.4}$$

where λ_r is given by (7.2) and $p(b, B)$ is given by (7.1) with $B = B(\lambda; C)$. Again let $\eta_k = E(\rho_k, C_k) - E(\rho_k, C_k - 1)$, where ρ_k is now the first argument of Erlang's formula in Equation (7.3).

Theorem 7.1. Where the derivatives exist,

$$\frac{d}{d\lambda_r(b)} W(\lambda; C) = p(b, B)\Big(w_r - \sum_j A_{jr}c_j\Big)$$

and

$$\frac{d}{dC_j} W(\lambda; C) = c_j$$

for $c = (c_1, c_2, \cdots, c_K)$ a solution to

$$c_k = \eta_k \sum_b p(b, B) \sum_r \lambda_r(b)\left(\frac{1-b_k}{1-B_k} - \frac{b_k}{B_k}\right)\left(w_r - \sum_{j \neq k} A_{jr} c_j\right). \tag{7.5}$$

Proof. The proof parallels closely the derivations of Lemmas and Theorems 2.3–2.9.

To illustrate the application of Theorem 7.1 we consider in more detail two examples of particular forms of alternative routing.

Example 1. In this example suppose the label r fixes a pair $(r(1), r(2))$ of routes between a source and destination node. Regard $r(1)$ as the route first attempted by a call labelled r; if this route is blocked the route $r(2)$ is attempted; if both routes are blocked then the call is lost. As before assume a call with label r which is carried generates a mean revenue of w_r. For simplicity assume that the routes $r(1)$ and $r(2)$ do not have any links in common. Let $\nu = (\nu_r, r \in R)$ where ν_r is the arrival rate of calls labelled r. In the formalism of this section the parameter $\lambda = \lambda(\nu)$ is determined by

$$\lambda_{r(1)}(b) = \nu_r \prod_j (1-b_j)^{A_{jr(1)}}$$

$$\lambda_{r(2)}(b) = \nu_r\left[1 - \prod_j (1-b_j)^{A_{jr(1)}}\right] \prod_j (1-b_j)^{A_{jr(2)}}.$$

Write $W(\nu; C)$ for the function (7.4) evaluated at $\lambda = \lambda(\nu)$. Then from Theorem 7.1 it follows that

$$\begin{aligned} \frac{d}{d\nu_r} W(\nu; C) = &\left[\prod_j (1-B_j)^{A_{jr(1)}}\right]\left(w_r - \sum_j A_{jr(1)} c_j\right) \\ &+ \left[1 - \prod_j (1-B_j)^{A_{jr(1)}}\right]\left[\prod_j (1-B_j)^{A_{jr(2)}}\right]\left(w_r - \sum_j A_{jr(2)} c_j\right) \end{aligned} \tag{7.6}$$

where

$$\begin{aligned} c_k = \eta_k \sum_r \Bigg\{ &A_{kr(1)} \nu_r \left[\prod_{j \neq k} (1-B_j)^{A_{jr(1)}}\right]\left(w_r - \sum_{j \neq k} A_{jr(1)} c_j\right) \\ &+ A_{kr(2)} \nu_r \left[1 - \prod_j (1-B_j)^{A_{jr(1)}}\right]\left[\prod_{j \neq k} (1-B_j)^{A_{jr(2)}}\right]\left(w_r - \sum_{j \neq k} A_{jr(2)} c_j\right) \\ &- A_{kr(1)} \nu_r \left[\prod_{j \neq k} (1-B_j^{A_{jr(1)}})\right]\left[\prod_j (1-B_j)^{A_{jr(2)}}\right]\left(w_r - \sum_j A_{jr(2)} c_j\right)\Bigg\}. \end{aligned} \tag{7.7}$$

