Performance Limits for Channelized Cellular Telephone Systems

Robert J. McEliece and Kumar N. Sivarajan

***Abstract*—In this paper, we study the performance of channel assignment algorithms for "channelized" (e.g., FDMA or TDMA) cellular telephone systems, via mathematical models, each of which is characterized by a pair (H, p), where H is a hypergraph describing the channel reuse restrictions, and p is a probability vector describing the variation of traffic intensity from cell to cell. For a given channel assignment algorithm, we define $T(r)$ to be the amount of carried traffic, as a function of the offered traffic, where both r and $T(r)$ are measured in Erlangs per channel. We show that for a given H and p, there exists a function $T_{H,p}(r)$, which can be computed by linear programming, such that for every channel assignment algorithm, $T(r) \leq T_{H,p}(r)$. Moreover, we show that there exist channel assignment algorithms whose performance approaches $T_{H,p}(r)$ arbitrarily closely as the number of channels increases. As a corollary, we show that for a given (H, p) there is a number r_0, which also can be computed by linear programming, such that if the offered traffic exceeds r_0, then for any channel assignment algorithm, a positive fraction of all call requests must be blocked, whereas if the offered traffic is less than r_0, all call requests can be honored, if the number of channels is sufficiently large. We call r_0, whose units are Erlangs per channel, the *capacity* of the cellular system.**

***Index Terms*— Cellular, performance limits, channel assignment, capacity.**

I. Introduction and Summary

One of the fastest growing segments of the telecommunications industry is cellular telephony. Nearly every major U.S. city is currently covered by a network of localized "cells," which allow mobile customers to be connected, via short-range radio links, to the international wire-line telephone network. In most of these cellular systems, a mobile user requesting service is assigned one of a number of nonoverlapping radio-frequency channels, each of which is characterized by either a fixed frequency-slot or time-slot allocation, or both [6], [12]. The key resources are the cells, which are the discrete locations in which call requests arise, and the channels, which are used to service these requests. Such systems are interesting for the theoretician, and lucrative for the service providers, largely because of the possibility of *channel reuse*, which allows a channel to be used simultaneously in several cells, under certain circumstances.

This paper represents an attempt to make an information-theoretic study of such "channelized" cellular telephone systems, and in particular to identify the ultimate limits of such systems, as measured by the maximum possible number of calls that can be carried simultaneously. Of course, we cannot deal directly with real cellular telephone systems, but must base our studies on certain models for such systems, models which are mathematically tractable but which we hope capture the essence of the real systems.[1] We should say at the beginning, however, that our models cannot be used to study nonchannelized cellular telephone systems, such as the CDMA systems proposed by Gilhousen *et al.* [5] or the frequency-hopped system proposed by Wallace [20].

In our models, we assume that there is a finite set of N *cells*, and an underlying offered traffic model for each cell, which is independent from cell to cell. The N cells share a common set of n channels, and the intensity of the offered traffic is measured in *Erlangs per channel*. Thus, if A_i denotes the expected number of calls that would be in progress in cell i at a given time if all call requests in that cell could be honored, then the intensity of the offered traffic in cell i is A_i/n Erlangs per channel. The overall offered traffic is then $A = \sum_i A_i$ Erlangs, and the overall intensity of offered traffic is $r = A/n$ Erlangs per channel. The ratio $p_i = A_i/A$ represents the fraction of the total traffic present in cell i, and we call the vector $p = (p_1, p_2, \cdots, p_N)$ the *traffic pattern*. In what follows, we shall assume the traffic pattern p is fixed, while the offered traffic intensity r may vary.

We further assume that when a call request arrives in a particular cell, it is either assigned to one of the n channels or blocked by a *channel assignment algorithm*. (A blocked call disappears from the system.) The channels assigned to calls cannot be arbitrary; they must satisfy certain *channel reuse constraints*, which can be described as follows. There is a fixed collection $E = \{E_1, E_2, \cdots, E_K\}$ of subsets of cells, called "forbidden" subsets. It is illegal for a given channel to be in use simultaneously in each cell of a forbidden set.[2] As

Manuscript received December 29, 1991; revised December 15, 1992. This work was supported by a grant from GTE Laboratories. The work of R. J. McEliece was supported by the AFOSR under Grant 91-0037 and a grant from Pacific Bell. This paper was presented in part at the 1990 Allerton Conference on Communications, Control, and Computing and at the 3rd IEE Conference on Telecommunications, Edinburgh, Scotland, 1991.

R. J. McEliece is with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125.

K. N. Sivarajan is with the IBM T. J. Watson Research Center, Yorktown Heights, NY 10598.

IEEE Log Number 9215198.

[1] Although we shall not emphasize the point, our models can also be thought of as models for multitasking concurrent computation, in which the "cells" correspond to various types of tasks and the "channels" correspond to the (identical) processors available to perform these tasks, it being understood that a given processor may be able to handle different tasks simultaneously.

[2] The assumption of fixed forbidden sets means, in effect, that we are restricting our attention to channel assignment algorithms which make decisions based only on the knowledge of which channels are in use in which cells, and not on more detailed information about the location of the mobile users within the cells, or current propagation conditions.

we will discuss in the next section, a finite set together with a collection of subsets is called a hypergraph, and so we call our cellular systems hypergraph systems. If the hypergraph associated with the system is denoted by H and the traffic pattern by p, we will call our system the (H, p) system.

For a given system (H, p), and with a given channel assignment algorithm in mind, we define the *carried traffic function* $T(r)$ to be the expected number of calls per channel that the channel assignment algorithm will permit to be in progress at a given time as a function of the offered traffic intensity r. If the offered traffic is small, one expects the carried traffic $T(r)$ to be nearly equal to r for an intelligent channel assignment algorithm. However, as r increases, the system will become overloaded, and one expects the difference between r and $T(r)$ to become pronounced. In this paper, we will show that it is possible to give a precise "asymptotic" description of the behavior of $T(r)$ for the best possible channel assignment algorithms for all values of $r \geq 0$. Indeed, for a given system (H, p), we shall define a function $T_{H,p}(r)$, which can be computed by linear programming, and which has the following significance. If $T(r)$ denotes the carried traffic function for any channel assignment algorithm for the (H, p) system, then $T(r) \leq T_{H,p}(r)$. On the other hand, in the limit as the number of available channels becomes large, there exist channel assignment algorithms for the (H, p) system whose carried traffic functions are arbitrarily close to $T_{H,p}(r)$. Thus, $T_{H,p}(r)$ can fairly be called *the* carried traffic function for the (H, p) system. The following two examples, which will be referred to throughout the paper, illustrate our general results.

Example 1.1: Consider the seven-cell system shown in Fig. 1(a). There are 14 minimal forbidden sets, viz. the 12 pairs of adjacent cells, and the sets $\{1, 3, 5\}$ and $\{2, 4, 6\}$. (From the definition of a forbidden subset, any superset of a forbidden subset is also a forbidden subset. Therefore, we need only consider minimal forbidden subsets.) The traffic pattern is $(p_1, \cdots, p_7) = (1/8, 1/8, 1/8, 1/8, 1/8, 1/8, 1/4)$. We shall see below that the $T_{H,p}(r)$ curve for this system is as shown in Fig. 1(b). □

Example 1.2: Consider the 19-cell system shown in Fig. 2(a). In an attempt to model a real system in which signals propagate isotropically and attenuate according to an inverse fourth-power law, we define a forbidden set E to be any subset of cells such that $\sum_{v \in E-\{u\}} d(u, v)^{-4} \geq 3/8$, for all $u \in E$, where $d(u, v)$ is the Euclidean distance between the center of cell u and the center of cell v, where the distance between the centers of adjacent cells is defined as 1. It turns out that there are 93 distinct, minimal, forbidden subsets of this kind: the 42 pairs of adjacent cells, and also 45 subsets of cardinality 5, and 6 of cardinality 6. For this example, we assume uniform traffic, i.e., the traffic pattern is $p = (1/19, 1/19, \cdots, 1/19)$. We shall see below that the $T_{H,p}(r)$ curve for this system is as shown in Fig. 2(b). □

The organization of the paper is as follows. In Section II, we present some preliminary material about hypergraphs, including a discussion of what we call *random hypergraph multicolorings*, a notion which is central to our analysis of channel assignment algorithms. In Section III, we will show that for any channel assignment algorithm, the carried-traffic function must satisfy $T(r) \leq T_0(r)$, where $T_0(r)$ is a simple function that can be computed by linear programming. In Section IV, on the other hand, we will give an asymptotic

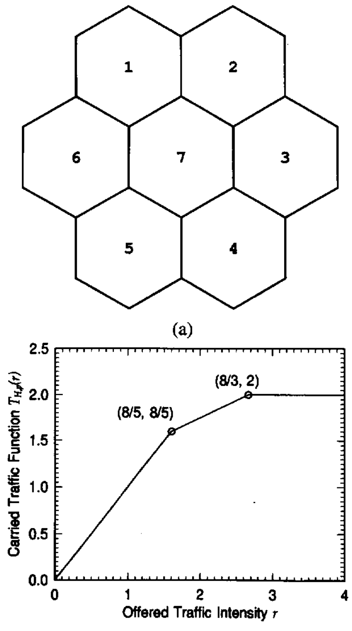

Fig. 1. (a) A seven cell system. There are 14 forbidden subsets, the 12 pairs of adjacent cells, and the sets $\{1,3,5\}$ and $\{2,4,6\}$. (b) The carried traffic function $T_{H,p}(r) = \min(r, \frac{3}{8}r + 1, 2)$ for the 7-cell (H, p) system of (a).

