

Performance of Global Load Balancing by Local Adjustment

BRUCE HAJEK, FELLOW, IEEE

Abstract—A set of M resource locations and a set of αM consumers are given. Each consumer requires a specified amount of resource and is constrained to obtain the resource from a specified subset of locations. The problem of assigning consumers to resource locations so as to balance the load among the resource locations as much as possible is considered. It is shown that there are assignments, termed uniformly most balanced assignments, that simultaneously minimize certain symmetric, separable, convex cost functions. The problem of finding such assignments is equivalent to a network flow problem with convex cost. Algorithms of both iterative and combinatorial type are given for computing the assignments. The distribution function of the load at a given location for a uniformly most balanced assignment is studied assuming that the set of locations each consumer can use is random. An asymptotic lower bound on the distribution function is given for M tending to infinity, and an upper bound is given on the probable maximum load. It is shown that there is typically a large set of resource locations that all have the maximum load, and that for large average loads the maximum load is near the average load.

I. PROBLEM FORMULATION

A. Integral Balance Problem

LARGE SCALE load balancing problems arise in many guises. They are encountered, for example, when several tasks are to be shared among several machines in a multiprocessor computer environment, or when traffic must be distributed over the communication links of a communication network. Such problems often call for a hierarchical solution in which a global balancing mechanism operates slowly using coarse, global information, while local balancing mechanisms use only readily available local information to orchestrate quick, local adjustments. In this paper we study a simple model for the problem faced by the local balancing mechanisms, which must typically operate under constraints. The constraints may be determined by a higher level control mechanism, or they may be imposed simply to reduce the cost (due, for example, to communication and storage) associated with shifting resource allocations.

Consider, for example, the case of "alternate routing" in a communication network. A station may be assigned

Manuscript received April 26, 1989; revised April 10, 1990. This work was supported by the Joint Services Electronics Program under Grant No. N00014-90-J-1270. This work was presented in part at the Eighteenth Conference on Stochastic Processes and Their Applications, University of Wisconsin, June 1989.

The author is with the Coordinated Sciences Laboratory and the Department of Electrical and Computer Engineering, University of Illinois, 1101 W. Springfield Ave., Urbana, IL 61801.

IEEE Log Number 9038009.

(on a slow time scale) multiple paths to a given destination. Based on current traffic readings (obtained from local information and on a fast time scale) the station allocates its traffic over the set of assigned paths. The question arises as to how well traffic can be controlled if there are, say, c alternate paths for each source-destination pair. This question is very difficult to answer in general. Our problem can be viewed as a very special case in which each path is only two links long and a restricted network topology is assumed. The assignment of radios to stations in a cellular radio system is a special case of alternate routing in networks which is fairly close to the model we consider. A constraint is that radios must be assigned to nearby stations, and the goal is to balance the load among the stations.

The main purpose of this paper is to study the performance of global load balancing by local adjustment. Specifically, we investigate how balanced the load is after balancing for a simple model that illustrates some interesting phenomena. For clarity, we will formulate the problem in terms of consumers and resource locations.

Consider a set U of consumers and a set V of resource locations, and let M denote the cardinality of V , $M = |V|$. Suppose that consumer u requires the use of m_u units of resource and can use at most $C_{u,v}$ units from location v . We will be most interested in the case that $N(u)$, defined by $N(u) = \{v \in V: C_{u,v} > 0\}$, is a small subset of V for each u . Also suppose that there is a load of size b_v on resource location v from consumers other than those in U . We refer to (U, V, C) as a network, m as the consumer demand vector, and b as the base load vector. An admissible assignment vector is a vector f , $f = (f_{u,v}: u \in U, v \in V)$, such that $0 \leq f_{u,v} \leq C_{u,v}$. The interpretation is that $f_{u,v}$ denotes the load on resource location v due to consumer u . An assignment vector f meets the consumer demand if

$$\sum_{v \in V} f_{u,v} = m_u, \quad \text{for } u \in U, \quad (1)$$

and the total load on resource location v , $x(v)$, is given by

$$x(v) = b_v + \sum_{u \in U} f_{u,v}, \quad \text{for } v \in V. \quad (2)$$

We are interested in the problem of choosing an admissible assignment vector f that meets the consumer demand such that the vector of loads, $x = (x(v): v \in V)$, is as

“balanced” as possible. A natural way to define “balanced” is to consider a convex optimization problem.