Write

$$s_{r(2)} = w_r - \sum_j A_{jr(2)} c_j \tag{7.8}$$

$$s_{r(1)} = w_r - \sum_j A_{jr(1)} c_j - (1 - L_{r(2)}) s_{r(2)} \tag{7.9}$$

where

$$1 - L_{r(i)} = \prod_j (1 - B_j)^{A_{jr(i)}}.$$

Then Equations (7.6) and (7.7) can be written as

$$\frac{d}{dv_r} W(v; C) = (1 - L_{r(1)})\left(w_r - \sum_j A_{jr(1)} c_j\right) + L_{r(1)}(1 - L_{r(2)})\left(w_r - \sum_j A_{jr(2)} c_j\right) \tag{7.10}$$

and

$$c_k = \eta_k (1 - B_k)^{-1} \sum_r v_r \{A_{kr(1)}(1 - L_{r(1)})(c_k + s_{r(1)}) + A_{kr(2)} L_{r(1)}(1 - L_{r(2)})(c_k + s_{r(2)})\}. \tag{7.11}$$

Equation (7.10) shows that the cost of increasing the offered traffic with label $r = (r(1), r(2))$ can be assessed by the following rule of thumb: an additional call with this label will be accepted by route $r(1)$ with probability $1 - L_{r(1)}$ and by route $r(2)$ with probability $L_{r(1)}(1 - L_{r(2)})$; if accepted it will earn w_r directly, but at a cost c_k for each link on the accepted route. Equation (7.11) can be interpreted as earlier, but now the surplus values $s_{r(1)}$ for first-choice routes are calculated by (7.9), an equation with an additional term. The additional term can be interpreted as follows: a call blocked on route $r(1)$ is not necessarily lost; with probability $1 - L_{r(2)}$ it will be carried on its second-choice route $r(2)$, generating there a surplus value of $s_{r(2)}$.

The above interpretations suggest various generalizations of the adaptive routing scheme described in Section 5. To estimate iteratively Equations (7.8) and (7.9) we need only replace Equation (5.6) by

$$s_{r(2)}(n+1) = w_r - \sum_j A_{jr(2)} c_j(n)$$

$$s_{r(1)}(n+1) = w_r - \sum_j A_{jr(1)} c_j(n) - (1 - L_{r(2)}(n)) s_{r(2)}(n+1)$$

and allow r to range over all possible first or second choice routes $r(1), r(2)$ in Equations (5.3) and (5.5). As earlier the costs c_k and surplus values $s_{r(1)}, s_{r(2)}$ estimated by the scheme can be used for various purposes. Additionally they can now be used to decide on whether to reroute blocked calls. For example, if $s_{r(2)}$ becomes negative while $s_{r(1)}$ remains positive the single route $r(1)$ is preferred to the pair $r = (r(1), r(2))$: a call blocked on the first-choice route $r(1)$ should be discarded rather than offered to the second-choice route $r(2)$.

Example 2. In this example suppose the label r fixes a pair $(\phi(r), \psi(r))$, where $\phi(r)$ and $\psi(r)$ are both sets of links. Interpret $\psi(r)$ as the path used by a call on route r, and $\phi(r)$ as a set of links each of which must be blocked in order for this path to be attempted. For example, suppose that in the network of Figure 1 a call between α and γ tries first the path $\{1, 2\}$; if link 1 is blocked it then tries the path $\{4, 5, 6\}$, while if link 1 is free and link 2 is blocked it tries the path $\{1, 3, 6\}$. This pattern of choices can be represented by the following three pairs $(\phi(r), \psi(r))$:

$$(\phi, \{1, 2\}), \quad (\{1\}, \{4, 5, 6\}), \quad (\{2\}, \{1, 3, 6\}).$$

In the formalism of this section

$$\lambda_r(b) = \nu_r\Big(\prod_{j\in\phi(r)} b_j\Big) \prod_{k\in\psi(r)} (1 - b_k)$$

and

$$\lambda_r = \nu_r\Big(\prod_{j\in\phi(r)} B_j\Big) \prod_{k\in\psi(r)} (1 - B_k)$$