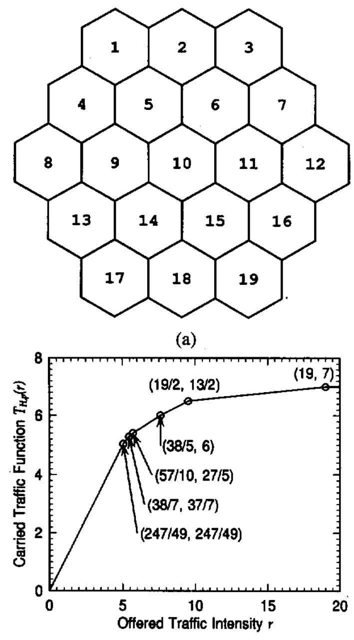

Fig. 2. (a) A 19-cell system. There are 93 forbidden subsets: the 42 pairs of adjacent cells, 45 subsets of size 5, and 6 of size 6. (b) The carried traffic function $T_{H,p}(r) = \min(r, \frac{12}{19}r + \frac{13}{7}, \frac{8}{19}r + 3, \frac{6}{19}r + \frac{18}{5}, \frac{5}{19}r + 4, \frac{1}{19}r + 6, 7)$ for the 19-cell (H, p) system of (a).

analysis of a class of "fixed" channel assignment algorithms, and show that in the limit as $n \to \infty$, these algorithms achieve carried traffic functions that are at least as large as $T_1(r)$, another simple function that can be computed by linear programming. In Section V, we will show that $T_0(r) = T_1(r)$. This common value, denoted by $T_{H,p}(r)$, is the function referred to above. In Section V, we will also describe some of the most important properties of the function $T_{H,p}(r)$, and identify the "most favorable" traffic patterns for a given hypergraph H. In Section VI, we will compare the performance of two specific channel assignment algorithms to our asymptotic performance limits. Finally, in Section VII, we will discuss the extension of our results to more general traffic models, including models which allow calls to be "handed off" from one cell to another. We will also show that for a given system (H, p), even for these more general traffic models, there is a quantity r_0, which we call the *capacity* of the system, such that if the offered traffic intensity exceeds r_0, then for any channel assignment algorithm, a positive fraction of all call requests must be blocked, while if the offered traffic intensity is less than r_0, all call requests can be honored if the number of channels is sufficiently large.

II. Hypergraph Multicolorings and Random Hypergraph Multicolorings

A *hypergraph* H is a pair (V, E), where $V = \{v_1, v_2, \cdots, v_N\}$ is a finite set of *vertices*, and $E = \{E_1, E_2, \cdots, E_K\}$ is a finite collection of subsets of V, called the *edges* of H. (See Berge [1] as a general reference for hypergraphs. Note that an ordinary graph is just a hypergraph in which every edge has two elements.) We shall assume that each edge of H contains at least two vertices. An *independent set* for H is a set of vertices which contains no edge as a subset. A *maximal independent set* is an independent set which is not a proper subset of any other independent set. We assume H has M maximal independent sets $\{V_1, V_2, \cdots, V_M\}$. For future reference, we also define the *indicator set* I_j for the maximal independent set V_j as $I_j = \{i\colon v_i \in V_j\}$, and the incidence matrix $A = (a_{ij})$ as

$$a_{ij} = \begin{cases} 1 & \text{if } v_i \in V_j \\ 0 & \text{if } v_i \notin V_j. \end{cases}$$

The hypergraph H can be reconstructed from A, and for our purposes, it is the preferred representation of H.

If $w = (w_1, w_2, \cdots, w_N)$ is a list of real numbers assigned to the vertices of H, define the *$v - m$ transform* (vertex-maximal independent set) of w as $W = (W_1, W_2, \cdots, W_M)$, where $W = wA$, i.e.,

$$W_j = \sum_{i:\, i \in I_j} w_i = \sum_{i=1}^{N} w_i a_{ij} \qquad \text{for } j = 1, 2, \cdots, M. \tag{2.1}$$

For example, if $w = (1, 1, \cdots, 1)$, then $W = (N_1, N_2, \cdots, N_M)$, where N_j is the size of the jth maximal independent set V_j. Similarly, if $X = (X_1, X_2, \cdots, X_M)$ is a list of real numbers assigned to the maximal independent sets of H, the *m-v transform* (maximal independent set-vertex transform) of X is $x = (x_1, x_2, \cdots, x_N)$, where $x^T = AX^T$, i.e.,

$$x_i = \sum_{j:\, i \in I_j} X_j = \sum_{j=1}^{M} X_j a_{ij} \qquad \text{for } i = 1, 2, \cdots, N. \tag{2.2}$$

For example, if $X = (1, 1, \cdots, 1)$, then $x = (M_1, M_2, \cdots, M_N)$, where M_i denotes the number of maximal independent sets containing vertex v_i.

Example 2.1: The seven-cell system of Fig. 1(a) can be viewed as a hypergraph with seven vertices and 14 edges, viz. the 12 pairs of adjacent cells, together with $\{1, 3, 5\}$ and $\{2, 4, 6\}$. This hypergraph has exactly ten maximal independent sets, viz. $V = \{V_1, V_2, \cdots, V_{10}\}$, where the corresponding adjacency matrix A is given by

$$A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \begin{array}{c} \begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array}. \tag{2.3}$$

Thus, e.g., $I_5 = \{2, 5\}$ and $I_{10} = \{7\}$. Note that the N_j's are the column sums of A and the M_i's are the row sums of A. In the example, $(N_1, N_2, \cdots, N_{10}) = (2, 2, 2, 2, 2, 2, 2, 2, 2, 1)$ and $(M_1, M_2, \cdots, M_7) = (3, 3, 3, 3, 3, 3, 1)$. □

Example 2.2: The 19-cell system of Fig. 2(a) can be viewed as a hypergraph with 19 vertices and 93 edges, as we saw in Example 1.2. It turns out that there are exactly 187 maximal independent sets, and so the incidence matrix A for this example has dimensions 19×187. □

An *n-multicoloring* of a hypergraph $H = (V, E)$ is an assignment of a set of distinct elements ("colors") from $\{1, 2, \cdots, n\}$ to each vertex in such a way that for all colors $c = 1, 2, \cdots, n$, the set of vertices assigned color c must be an independent set. In principle, an n-multicoloring can be described by an $N \times n$ matrix (m_{ic}) of 0's and 1's such that $m_{ic} = 1$ if color c is assigned to vertex i and 0 otherwise. Given an n-multicoloring, if the number of colors assigned to v_i is m_i, then plainly,

$$m_i = \sum_{c=1}^{n} m_{ic}, \qquad i = 1, 2, \cdots, N. \tag{2.4}$$

Theorem 2.1: Suppose (m_{ic}) is an n-multicoloring of the hypergraph H. If $(w_1, w_2, \cdots, w_N)$ is any set of nonnegative weights attached to the vertices of H, we have

$$\sum_{i=1}^{N} w_i m_i \le n W_{\max} \tag{2.5}$$

where $W_{\max}$ is the maximum component of the $v-m$ transform of $(w_1, w_2, \cdots, w_N)$.

Proof: From (2.4), we have

$$\sum_{i=1}^{N} w_i m_i = \sum_{i=1}^{N} w_i \sum_{c=1}^{n} m_{ic} = \sum_{c=1}^{n}\sum_{i=1}^{N} w_i m_{ic}. \tag{2.6}$$

For a fixed value of c, the inner sum in (2.6) is equal to $\sum_{i\in J_c} w_i$, where $J_c = \{i: m_{ic} = 1\}$. But by the definition of a multicoloring, the set of vertices assigned a fixed color must be an independent set, so that $J_c \subseteq I_j$ for some index j. Hence (recall that the weights w_i are nonnegative),

$$\sum_{i=1}^{N} w_i m_{ic} = \sum_{i\in J_c} w_i \le \sum_{i\in I_j} w_i = W_j \le W_{\max}. \tag{2.7}$$

Thus, combining (2.5) and (2.6), we have

$$\sum_{i=1}^{N} w_i m_i \le \sum_{c=1}^{n} W_{\max} = nW_{\max},$$

as asserted. □

We next define a *random n-multicoloring* of H as a random $N \times n$ matrix $M = (m_{ic})$ of 0's and 1's such that for each point ω in the underlying sample space, $M(\omega)$ is an n-multicoloring of H.

Theorem 2.2: If M is a random n-multicoloring for H, then for any set of numbers $(y_1, y_2, \cdots, y_N)$ satisfying $0 \le y_i \le 1$, we have

$$E\left(\sum_{i=1}^{N} m_i\right) \le \sum_{i=1}^{N} E(m_i) y_i + n \max_j (N_j - Y_j)$$

where N_j is the size of the jth maximal independent set V_j, and $(Y_1, Y_2, \cdots, Y_M)$ is the $v-m$ transform of $(y_1, y_2, \cdots, y_N)$.

Proof: For any ω, we have

$$\sum_{i=1}^{N} m_i(\omega) = \sum_{i=1}^{N} m_i(\omega) y_i + \sum_{i=1}^{N} m_i(\omega)(1 - y_i).$$

The theorem now follows by applying Theorem 2.1 to the second sum on the right side, and then taking expectations of both sides. □

III. Upper Bounds on the Performance of Channel Assignment Algorithms

Let us review the system model we introduced in Section I, using the terminology developed in Section II. We are given a hypergraph $H = (V, E)$, with vertex set $V = \{v_1, v_2, \cdots, v_N\}$ and edge set $E = \{E_1, E_2, \cdots, E_K\}$, a probability vector $p = (p_1, p_2, \cdots, p_N)$, and a positive integer n. The vertices of H represent the cells of our system, the edges of H represent the minimal forbidden reuse sets, the components of p represent the relative distribution of the offered traffic, and n represents the number of available channels.

We assume that calls arrive randomly, and that the normalized traffic intensity is r Erlangs per available channel, so that the expected number of offered calls in the system is rn. The traffic pattern $p = (p_1, p_2, \cdots, p_N)$ says that the traffic intensity in the ith cell is $p_i n$, i.e., the offered traffic in the ith cell is $p_i rn$. The traffic is assumed to be independent from cell to cell.