where ν_r is the arrival rate at the network of calls potentially served by route r. The Equations (7.5) of Theorem 7.1 can be rewritten in the form

$$c_k = \eta_k\Big[(1 - B_k)^{-1} \sum_{r:k\in\psi(r)} \lambda_r(s_r + c_k) - B_k^{-1} \sum_{r:k\in\phi(r)} \lambda_r s_r\Big]$$

$$s_r = w_r - \sum_{k\in\psi(r)} c_k.$$

In a recent study [6] of a network with about 50 nodes, 100 links and 5000 routes (including multiple alternative routes) it was found that derivatives of W could be calculated in seconds (on an Acorn Cambridge Workstation) by using a simple iterative scheme to solve these equations.

Acknowledgements

I am grateful to Peter Key, Peter Whittle and the referee for a number of valuable comments on earlier versions of this paper.

References

[1] BAUDET, G. M. (1978) Asynchronous iterative methods for multiprocessors. *J. Assoc. Comp. Mach.* **25,** 226–244.

[2] BENEŠ, V. E. (1966) Programming and control problems arising from optimal routing in telephone networks. *Bell Syst. Tech. J.* **9,** 1373–1438.

[3] BROCKMEYER, E., HALSTROM, H. L. AND JENSEN, A. (1948) *The Life and Works of A. K. Erlang.* Academy of Technical Sciences, Copenhagen.

[4] BURLEY, D. M. (1972) Closed form approximations for lattice systems. In *Phase Transitions and Critical Phenomena,* Vol. 2, ed. C. Domb and M. S. Green. Academic Press, London, 329–374.

[5] COOPER, R. B. AND KATZ, S. S. (1964) Analysis of alternate routing networks with account taken of the nonrandomness of overflow traffic. Bell Laboratories.

[6] COPE, G. A. AND KELLY, F. P. (1986) The use of implied costs for dimensioning and routing. Report prepared by Stochastic Networks Group, Cambridge, for British Telecom Research Laboratories.

[7] GALLAGER, R. G. (1977) A minimum delay routing algorithm using distributed computation. *IEEE Trans. Comm.* **25,** 73–85.

[8] GIBBENS, R. J. AND KELLY, F. P. (1986) Dynamic routing in fully connected networks.

[9] HARRIS, R. J. (1973) Concepts of optimality in alternate routing networks. *7th Int. Teletraffic Cong.*

[10] KELLY, F. P. (1979) *Reversibility and Stochastic Networks.* Wiley, Chichester.

[11] KELLY, F. P. (1985) Stochastic models of computer communication systems. *J. R. Statist. Soc.* **B47,** 379–395.

[12] KELLY, F. P. (1986) Blocking probabilities in large circuit-switched networks. *Adv. Appl. Prob.* **18,** 473–505.

[13] KELLY, F. P. (1986) Blocking and routing in circuit-switched networks. In *Teletraffic Analysis and Computer Performance Evaluation,* ed. O. J. Boxma, J. W. Cohen and H. C. Tijms, Elsevier, Amsterdam, 37–45.

[14] KELLY, F. P. (1986) Instability in a communications network. In *Fundamental Problems in Communication and Computation,* ed. T. Cover and B. Gopinath. Springer-Verlag, Berlin.

[15] KELLY, F. P. (1987) One-dimensional circuit-switched networks. *Ann. Prob.* **15,** 1166–1179.

[16] KUNG, H. T. (1976) Synchronized and asynchronous parallel algorithms for multiprocessors. In *Algorithms and Complexity,* ed. J. F. Traub, Academic Press, New York, 153–200.

[17] LIGGETT, T. M. (1985) *Interacting Particle Systems.* Springer-Verlag, New York.

[18] LIN, P. M., LEON, B. J. AND STEWART, C. R. (1978) Analysis of circuit-switched networks employing originating-office control with spill-forward. *IEEE trans. Comm.* **26,** 754–765.