We wish to analyze the performance of a given channel assignment algorithm, which takes each call request in each cell and either assigns it to a channel or blocks it. We shall make no formal attempt to define a channel-assignment algorithm, except to assume that at any point in time the set of cells using a given channel must be a subset of a maximal independent set of H, i.e., that any such algorithm produces a random n-multicoloring of H. If we denote the number of channels being used in cell i by μ_i, then the carried traffic, which is the expected number of channels in use at a given time, is $E(\sum_{i=1}^{N} \mu_i)$. As mentioned in Section I, we measure the performance of a given channel assignment algorithm by its carried-traffic function $T(r)$, defined as the expected number of accepted calls per available channel:

$$T(r) = \frac{1}{n} E(\mu_1 + \cdots + \mu_N). \tag{3.1}$$

The main result of this section is the following.

Theorem 3.1: Let $(y_1, y_2, \cdots, y_N)$ be any list of N numbers satisfying $0 \le y_i \le 1$ for $i = 1, 2, \cdots, N$. Then, for any channel assignment algorithm,

$$T(r) \le r \sum_{i=1}^{N} p_i y_i + \max_j (N_j - Y_j) \tag{3.2}$$

where $(Y_1, Y_2, \cdots, Y_M)$ is the $v-m$ transform of $(y_1, y_2, \cdots, y_N)$.

Proof: Since the average carried traffic cannot exceed the average offered traffic, and since the average offered traffic in the ith cell is $p_i rn$, then $E(\mu_i) \le p_i rn$ for $i = 1, 2, \cdots, N$. The result now follows from (3.1) and Theorem 2.2. □

The following result is a simple corollary to Theorem 3.1, but it allows us to define the important function $T_0(r)$, which is an upper bound on the carried traffic function for any channel assignment algorithm for the (H, p) system.

Theorem 3.2: Suppose $T_0(r)$ is the value of the following linear program:

$$r \sum_{i=1}^{N} p_i y_i + y_{N+1} = \text{minimum, subject to} \tag{3.3}$$

$$0 \le y_i \le 1 \qquad i = 1, 2, \cdots, N \tag{3.4}$$

$$\sum_{i=1}^{N} y_i a_{ij} + y_{N+1} \ge N_j \qquad j = 1, 2, \cdots, M. \tag{3.5}$$

Then, for any channel-assignment algorithm for the (H, p) system, $T(r) \le T_0(r)$.

Proof: If (3.4) is satisfied, then by Theorem 3.1, the bound (3.2) holds. If now y_{N+1} is a real number satisfying (3.5), then by the definition (2.1) of the $v-m$ transform,

$$y_{N+1} \ge N_j - Y_j \qquad j = 1, 2, \cdots, M. \tag{3.6}$$

Thus, from (3.2), it follows that

$$T(r) \le r\sum_{i=1}^{N} p_i y_i + y_{N+1} \tag{3.7}$$

for any set of numbers $y_1, y_2, \cdots, y_{N+1}$ satisfying (3.4) and (3.5). This completes the proof. □

Example 3.1: We can illustrate Theorem 3.1 with the (H, p) system of Fig. 1(a) for which $(N_1, \cdots, N_{10}) = (2, 2, 2, 2, 2, 2, 2, 2, 2, 1)$. If we take $y = (1, 1, 1, 1, 1, 1, 1)$, then $Y = (2, 2, 2, 2, 2, 2, 2, 2, 2, 1)$ and $\max_j(N_j - Y_j) = 0$. Thus, Theorem 3.1 implies $T(r) \le r$ for all $r \ge 0$. If $y = (0, 0, 0, 0, 0, 0, 0)$, then $Y = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$ and $\max_j(N_j - Y_j) = 2$, so that $T(r) \le 2$ for all $r \ge 0$. Finally, if $y = (1/2, 1/2, 1/2, 1/2, 1/2, 1/2, 0)$, then $Y = (1, 1, 1, 1, 1, 1, 1, 1, 1, 0)$ and $\max_j(N_j - Y_j) = 1$, so that $T(r) \le 3r/8 + 1$ for all $r \le 0$. Combining these three inequalities, we find that the carried traffic function $T(r)$ for the (H, p) system in Fig. 1(a) must satisfy $T(r) \le \min(r, 3r/8 + 1, 2)$. Indeed it is possible to show that $T_0(r) = \min(r, 3r/8 + 1, 2)$ for this system. [See Fig. 1(b).] □

Example 3.2: Consider the 19-cell example of Fig. 2(a). If we take $y = (1, 1, \cdots, 1)$, $\max(N_j - Y_j) = 0$, and Theorem 3.1 implies $T(r) \le r$ for all $r \ge 0$. If we take $y = (0, 0, \cdots, 0)$, $\max(N_j - y_j) = 7$, and we get $T(r) \le 7$ for all $r \ge 0$. Therefore, $T(r) \le \min(r, 7)$ for all $r \ge 0$. Solving the linear program of Theorem 3.2 using a computer, we get

$$T_0(r) = \min\left(r, \frac{12}{19}r + \frac{13}{7}, \frac{8}{19}r + 3, \frac{6}{19}r + \frac{18}{5}, \frac{5}{19}r + 4, \frac{1}{19}r + 6, 7\right).$$

This function is shown in Fig. 2(b). □

IV. Asymptotic Performance of Fixed Channel Assignment Algorithms

In this section, we will study the asymptotic performance of a class of channel assignment algorithms which we call *fixed channel assignment algorithms*. By asymptotic, we mean that n, the number of available channels, is large. We shall not be precise about the underlying model of the offered traffic, except to require that it satisfy the following "asymptotic traffic property (ATP)," originally introduced by McEliece and Sivarajan [13], which can be defined by the performance of a simple one-cell channel assignment algorithm.

Suppose, then, that there is just one cell, and that there are n available channels. An obvious channel assignment algorithm in this situation is a "greedy" algorithm, i.e., one in which when a new call request arrives, it is assigned to any unoccupied channel, if there is one, and otherwise it is blocked. If the intensity of the offered traffic is k Erlangs, we denote by $C(k, n)$ the carried traffic, i.e., the expected number of occupied channels for the greedy algorithm. The ATP referred to above is

$$\lim_{n\to\infty} \frac{C(k_n, n)}{n} = \min(r, 1) \qquad \text{if } k_n/n \to r. \tag{4.1}$$

The ATP says that if the offered traffic is less than the number of available channels, then, asymptotically, the fraction of offered calls that are blocked approaches zero, whereas if the offered traffic exceeds the number of available channels, then, asymptotically, the fraction of the available channels that are occupied approaches one. It is thus a kind of law or large numbers. Most common traffic models satisfy the ATP, including the standard Poisson arrivals with exponential call durations. (We give a proof of this in Section VI.)

We will now define the family of channel assignment algorithms, which we call *fixed channel assignment algorithms*, and proceed to analyze their asymptotic performance, assuming the ATP.

Thus, let $X = (X_1, X_2, \cdots, X_M)$ be a list of M real numbers satisfying $X_j \ge 0$ and $\sum_j X_j = 1$. For $j = 1, 2, \cdots, M$, we define $n_j = \lfloor nX_j \rfloor$, and create M disjoint classes of channels $C_1, C_2, \cdots, C_M$, with class C_j containing exactly n_j channels. Then, for $j = 1, 2, \cdots, M$, we allocate each of the channels in C_j to each vertex in the jth maximal independent set V_j. Thus, $m_i = \sum_j n_j a_{ij}$ channels are allocated to vertex v_i. But since $nX_j - 1 < n_j \le nX_j$, and since $\sum_j X_j a_{ij} = x_i$, where $(x_1, x_2, \cdots, x_N)$ is the $m-v$ transform of X, it follows that $nx_i - M_i < m_i \le nx_i$, and so

$$\lim_{n\to\infty} \frac{m_i}{n} = x_i \qquad \text{for } i = 1, 2, \cdots, N. \tag{4.2}$$

For a given X and n, we define a channel assignment algorithm as follows. When a call request arrives at v_i, assign it one of the m_i channels available at v_i, if at least one is not in use; otherwise, block the call. We call this the *X fixed channel assignment algorithm*. It is, in effect, N independent greedy algorithms, one for each cell in the system.

At any given time, the channels in use in each of the cells constitute an n-multicoloring of the hypergraph H. Moreover, these n-multicolorings have the property that the channels assigned to call requests in cell i in each of these n-multicolorings is a subset of the m_i channels allocated to cell i.

Theorem 4.1: Assume the ATP. If $T_X(r)$ denotes the carried traffic function of the X fixed channel assignment algorithm for the (H, p) system, then

$$\lim_{n\to\infty} T_X(r) = \sum_{i=1}^{N} \min(rp_i, x_i) \qquad \text{for all } r \ge 0. \tag{4.3}$$

Proof: In the X fixed channel assignment algorithm, each of the N cells operates independently of the others. Within the ith cell, the algorithm is a greedy algorithm with m_i available channels and the offered traffic is $p_i rn$ Erlangs. Thus, the carried traffic in the ith cell is $E(\mu_i) = C(p_i rn, m_i)$, and the carried traffic for the entire system is $\sum_{i=1}^{N} C(p_i rn, m_i)$, so that

$$T_X(r) = \frac{1}{n}\sum_{i=1}^{N} C(p_i rn, m_i). \tag{4.4}$$

But from the ATP property (4.1) and the known rate of growth of m_i (4.2),

$$\lim_{n\to\infty} \frac{1}{n} C(p_i rn, m_i) = \min(p_i r, x_i). \tag{4.5}$$

Combining (4.4) and (4.5), we obtain (4.3). □

Theorem 4.2: Suppose $T_1(r)$ is the value of the following linear program:

$$\sum_{j=1}^{M} N_j X_j - \sum_{i=1}^{N} z_i = \text{maximum, subject to} \tag{4.6}$$

$$X_j \geq 0 \qquad j = 1, 2, \cdots, M \tag{4.7}$$

$$z_i \geq 0 \qquad i = 1, 2, \cdots, N \tag{4.8}$$

$$\sum_{j=1}^{M} X_j = 1 \tag{4.9}$$

$$\sum_{j=1}^{M} X_j a_{ij} - z_i \leq p_i r \qquad i = 1, 2, \cdots, N. \tag{4.10}$$

Then, for any fixed r, there exists a fixed channel assignment algorithm for the (H, p) system whose asymptotic carried traffic function is arbitrarily close to $T_1(r)$.

Proof: We begin with vectors $X = (X_1, X_2, \cdots, X_M)$ and $z = (z_1, z_2, \cdots, z_N)$ whose components satisfy (4.7)–(4.10). Let $(x_1, x_2, \cdots, x_n)$ be the $m - v$ transform of X. We note that (4.10) is equivalent to $x_i - z_i \leq p_i r$, and so by (4.8) and (4.10), we have $x_i - z_i \leq \min(p_i r, x_i)$. Therefore,

$$\sum_{i=1}^{N} (x_i - z_i) \leq \sum_{i=1}^{N} \min(p_i r, x_i). \tag{4.11}$$

Note also

$$\begin{aligned}\sum_{i=1}^{N} x_i &= \sum_{i=1}^{N}\sum_{j=1}^{M} X_j a_{ij} \\ &= \sum_{j=1}^{M} X_j \sum_{i=1}^{N} a_{ij} \\ &= \sum_{j=1}^{M} X_j N_j \end{aligned} \tag{4.12}$$

so that

$$\sum_{i=1}^{N} (x_i - z_i) = \sum_{j=1}^{M} X_j N_j - \sum_{i=1}^{N} z_i. \tag{4.13}$$

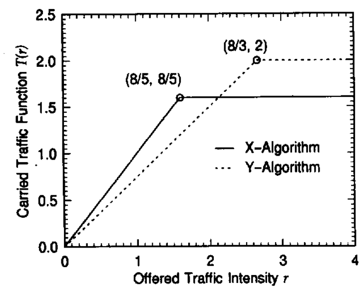

Fig. 3. The functions $T_X(r) = \min(r, \frac{8}{5})$ and $T_Y(r) = \min(\frac{3}{4}r, 2)$ for the 7-cell (H, p) system of Fig. 1(a). Here, $X = (0, \frac{1}{5}, 0, 0, \frac{1}{5}, 0, 0, \frac{1}{5}, 0, \frac{2}{5})$ and $Y = (0, \frac{1}{3}, 0, 0, \frac{1}{3}, 0, 0, \frac{1}{3}, 0, 0)$.

Therefore, we have

$$\begin{aligned}\lim_{n\to\infty} T_X(r) &= \sum_{i=1}^{N} \min(p_i r, x_i) \quad \text{by Theorem 4.1} \\ &\geq \sum_{i=1}^{N} (x_i - z_i) \quad \text{by (4.11)} \\ &= \sum_{j=1}^{S} N_j X_j - \sum_{i=1}^{N} z_i \quad \text{by (4.13)}\end{aligned} \tag{4.14}$$

Thus, the X fixed channel assignment algorithm has a carried traffic function which is asymptotically at least as large as the objective function (4.6), and this proves the theorem. □

Example 4.1: To illustrate Theorem 4.1, we return to the hypergraph of Fig. 1(a). If we let $X_2 = X_5 = X_8 = 1/5$, $X_{10} = 2/5$, and $X_j = 0$ for all other values of j, then the $m-v$ transform of X is $(1/5, 1/5, 1/5, 1/5, 1/5, 1/5, 2/5)$, and Theorem 4.1 implies that the X algorithm's asymptotic carried traffic curve is $T_X(r) = 6\min(\frac{1}{8}r, \frac{1}{5}) + \min(\frac{1}{4}r, 2/5) = \min(\frac{3}{4}r, \frac{6}{5}) + \min(\frac{1}{4}r, 2/5) = \min(r, \frac{8}{5})$. Similarly, if $Y_2 = Y_5 = Y_8 = 1/3$, and $Y_j = 0$ for all other values of j, then the $m - v$ transform of Y is $(1/3, 1/3, 1/3, 1/3, 1/3, 1/3, 0)$, and by Theorem 4.1, the Y algorithm's asymptotic carried traffic curve is $T_Y(r) = 6\min(\frac{1}{8}r, \frac{1}{3}) = \min(\frac{3}{4}r, 2)$. These two functions are shown in Fig. 3. By taking all possible convex combinations of X and Y, i.e., vectors of the form $Z = \lambda X + (1 - \lambda)Y$, we obtain a family of curves which give the convex hull of $T_X(r)$ and $T_Y(r)$. But this convex hull is the same as the curve $T_0(r)$ given in Fig. 1(b). Since we saw in Section III that no point above this curve is achievable by any channel assignment algorithm, and we have shown that every point below the curve is asymptotically achievable, we are justified in asserting that the function $T_0(r)$ is *the* achievable region for the performance of channel assignment algorithms for the (H, p) system of Fig. 1(a). In the next section, we will see that this is no accident, but an instance of a general rule.

Example 4.2: For another illustration of Theorem 4.1, we consider the 19-cell hypergraph of Fig. 2(a). For this hypergraph, there exist vectors $X^{(k)}$, $k = 1, \cdots, 6$ (of length $M = 187$) whose $m - v$ transforms are the respective vectors

$x^{(k)}$ (of length $N = 19$) listed below. The vectors $x^{(k)}$ satisfy

$$x_1^{(k)} = x_3^{(k)} = x_8^{(k)} = x_{12}^{(k)} = x_{17}^{(k)} = x_{19}^{(k)}$$

$$x_2^{(k)} = x_4^{(k)} = x_7^{(k)} = x_{13}^{(k)} = x_{16}^{(k)} = x_{18}^{(k)}$$

and

$$x_5^{(k)} = x_6^{(k)} = x_9^{(k)} = x_{11}^{(k)} = x_{14}^{(x)} = x_{15}^{(k)}$$

and hence are completely specified by

$$x_1^{(1)} = x_2^{(1)} = x_5^{(1)} = x_{10}^{(1)} = \frac{13}{49}$$

$$x_1^{(2)} = x_2^{(2)} = x_5^{(2)} = \frac{2}{7}, \quad x_{10}^{(2)} = \frac{1}{7}$$

$$x_1^{(3)} = x_2^{(3)} = x_5^{(3)} = \frac{3}{10}, \quad x_{10}^{(3)} = 0$$

$$x_1^{(4)} = x_2^{(4)} = \frac{2}{5}, \quad x_5^{(4)} = \frac{1}{5}, \quad x_{10}^{(4)} = 0$$

$$x_1^{(5)} = x_2^{(5)} = \frac{1}{2}, \quad x_5^{(5)} = 0, \quad x_{10}^{(5)} = \frac{1}{2}$$

and

$$x_1^{(6)} = 1, \quad x_2^{(6)} = x_5^{(6)} = 0, \quad x_{10}^{(6)} = 1.$$

For example, there exists a vector $X^{(1)}$ whose $m-v$ transform is the vector $x^{(1)}$, and hence, a fixed CAA, viz. the $X^{(1)}$-fixed CAA, that allocates a fraction 13/49 of all the available channels to each of the 19 cells. It turns out again that the convex hull of the carried traffic functions $T_{X^{(k)}}(r)$ of the $X^{(k)}$ algorithms corresponding to these vectors $X^{(k)}$ is the same as the curve $T_0(r)$ given in Fig. 2(b). Therefore, as in the previous example, the function $T_0(r)$ is the achievable region for the performance of channel assignment algorithms for the (H, p) system of Fig. 2(a). □

V. Equality of $T_0(r)$ and $T_1(r)$: General Properties of This Function

In Section III, we showed that for any channel assignment algorithm for the (H, p) system, the corresponding carried traffic function was bounded above by $T_0(r)$. On the other hand, in Section IV, we showed that if n is sufficiently large, then the performance of certain fixed channel assignment algorithms for the (H, p) system is bounded *below* by $T_1(r)$. Interestingly, however, these two functions are equal.

Theorem 5.1: $T_0(r) = T_1(r)$, *for all* $r \geq 0$.

Proof: By Theorem 3.2, $T_0(r)$ is the value of a certain linear program, and by Theorem 4.2, $T_1(r)$ is the value of another linear program. However, these programs are dual programs (see Franklin [4, Sect. 1.2]), and so by the Duality Theorem of Linear Programming [4, Sect. 1.8], the values of these two programs are equal, provided both programs are feasible. It is easy to show that both programs are feasible: a feasible solution for the $T_0(r)$ program is $y_1 = y_2 = \cdots = y_N = 0$ and $y_{N+1} = \max_j N_j$, and a feasible solution for the $T_1(r)$ program is $X_1 = 1$, $X_2 = \cdots = X_M = 0$ and $z_i = a_{i1}$ for $i = 1, 2, \cdots, N$. Thus, $T_0(r) = T_1(r)$, as asserted. □

Let us denote the common value of the functions $T_0(r)$ and $T_1(r)$ by $T_{H,p}(r)$. The next theorem gives the most important general properties of this function.

Theorem 5.2: The function $T_{H,p}(r)$ *has the following properties.*

a) $T_{H,p}(r)$ *is nondecreasing, continuous, piecewise linear, and convex* ∩.

b) $T_{H,p}(r) = r$ *for all* $r \leq r_0$, *where* r_0 *is the value of the following linear program*:

$$r = \text{maximum, subject to} \tag{5.1}$$

$$X_j \geq 0 \qquad j = 1, 2, \cdots, M \tag{5.2}$$

$$\sum_{j=1}^{M} X_j = 1 \tag{5.3}$$

$$\sum_{j=1}^{M} X_j a_{ij} \geq p_i r \qquad i = 1, 2, \cdots, N. \tag{5.4}$$

Furthermore, $r_0 \geq 1$.

c) If $p_i > 0$ *for all* i, *then* $T_{H,p}(r) = \max_j N_j$ *for all* $r \geq r_1$, *where* r_1 *is the value of the following linear program (here, and henceforth* $V_1, V_2, \cdots, V_{M^*}$ *are the maximal independent sets of largest cardinality)*:

$$r = \text{minimum, subject to} \tag{5.5}$$

$$X_j \geq 0 \qquad j = 1, 2, \cdots, M^* \tag{5.6}$$

$$\sum_{j=1}^{M^*} X_j = 1 \tag{5.7}$$

$$\sum_{j=1}^{M^*} X_j a_{ij} \leq p_i r \qquad i = 1, 2, \cdots, N. \tag{5.8}$$

If some of the p_i*'s are zero, then the above statement must be modified by replacing the original system* (H, p) *with the reduced system* (H', p') *consisting of only those cells with nonzero* p_i.