[19] LOTTIN, J. AND FORESTIER, J. P. (1980) A decentralized control scheme for large telephone networks. *Proc. IFAC Symp. on Large Scale Systems Theory and Applications,* Toulouse, France.

[20] LOUTH, G. M. (1986) Phase transition in a circuit-switched network.

[21] LUBACHEVSKY, B. AND MITRA, D. (1986) A chaotic, asynchronous algorithm for computing the fixed point of a nonnegative matrix of unit spectral radius. *J. Assoc. Comp. Mach.* **33,** 130–150.

[22] LUNDY, M. AND MEES, A. (1986) Convergence of the annealing algorithm. *Math. Programming* **34,** 111–124.

[23] MALINVAUD, E. (1972) *Lectures on Microeconomic Theory.* North-Holland, Amsterdam.

[24] MASON, L. G. (1985) Equilibrium flows, routing patterns and algorithms for store-and-forward networks. *Large Scale Systems* **8.**

[25] MITRA, D. (1987) Asymptotic analysis and computational methods for a class of simple circuit-switched networks with blocking. *Adv. Appl. Prob.* **19,** 219–239.

[26] MITRA, D., ROMEO, F. AND SANGIOVANNI-VINCENTELLI, A. (1986) Convergence and finite-time behaviour of simulated annealing. *Adv. Appl. Prob.* **18,** 747–771.

[27] NAKAGOME, Y. AND MORI, H. (1973) Flexible routing in the global communication network. *7th Int. Teletraffic Cong.*

[28] NARENDRA, K. S., WRIGHT, E. A. AND MASON, L. G. (1977) Applications of learning automata to telephone traffic routing and control. *IEEE Trans. Syst., Man Cybernet.* **7,** 785–792.

[29] NARENDRA, K. S. AND MARS, P. (1983) The use of learning algorithms in telephone traffic routing —a methodology. *Automatica* **19,** 495–502.

[30] OTT, T. J. AND KRISHNAN, K. R. (1985) State dependent routing of telephone traffic and the use of separable routing schemes. *11th Int. Teletraffic Cong.* (ed. M. Akiyama), Elsevier, Amsterdam.

[31] PIORO, M. AND WALLSTROM, B. (1985) Multihour optimization of non-hierarchical circuit-switched communication networks with sequential routing. *11th Int. Teletraffic Cong.* (ed. M. Akiyama), Elsevier, Amsterdam.

[32] SEVCIK, K. C. AND MITRANI, I. (1981) The distribution of queueing network states at input and output instants. *J. Assoc. Comput. Mach.* **28,** 358–371.

[33] SPIVAK, M. (1965) *Calculus on Manifolds.* Benjamin, New York.

[34] STIDHAM, S. (1984) Optimal control of admission, routing, and service in queues and networks of

queues: a tutorial review. In *Analytical and Computational Issues in Logistics R&D,* U.S. Army Research Office, 330–377.

[35] STIDHAM, S. (1985) Optimal control of admission to a queueing system. *IEEE Trans. Autom. Congr.* **30,** 705–713.

[36] SYSKI, R. (1960) *Introduction to Congestion Theory in Telephone Systems.* Oliver and Boyd, London.

[37] WHITT, W. (1985) Blocking when service is required from several facilities simultaneously. *A.T. & T. Tech. J.* **64,** 1807–1856.

[38] YOUNG, H. P. (1985) Cost allocation. In *Proc. Symp. Appl. Math.* **33,** Amer. Math. Soc., 69–94.

[39] ZIEDINS, I. (1985) Blocking in Queueing and Loss Systems. Knight Prize Essay, University of Cambridge.

[40] ZIEDINS, I. (1987) Quasi-stationary distributions and one-dimensional circuit-switched networks. *J. Appl. Prob.* **24,** 965–977.