Proof: a) According to Theorem 4.2, a feasible solution for the $T_1(r)$ program will also be feasible for all $r' \geq r$, and so $T_1(r') \geq T_1(r)$ for all $r' \geq r$. Thus, $T_{H,p}(r)$ is nondecreasing. Furthermore, the $T_1(r)$ program is a parametric linear program (with parameter r) in the sense of [4, Sect. 1.9], and so by a result proved there (p. 70), the function $T_1(r)$ is continuous, convex $\cap$, and piecewise linear.

b) We use the $T_1(r)$ program (Theorem 4.2) as our definition of $T_{H,p}(r)$. Since $N_j = \sum_i a_{ij}$, the objective function (4.6) can be written as

$$\sum_{i=1}^{N}\left(\sum_{j=1}^{M} X_j a_{ij} - z_i\right). \tag{5.9}$$

By the constraint (4.10), it follows that $T_1(r) \leq r$, with equality if and only if $\sum_j X_j a_{ij} - z_i = p_i r$ for $i = 1, 2, \cdots, N$. Thus, since $z_i \geq 0$, we have that $T_1(r) = r$ if and only if there exists a vector $(X_1, \cdots, X_M)$ satisfying (5.2)–(5.4).

To prove that $r_0 \geq 1$, we consider the following choice for the X_j's:

$$X_j = \sum_{i=1}^{N} \frac{p_i a_{ij}}{M_i} \qquad \text{for } j = 1, \cdots, M \tag{5.10}$$

where, in (5.10), we recall the notation introduced in Section II, $M_i = \sum_j a_{ij}$. Plainly, $X_j \geq 0$ for all j, and a routine calculation shows that $\sum_j X_j = 1$. We now consider the sum $\sum_j X_j a_{ij}$:

$$\begin{aligned}\sum_j X_j a_{ij} &= \sum_i a_{ij} \sum_k \frac{p_k a_{kj}}{M_k} \\ &= \sum_k \frac{p_k}{M_K} \sum_j a_{ij} a_{kj} \\ &\geq \frac{p_i}{M_i} \sum_j a_{ij}^2 \\ &= \frac{p_i}{M_i} \sum_j a_{ij} \\ &= \frac{p_i}{M_i} \cdot M_i \\ &= p_i.\end{aligned}$$

Thus, the X_j's defined in (5.10) satisfy the constraints (5.2)–(5.4) with $r = 1$, which shows that $r_0 \geq 1$.

c) We again use the $T_1(r)$ program definition of $T_{H,p}(r)$, and define $N_{\max} = \max_j N_j$. Because of the constraints (4.7)–(4.9), the objective function (4.6) is at most $N_{\max}$, with equality if and only if the z_i's are all zero and $X_j = 0$ for $j > M^*$. Thus, $T_1(r) = N_{\max}$ if and only if there exists a vector $(X_1, \cdots, X_M)$ satisfying (5.6)–(5.8). To show that the program (5.5)–(5.8) is feasible, we consider the choice $X_1 = 1$, $X_j = 0$ for $j > 1$. Since (5.6) and (5.7) are satisfied, this vector will be feasible iff (5.8) is satisfied, i.e., if $a_{i1} \leq p_i r$ for $i = 1, 2, \cdots, N$. If $p_i > 0$ for all i, this will hold for all sufficiently large r.

If, on the other hand, some of the p_i's are zero, the program (5.5)–(5.8) may no longer be feasible. However, if we remove from the system all cells v_i in which there is no traffic, i.e., for which $p_i = 0$, then plainly we will not affect the function $T(r)$, and then the above reasoning may safely be applied to the reduced system. □

For a given hypergraph H, Theorem 5.2 gives us the following upper and lower bounds on $T_{H,p}(r)$, which do not depend on the traffic pattern p:

$$\min(r, 1) \leq T_{H,p}(r) \leq \min(r, N_{\max}). \tag{5.11}$$

If, for a particular p, we have $T_{H,p}(r) = \min(r, 1)$, we say that p is an *unfavorable traffic pattern* for H. On the other hand, if $T_{H,p}(r) = \min(r, N_{\max})$, we say that p is a *favorable traffic pattern*. The next theorem identifies these extreme traffic patterns.

Theorem 5.3: Let $N'_{\max}$ denote the size of the largest maximal independent set of H'. For a fixed hypergraph H, the traffic pattern p is unfavorable if and only if $N'_{\max} = 1$. On the other hand, p is favorable if and only if the vector $N_{\max}p$ is a convex combination of the first M^* columns of the incidence matrix A, i.e., if there exists a vector $(X_1, \cdots, X_{M^*})$ satisfying (5.6) and (5.7) such that

$$\sum_{j=1}^{M} X_j a_{ij} = N_{\max} p_i \qquad \text{for } i = 1, 2, \cdots, N.$$

Proof: Suppose that p is unfavorable for H. By Theorem 5.2c), this implies $N'_{\max} = 1$. On the other hand, if $N'_{\max} = 1$, then combining the result $T_{H,p}(r) = r$ for $r \leq 1$ from Theorem 5.2b) with $T_{H,p}(r) = N'_{\max} = 1$ for $r \geq r_1$ from part c) and $T_{H,p}(r)$ is nondecreasing from part a), we obtain that $T_{H,p}(r) = \min(r, 1)$, i.e., p is unfavorable.

We turn now to favorable traffic patterns. By Theorem 5.2, p is favorable if and only if $r_0 = r_1$. If this holds, then $r_0 = T_{H,p}(r_0) = T_{H,p}(r_1) = N_{\max}$, and so p is favorable if and only if $T_1(N_{\max}) = N_{\max}$. We saw in the proof of Theorem 5.2c), however, that $T_1(r) = N_{\max}$ if and only if $z_i = 0$ and $X_j = 0$ for $j > M^*$ in the $T_1(r)$ program. Then, by (4.10), with $r = N_{\max}$, the components x_i of the $m - v$ transform of X satisfy

$$x_i = \sum_{j=1}^{M^*} X_j a_{ij} \leq p_i N_{\max} \qquad i = 1, \cdots, N.$$

But also,

$$\sum_{i=1}^{N} x_i = \sum_{j=1}^{M^*} X_j \sum_{i=1}^{N} a_{ij} = \sum_{j=1}^{M^*} X_j N_j = N_{\max}.$$

On the other hand, $\sum_i N_{\max} p_i = N_{\max}$, so if $x_i < N_{\max} p_i$ for any value of i, we would have $\sum_i x_i < N_{\max}$, a contradiction. □

To illustrate Theorem 5.3, we return to the hypergraph described in Fig. 1(a). Here, $N_{\max} = 2$, and so a traffic pattern p is favorable if and only if $2p$ is a convex combination of the first nine columns of the incidence matrix A given in (2.3) (see Fig. 4). For this hypergraph, $N'_{\max} = 1$ if and only if the reduced system (H', p') consists of one of the following sets of three mutually adjacent cells: $\{1, 2, 7\}$, $\{2, 3, 7\}$, $\{3, 4, 7\}$, $\{4, 5, 7\}$, $\{5, 6, 7\}$, and $\{6, 1, 7\}$. Therefore, a

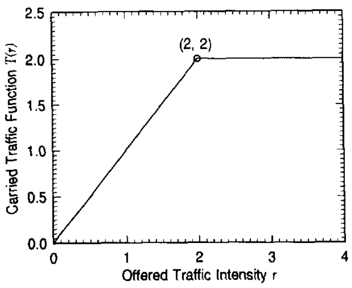

Fig. 4. The carried traffic function for the (H, p) system, where H is the hypergraph in Fig. 1(a), for a favorable traffic pattern p.

traffic pattern p is unfavorable if and only if p is concentrated in some one of these sets.

Theorem 5.2a) tells us that $T_{H,p}(r)$ is piecewise linear, but it does not say how many pieces there are. Nor do we know. However, if we define a *breakpoint* as a value of the offered traffic intensity $r > 0$ for which the slope of $T_{H,p}(r)$ changes its value, we offer the following conjecture.

Conjecture: For an N-cell system, there are at most N breakpoints.

For the 7-cell hypergraph of Fig. 1(a) there are two breakpoints, viz. $r = 8/5$ and $r = 8/3$. For the 19-cell hypergraph of Fig. 2(a), there are six breakpoints, viz. $r = 247/49, 38/7, 57/10, 38/5, 19/2$, and 19. Neither of these examples strains our conjecture very far. The following example, however, shows that the value N cannot be replaced by any smaller number.

Example 5.1: Consider an N-cell system with *no* forbidden subsets, i.e., the same channel may be used simultaneously in all N cells. For such a system, the transpose of the vertex-maximal independent set incidence matrix A is given by $A^T = (1, 1, \cdots, 1)$. From Theorem 4.2, for this system,

$$T_{H,p}(r) = \sum_{i=1}^{N} \min(p_i r, 1).$$

The breakpoints for this system are given by $r = 1/p_i$, $i = 1, \cdots, N$. If the vector p is such that no two of the p_i's are equal, there will be N distinct breakpoints. □

Computational Complexity: For given H, p, and r, $T_{H,p}(r)$ is the value of a linear program with $N + 1$ variables and $M + N$ constraints [(3.3)–(3.5)] or vice versa [(4.6)–(4.10)], where N is the number of vertices of H and M, the number of maximal independent sets. In general, the number of maximal independent sets M of a (hyper)graph can be exponential in the number of vertices N [16]. Therefore, the worst-case complexity of computing $T_{H,p}(r)$ is exponential in the number of vertices of H. This implies that, in general, $T_{H,p}(r)$ can be explicitly calculated only for systems of moderate size. However, large linear programs commonly occur in the solution of many problems of practical importance, and one of the practical uses of our results may be the evaluation of practical channel assignment algorithms whose performance can be compared against $T_{H,p}(r)$ for systems of moderate size.

VI. Performance of Actual Channel Assignment Algorithms

So far, we have not specified a definite probabilistic traffic model for call arrivals and holding times (durations). We have only required that such a model must satisfy the asymptotic traffic property (ATP). In this section, we focus on a specific model that satisfies the ATP. Namely, we assume that the process of call arrivals is Poisson with rate λ per second, i.e., the probability that there are n call arrivals in an interval of length T seconds is given by

$$e^{-\lambda T}\frac{(\lambda T)^n}{n!}.$$

Since we are assuming that call arrivals are independent from cell to cell, each cell will have a different λ associated with it. We assume that the mean duration of a call is $1/\mu$ seconds, but we will not need to assume any specific distribution for call durations.

We will first show that this model for offered traffic satisfies the ATP. Then, using this model, we will compare the performance of two specific channel assignment algorithms to each other and to the asymptotic performance limits of Theorem 5.2

Consider first a one-cell system. Let $\rho = \lambda/\mu$. Then, ρ is the offered traffic (in Erlangs). If n denotes the number of available channels, the probability that m calls are in progress is given by the well-known truncated Poisson distribution

$$p_m(\rho, n) = \frac{\rho^m/m!}{\sum_{k=0}^{n} \rho^k/k!}.$$

(See Syski [19, p. 147] or Bertsekas and Gallager [2, p. 140].) In particular, the probability that an incoming call is blocked is given by the celebrated *Erlang B formula*:

$$P_b(\rho, n) \stackrel{\text{def}}{=} p_n(\rho, n) = \frac{\rho^n/n!}{\sum_{k=1}^{n} \rho^k/k!}. \tag{6.1}$$

The following lemma, whose proof we leave to the reader, characterizes the asymptotic behavior of $P_b(\rho, n)$.

Lemma: If $f(r)$ denotes the limits, as $n \to \infty$, of the quantity $P_b(\rho, n)$, for a fixed $r = \rho/n$, then

$$f(r) = \begin{cases} 0 & 0 \le r \le 1 \\ 1 - 1/r & r > 1. \end{cases}$$

Theorem 6.1: For the traffic model with Poisson arrivals and arbitrary holding times, the ATP is satisfied, i.e.,

$$\lim_{n\to\infty} \frac{C(rn, n)}{n} = \min(r, 1) \qquad \text{for all } r \ge 0 \tag{6.2}$$

where $C(\rho, n)$ denotes the carried traffic for an offered traffic of ρ Erlangs and n channels.

Proof: From Syski [19, p. 147],

$$C(\rho, n) = \rho(1 - P_b(\rho, n)). \tag{6.3}$$

Using this and the above lemma,

$$\lim_{n\to\infty} \frac{C(rn, n)}{n} = \lim_{n\to\infty} r(1 - P_b(rn, n)) = \min(r, 1) \qquad \text{for all } r \ge 0$$

and the theorem is proved. □

Having now established that our Poisson traffic model indeed satisfies the ATP, we know that the limits described in Theorem 5.2 apply to any channel assignment algorithm for any (H, p) cellular system with Poisson traffic. We next introduce two specific channel assignment algorithms, whose performance we will then study in detail.

Dynamic Channel Assignment Algorithm (DCAA): Let the channels be numbered from 1 through n. A channel is said to be *available* in a cell if it can be assigned to a new call in that cell without violating any of the reuse constraints. The DCAA is a *greedy* algorithm which assigns the first (lowest numbered) available channel in a cell to a new call. Because of the difficulty of analyzing the performance of this algorithm, we have simulated its performance, and the results are presented below. For the simulations, we make the further assumption that the call durations are exponentially distributed with a mean of 3 min.

Fixed Channel Assignment Algorithm (FCAA): For a given value of r, let X be a solution to the linear program in Theorem 4.2 ($T_1(r)$ program) and let $x_i = \sum_j X_j a_{ij}$. From Theorem 4.2, a fixed channel assignment algorithm (FCAA) that assigns a fraction x_i of the total number of channels to cell i when the offered traffic intensity is r is asymptotically optimal. Given n, consider an FCAA that assigns $x_i n$ channels to cell i. For the rest of this section, when we speak of the FCAA, we will be referring to this algorithm. This algorithm is not, as it stands, a practical one since it needs to know the intensity of the offered traffic, which may be unknown and changing in time. An additional aspect that may render the algorithm impractical for large systems is the fact that it requires us to solve the linear program in Theorem 4.2, the complexity of which can be exponential in the number of cells in the system, as remarked at the end of Section V.

The carried traffic in cell i for the FCAA is given by (Syski [19, p. 147])

$$C(p_i rn, x_i n) = p_i rn(1 - P_b(p_i rn, x_i n))$$

where $P_b(\rho, n)$ is given by the Erlang B formula (6.1). An alternate expression for $P_b(\rho, n)$ is

$$P_b(\rho, n) = \frac{\rho^n e^{-n}}{\Gamma_\rho(n+1)}$$

where

$$\Gamma_\rho(n+1) = \int_\rho^\infty t^n e^{-t}\, dt$$

is the incomplete Gamma function (Syski [19, p. 497]). This is the formula we used for our computations.

We compute the channel assignments at the breakpoints (defined in Section V) using the linear program, and for values of r between breakpoints, we use the linear combination of the assignments at the breakpoints (see Section IV), which is asymptotically optimal. For $r \le r_0$, we use the solution for $r = r_0$, and for $r \ge r_1$, we use the solution for $r = r_1$. (The solutions at the breakpoints that we use for the 7-cell and the 19-cell examples are listed in Examples 4.1 and 4.2, respectively.)

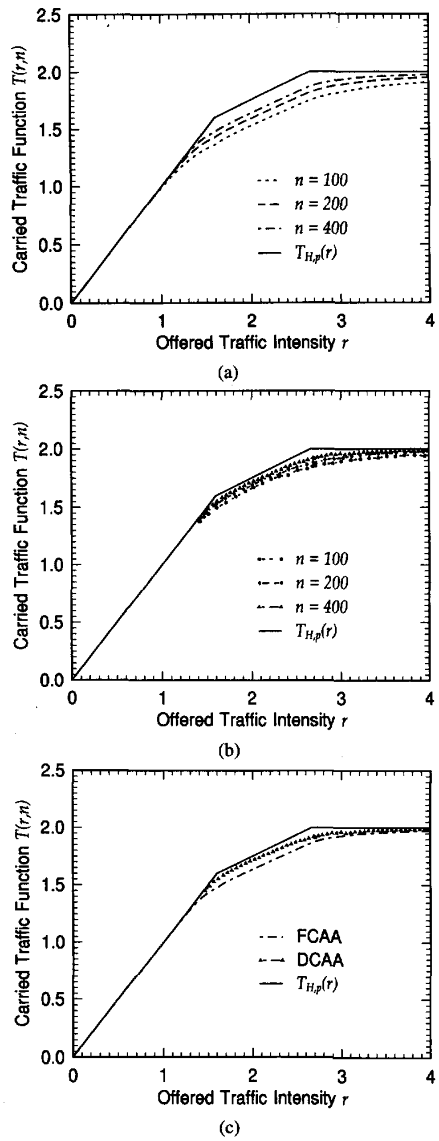

Fig. 5. (a) The performance of the FCAA for the 7-cell (H, p) system of Fig. 1(a). n is the number of channels available in the system. The function $T_{H,p}(r)$ of Fig. 1(b) is also shown for comparison. (b) The performance of the DCAA for the 7-cell (H, p) system of Fig. 1(a). n is the number of channels available in the system. The function $T_{H,p}(r)$ of Fig. 1(b) is also shown for comparison. (c) A comparison of the performance of the FCAA and the DCAA for the 7-cell (H, p) system of Fig. 1(a) when the number of channels $n = 400$. The function $T_{H,p}(r)$ of Fig. 1(b) is also shown for comparison.

Having described our two channel assignment algorithms, we now present, in Figs. 5(a)–(c) and 6(a)–(c), the results of our studies of their performances on the 7- and 19-cell systems. All figures show $T_{H,p}(r)$ for the purposes of comparison. In addition, for the 7-cell example,

- Fig. 5(a) shows the performance of the FCAA for $n =$ 100, 200, and 400,
- Fig. 5(b) shows the performance of the DCAA for $n =$ 100, 200, 400, and

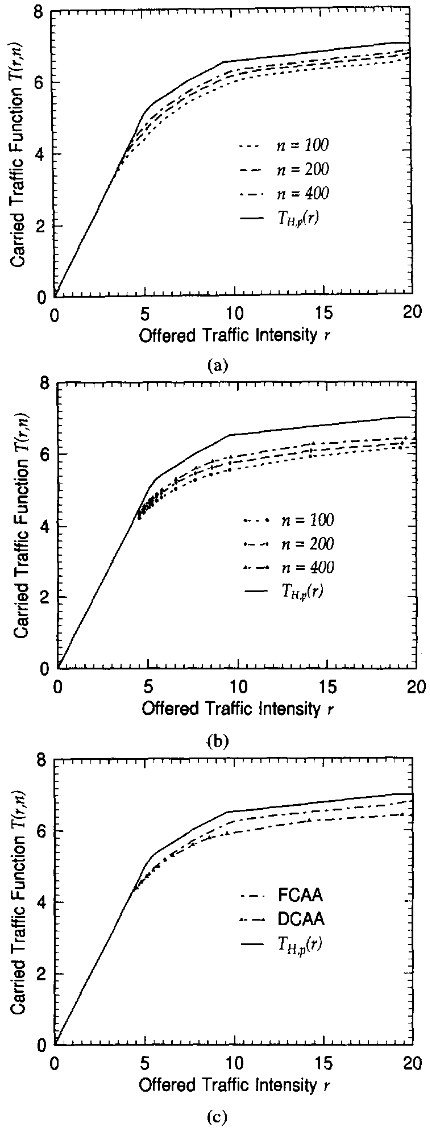

Fig. 6. (a) The performance of the FCAA for the 19-cell (H, p) system of Fig. 2(a) n is the number of channels available in the system. The function $T_{H,p}(r)$ of Fig. 2(b) is also shown for comparison. (b) The performance of the DCAA for the 19-cell (H, p) system of Fig. 2(a) n is the number of channels available in the system. The function $T_{H,p}(r)$ of Fig. 2(b) is also shown for comparison. (c) A comparison of the performance of the FCAA and the DCAA for the 19-cell (H, p) system of Fig. 2(a) when the number of channels $n = 400$. The function $T_{H,p}(r)$ of Fig. 2(b) is also shown for comparison.

- Fig. 5(c) compares the performance of the FCAA and the DCAA for $n = 400$.

Similarly, for the 19-cell example,

- Fig. 6(a) shows the performance of the FCAA for $n =$ 100, 200, and 400.
- Fig. 6(b) shows the performance of the DCAA for $n =$ 100, 200, 400, and
- Fig. 6(c) compares the performance of the FCAA and the DCAA for $n = 400$.

We see that the performance of the DCAA is better than that of the FCAA, and quite close to the asymptotic limit in the 7-cell example for $n = 400$. Thus, the DCAA cannot be improved upon by much, if at all, in this example, and based on these performance results, we conjecture that it, like the FCAA, is asymptotically optimal. This is surprising because the DCAA is quite unsophisticated (and greedy), and its implementation does not require an estimate of the offered traffic, unlike the FCAA.

The performance of the DCAA is quite good in the case of the 19-cell example, too, although the "impractical" FCAA does do better.

We find these numerical results, especially those of the DCAA, encouraging. They suggest that the asymptotic limits of Theorem 5.2 are closely approached for relatively small values of n with relative unsophisticated channel assignment algorithms. Perhaps future researchers will be able to verify this more convincingly. In any case, we have been able to estimate the rate of convergence of the performance of the "impractical" FCAA to the limit $T_{H,p}(r)$ quite closely. We present these results in the last part of this section.

The carried traffic function of the FCAA is given by

$$T(r, n) = \frac{C(rn, n)}{n}.$$

[This is the same function that is defined by (3.1), but now we explicitly exhibit the dependence of the carried traffic function of a CAA on n by writing $T(r, n)$ rather than $T(r)$.]

From Theorem 4.2, for all $r \geq 0$,

$$\lim_{n\to\infty} T(r, n) = T_{H,p}(r).$$

We wish to know how "far away" $T(r, n)$ is from $T_{H,p}(r)$ for some given, finite n. A natural notion of the distance between the two functions of r is

$$d(n) \stackrel{\text{def}}{=} \sup_{r\geq 0} |T_{H,p}(r) - T(r, n)| = \sup_{r\geq 0} T_{H,p}(r) - T(r, n). \tag{6.4}$$

The last step holds since $T_{H,p}(r) \geq T(r, n)$ (by Theorem 3.2). Clearly,

$$\lim_{n\to\infty} d(n) = 0.$$

We will use this function $d(n)$ to measure the distance between $T(r, n)$ and $T_{H,p}(r)$. We begin with a simple single-cell system.

For a single-cell system, $T_{H,p}(r) = \min(r, 1)$, and $T(r, n) = C(rn, n)/n$ is easily expressed in terms of the Erlang B formula using (6.3). Fig. 7 compares $T(r, 100), T(r, 200), T(r, 400)$, and $T_{H,p}(r)$ for $r \in [0, 2]$. Since $T(r, n)$ is a monotonically nondecreasing function of r (from the Erlang B formula) and $T_{H,p}(r)$ is constant for $r \geq 1$, it is clear that the supremum in (6.4) is achieved for some $r \leq 1$, and we can write max instead of sup in (6.4). Therefore,

$$d(n) = \max_{0\leq r\leq 1} T_{H,p}(r) - T(r, n) = \max_{0\leq r\leq 1} r - T(r, n).$$

It can be shown from the Erlang B formula that $r - T(r, n)$ is a monotonically nondecreasing function of r. Therefore, the

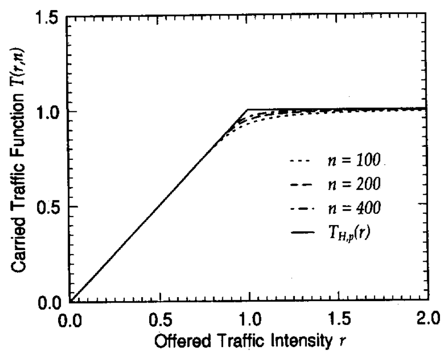

Fig. 7. The performance of the FCAA (and the DCAA) for a single-cell system. n is the number of channels available in the cell. The function $T_{H,p}(r)$ is also shown for comparison.

maximum is achieved for $r = 1$, and if $\delta(n)$ is the value of $d(n)$ for a single-cell system,

$$\delta(n) = 1 - T(1, n) = P_b(n, n) = \frac{n^n/n!}{\sum_{k=0}^{n} n^k/k!}.$$

Ramanujan developed the following asymptotic expression for $\delta(n)$ (Knuth [11, p. 117]):

$$\frac{1}{\delta(n)} = \sqrt{\frac{\pi n}{2}} + \frac{2}{3} + \frac{1}{12}\sqrt{\frac{\pi}{2n}} - \frac{4}{135n} + O\left(\frac{1}{n^{3/2}}\right) \quad \text{as } n \to \infty.$$

Using this, we obtain, for a single-cell system,

$$T_{H,p}(r) - T(r, n) = O(n^{-1/2}) \qquad \text{as } n \to \infty. \tag{6.5}$$

(Since the FCAA and DCAA are identical for a single-cell system, this holds for DCAA as well.)

Now, we consider a multiple-cell system. Again, since $T(r, n)$ is monotonically nondecreasing and $T_{H,p}(r)$ is constant for $r \geq r_1$, we have,

$$d(n) = \max_{0 \leq r \leq r_1} T_{H,p}(r) - T(r, n).$$

Since $T_{H,p}(r) = r$ for $r \leq r_0$ and $r - T(r, n)$ is a monotonically nondecreasing function of r,

$$d(n) = \max_{r_0 \leq r \leq r_1} T_{H,p}(r) - T(r, n)$$

$$T(r, n) = \sum_{i=1}^{N} p_i r(1 - P_b(p_i r n, x_i n)) \tag{6.6}$$

so that

$$\begin{aligned} T_{H,p}(r) - T(r, n) &= \sum_{i=1}^{N} O(n^{-1/2}) = O(Nn^{-1/2}) \\ &= O(n^{-1/2}) \qquad \text{as } n \to \infty \end{aligned}$$

for finite N.

The determination of the constant implied by the O notation appears difficult in general, but can be determined in some special cases. Consider the systems (H, p) for which $r_0 = r_1$. Examples of such systems are those hypergraphs H for which p is either a favorable or an unfavorable traffic pattern (Theorem 5.3). For such systems, the FCAA that assigns $p_i r_0 n$ channels to cell i, for all values or r, is asymptotically optimal. Therefore,

$$\begin{aligned} d(n) &= T_{H,p}(r_0) - T(r_0, n) \\ &= \sum_{i=1}^{N} p_i r_0 P_b(p_i r_0 n, x_i n) = \sum_{i=1}^{N} p_i r_0 \delta(p_i r_0 n) \end{aligned}$$

since $p_i r_0 = x_i$. If $p_i = 1/N$ (uniform traffic),

$$d(n) = r_0 \delta(r_0 n/N). \tag{6.7}$$

Example 6.1: Consider a cellular system represented by the cycle on N vertices, i.e., cell v_i and v_j may not use the same channel if and only if $j = i \pm 1 \pmod N$, and let $p_i = 1/N$ for all i (uniform traffic). For these systems, $r_0 = r_1 = \lfloor N/2 \rfloor$, and using (6.7),

$$d(n) = \begin{cases} \frac{N}{2}\delta\left(\frac{n}{2}\right) & \text{if } N \text{ even} \\ \frac{N-1}{2}\delta\left(\frac{n(N-1)}{2N}\right) & \text{if } N \text{ odd.} \end{cases}$$

For large n,

$$d(n) \sim \begin{cases} \frac{N}{\sqrt{\pi n}} & \text{if } N \text{ even} \\ \frac{\sqrt{N(N-1)}}{\sqrt{\pi n}} & \text{if } N \text{ odd.} \end{cases} \qquad \square$$

Example 6.2: Consider the 7-cell example. The maximum in the definition of $d(n)$ is achieved for $r = 8/3$, independent of n. For $r = 8/3$, the asymptotically optimum FCAA assigns $n/3$ channels to all cells except the central cell and no channels to the central cell (cell 7). Therefore, $x_i = 0$ for the central cell and $p_i r = x_i = 1/3$ for each of the other cells. Using (6.4) and (6.6),

$$\begin{aligned} d(n) &= 2\delta(n/3) \sim 2\sqrt{\frac{6}{\pi n}} \\ &= 2.76n^{-1/2} \qquad \text{for large } n. \end{aligned} \qquad \square$$

Example 6.3: Consider the 19-cell example. The maximum in the definition of $d(n)$ is achieved for $r = 57/10$, independent of n. For $r = 57/10$, the asymptotically optimum FCAA assigns $3n/10$ channels to all cells except the central cell and no channels to the central cell (cell 10). Therefore, $x_i = 0$ for the central cell and $p_i r = x_i = 3/10$ for each of the other cells. Using (6.4) and (6.6),

$$\begin{aligned} d(n) &= \frac{27}{5}\delta(3n/10) \sim \frac{27}{5}\sqrt{\frac{20}{3\pi n}} \\ &= 7.87n^{-1/2} \qquad \text{for large } n. \end{aligned} \qquad \square$$

VII. Generalizations: Capacity of a Cellular System

In obtaining our asymptotic performance limits for channel assignment algorithms, we have assumed that the offered traffic is independent from cell to cell. Unfortunately, this assumption may be violated in practice. For example, if users are free to move from one cell to another when a call is in progress, the offered traffic in one cell will depend on the carried traffic in adjacent cells. Similarly, if the offered traffic includes requests for intercell calls, call requests can arrive simultaneously in two different cells. In this section, we will briefly discuss how our results extend to the case of traffic arrival models with intercell dependencies. We will see that such dependencies do not change the carried traffic function for $r \leq r_0$, but for $r > r_0$, they will in general decrease $T(r)$. The exact amount of the decrease, we do not know.

We first recall our upper bounds on the performance of channel assignment algorithms. In Section I, we defined the offered traffic A_i in cell i as the expected number of calls that would be in progress in cell i at a given time if all call requests in *that cell* could be honored. When the traffic is not independent from cell to cell, and in particular when the offered traffic in one cell may depend on the carried traffic in other cells, this definition needs to modified. In this more general case, we define the offered traffic A_i to be the expected number of calls that would be in progress in cell i at a given time if all call requests in *all cells* could be honored. (Of course, if the traffic is independent from cell to cell, the two definitions are equivalent.) With this more general definition, we can see that Theorems 3.1 and 3.2 hold, even for dependent traffic, because the proofs depend only on the assumptions that the carried traffic cannot exceed the offered traffic, and that any channel assignment algorithm produces a random n-multicoloring of the underlying hypergraph. Both of these assumptions hold even when the offered traffic is not independent from cell to cell. In short, traffic dependencies cannot increase the maximum possible carried traffic.

However, simple examples show that traffic dependencies can decrease the maximum possible carried traffic for $r > r_0$, and we are currently studying this interesting phenomenon. Nevertheless, if we make a certain plausible assumption about the asymptotic behavior of the traffic, we can show that $T_{H,p}(r) = r$ for $r \leq r_0$, i.e., Theorem 5.2b) holds, even when intercell dependencies are present. We will now discuss this extension of our results.

The assumption we need to make we call the "weak asymptotic traffic property (WATP)." In order to define this property, we extend the X fixed channel assignment algorithm introduced in Section IV to handle the case of dependent traffic. Given a vector $X = (X_1, \cdots, X_M)$ and an integer n, the X fixed channel assignment algorithm allocates $m_i = \sum_j \lfloor nX_j \rfloor a_{ij}$ channels to cell i. When a call request arrives in cell i, the algorithm assigns it any one of the m_i channels available in cell i if at least one is not in use; otherwise, it blocks the call. The algorithm does not distinguish among new calls, handoff calls, intercell calls, etc. In other words, it is, just as in the independent traffic case, N independent greedy algorithms, one for each cell in the system.

As in Section IV, we denote by $C(k, n)$ the carried traffic for this algorithm when the offered traffic is k Erlangs and there is a total of n channels. Recall that [see (4.2)]

$$\lim_{n \to \infty} \frac{m_i}{n} = x_i \qquad \text{for } i = 1, 2, \cdots, N$$

where $x_i = \sum_j X_j a_{ij}$. The WATP referred to above is

$$\text{If } x_i \geq p_i r \quad \text{for all } i,$$
$$\text{then } \lim_{n \to \infty} \frac{C(p_i rn, m_i)}{n} = p_i r \quad \text{for all } i.$$

In words, the WATP says that, if for every cell in the system the offered traffic is less than the number of available channels, then, asymptotically, the fraction of offered calls that are blocked approaches zero. We offer as a thesis the assertion that any reasonable traffic model which includes intercell dependencies must satisfy this property.

Note that the original ATP for the case of independent traffic states that

$$\lim_{n \to \infty} \frac{C(p_i rn, m_i)}{n} = \min(p_i r, x_i).$$

Since $\min(p_i r, x_i) = p_i r$ if $x_i \geq p_i r$, the ATP implies the WATP. The term "weak" refers to the fact that the WATP, unlike the original ATP, makes no assertion about the case when one or more of the $x_i < p_i r$. It appears to be very difficult to formulate a version of the ATP in this case because of the possible dependence of the offered traffic in one cell on the carried traffic in other cells. In any case, the WATP is enough to allow us to prove the result cited above, viz. $T_{H,p}(r) = r$ for $r \leq r_0$.

Theorem 7.1: Assume the WATP. Then, even in the presence of intercell dependencies, $T_{H,p}(r) = r$ *for all* $r \leq r_0$, *where* r_0 *is the value defined in Theorem 5.2b). Furthermore,* $T_{H,p}(r) < r$ *for all* $r > r_0$.

Proof: Given any feasible solution to the linear program of Theorem 5.2b), we consider the corresponding X fixed channel assignment algorithm that allocates $m_i = \sum_j \lfloor nX_j \rfloor a_{ij}$ channels to cell i. By (5.4), $x_i \geq p_i r$ for all i, and so by the WATP,

$$\lim_{n \to \infty} T_X(r) = \sum_{i=1}^{N} p_i r = r.$$

Therefore, $T_{H,p}(r) = r$ for all $r \leq r_0$. That $T_{H,p}(r) < r$ for $r > r_0$ follows from Theorem 3.2, which holds for dependent traffic, as we have seen above. □

Theorem 7.1 says that for very general traffic models, including models which include intercell dependencies, the quantity r_0 has the following significance.

- If the offered traffic intensity r exceeds r_0, then for any channel assignment algorithm, a positive fraction (independent of n) of all call requests must be blocked.
- On the other hand, if the offered traffic intensity is less than r_0, all call requests can be honored if the number of channels is sufficiently large.

Because of this strong and general property, we feel justified in calling r_0 the *capacity* of the cellular system, as measured in Erlangs per channel. For example, the capacity of our 7-cell example is $8/5 = 1.60$ Erlangs per channel, and that of the 19-cell example is $247/49 = 5.04$ Erlangs per channel. This result is significant enough to merit a restatement, with which we end our paper.

Theorem 7.2: The capacity $C(H, p)$ of the (H, p) cellular system, measured in Erlangs per channel, is given by the following linear program:

$$C(H, p)^{-1} = \min\left\{\sum_{j=1}^{M} X_j \colon X_j \geq 0, \sum_{j=1}^{M} X_j a_{ij} \geq p_i\right\}.$$

ACKNOWLEDGMENT

A preliminary version of this paper, dealing with uniform traffic on systems described by ordinary graphs, was presented in [13]. The extension to hypergraphs first appeared in [14]. Some other papers on the performance of cellular networks are [3], [7], [8], [10], [17], [18]. Some of the results in [9], whose topic is routing in circuit-switching networks, can also be applied to the analysis of cellular networks, and the relationship between the two problems is further explored in [15].

We thank the referees for their careful reading of the paper, their useful comments, especially on computational complexity, and for the additional references.

REFERENCES

[1] C. Berge, *Hypergraphs*. Amsterdam: North-Holland, 1989.
[2] D. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1987.
[3] D. E. Everitt and N. W. Macfadyen, "Analysis of multi-cellular mobile radiotelephone systems with loss," *British Telecommun. J.*, vol. 1, pp. 37–47.
[4] J. Franklin, *Methods of Mathematical Economics*. New York: Springer-Verlag, 1980.
[5] K. S. Gilhousen *et al.*, "On the capacity of a cellular CDMA system," *IEEE Trans. Veh . Technol.*, vol. 40, pp. 303–312, May 1991.
[6] D. J. Goodman, "Trends in cellular and cordless communications," *IEEE Comm. Mag.*, vol. 29, pp. 31–40, June 1991.
[7] F. P. Kelly, "Stochastic models of computer communication systems," *J. Roy. Statist. Soc.*, vol. B47, pp. 379–395, sect. 3, 1985.
[8] ——, "Blocking probabilities in large circuit-switched networks," *Adv. Appl. Prob.*, vol. 18, pp. 473–505, sect. 6.3, 1986.
[9] ——, "Loss networks," *Ann. Appl. Prob.*, vol. 1, pp. 317–378, sect. 3.3, 1991.
[10] P. Key, "Optimal control and trunk reservation in loss networks," *Prob. Eng. Appl. Sci.*, vol. 4, pp. 203–242, sect. 2.2, 1990.
[11] D. E. Knuth, *Fundamental Algorithms*. New York: Addison-Wesley, 1973.
[12] W. C. Y. Lee, "Overview of cellular CDMA," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 291–302, May 1991.
[13] R. J. McEliece and K. N. Sivarajan, "Performance limits for FDMA cellular telephone systems," in *Proc. 1990 Allerton Conf. Commun. Contr. Computing*, pp. 869–880.
[14] ——, "Performance limits for FDMA cellular systems described by hypergraphs," in *Proc. 3rd IEE Conf. Telecommun.*, Edinburgh, 1991, pp. 360–365.
[15] ——, "Maximizing marginal revenue in generalized blocking service networks," in *Proc. 1992 Allerton Conf. Commun. Contr. Computing*, pp. 455–464.
[16] J. W. Moon and L. Moser, "On cliques in graphs," *Israel J. Math.*, vol. 3, pp. 23–28, 1965.
[17] P.-A. Raymond, "Performance analysis of cellular networks," *IEEE Trans. Commun.*, vol. 39, pp. 1787–1793, Dec. 1991.
[18] D. Robinson, "The optimality of fixed channel assignment policies for cellular radio systems," *Adv. Appl. Prob.*, vol. 24, pp. 474–495, 1992.
[19] R. Syski, *Introduction to Congestion Theory in Telephone Systems*. Amsterdam: North-Holland, 1986.
[20] M. S. Wallace, "High capacity digital cellular communications through slow frequency-hopping CDMA," in *Proc. 1991 Allerton Conf. Commun., Contr., and Computing*.