We shall first formulate the balance problem in case resources are only available in integral numbers of units. Thus, suppose that the constraint matrix C , base load vector b , and consumer demand vector m have integer coordinates. Let Φ_o be a convex function on \mathbb{Z}_+ , the set of nonnegative integers, and define $J_o(f)$ for an assignment vector f by

$$J_o(f) = \sum_{v=1}^M \Phi_o(x(v))$$

where x is the load vector, given as usual by (2). Consider the problem of finding an assignment vector f that solves the following optimization problem.

Problem (P_o):

$\min \{J_o(f) : f \text{ is an admissible, integral assignment meeting demand } m\}.$

Note that the mean load per location \bar{x} is constrained to be $(\sum_{u \in U} m_u + \sum_{v \in V} b_v)/M$ and so does not depend on f . The sample variance of the load per location is

$$V(x) = \frac{1}{M} \left(\sum_{v \in V} x(v)^2 \right) - \bar{x}^2.$$

The sample mean of the load at the same location as the location of a typical unit of load (i.e., the mean of $x(v)$ when location v is chosen with probability $x(v)/(M\bar{x})$) is

$$\bar{x}_1 = \frac{1}{M\bar{x}} \sum_{v \in V} x(v)^2 = \bar{x} + \frac{V(x)}{\bar{x}}.$$

Thus, both $V(x)$ and \bar{x}_1 are minimized when f is a solution to Problem P_o for $\Phi_o(k) = k^2$. For another example, if $\Phi_o(0) = 1$ and $\Phi_o(k) = 0$ for $k > 0$ then the solution f to Problem P_o minimizes the number of locations with zero load.

It is not clear what the best choice for Φ_o is, but fortunately to a large extent it does not matter. Indeed, we show in Section II that if an assignment solves Problem P_o for one strictly convex function Φ_o then it solves Problem P_o for any convex function Φ_o . A solution to P_o for some strictly convex function Φ_o will hence be called a *uniformly most balanced integral assignment*.

For some applications it may be more natural to think of V as a set of consumers, U as a set of resource providers where resource u has m_u units of resource to distribute among consumers in $N(u)$ subject to constraints, and f as an allocation. For f to be balanced in our original terminology can be thought of as a requirement that the allocation be fair. What we call uniformly most balanced assignments could then be called “maximally fair” assignments. To avoid confusion, we will henceforth only use the terminology in which U is called the set of consumers.

B. Continuous Assignment Problem

The continuous assignment problem is obtained by dropping the constraint that the assignment vector f must have integral coordinates. The assumption that the constraint matrix C , base load vector b , and consumer demand vector m have nonnegative integral coordinates is replaced by the assumption that they have nonnegative real coordinates. Let Φ be a convex function on the nonnegative reals, \mathcal{R}_+ , and define $J(f)$ for an assignment vector f by

$$J(f) = \sum_{v=1}^M \Phi(x(v))$$

where x is the load vector, given by (2). The “continuous” load balancing problem, called Problem P , is as follows.

Problem (P):

$\min \{J(f) : f \text{ is an admissible assignment meeting demand } m\}.$

As in the case of the integral assignment problem, different choices of Φ give seemingly different properties of the solution to problem P . We shall show in Section II that if f is a solution to problem P for one strictly convex function Φ then it is a solution for any convex function Φ , and in that case we call f a *uniformly most balanced assignment*.

A close connection between uniformly most balanced assignments and uniformly most balanced integral assignments will also be given in Section II. Section III describes algorithms for computing both types of assignments.

C. Random Network Model

Situations may arise in which the network (U, V, C) and demand vector m are not known in advance, so that it may be reasonable to assume that C and/or m are random. If U and V are large, one might expect that balanced assignments have particular properties with high probability. In this paper we will investigate a simple random model. Let M , αM and c be positive integers. Let U and V be disjoint sets with $|U| = \alpha M$ and $|V| = M$. For each $u \in U$ let $N(u)$ be a random subset of V with $|N(u)| = c$ such that all $\binom{M}{c}$ possibilities are equally likely. Assume that $(N(u) : u \in U)$ are mutually independent. Let $C_{u,v} = 1$ if $v \in N(u)$, and $C_{u,v} = 0$ otherwise. Assume that each consumer has one unit of demand, so that $m_u = 1$ for all u , and assume that the base load vector b is identically zero. Results of some Monte Carlo simulations of this random network are described in Section VI. The reader is encouraged to examine that section before reading the rest of the paper.

Consider a fixed resource location v_o in V . Since the constraints on the consumers are random, the load resulting at location v_o under a uniformly most balanced assignment is also random. Let $F(\tau; M, \alpha, c)$ denote the probability that said load is less than or equal to τ . Note that the assignment need not be integral. However, be-

cause of the close connection between integral and continuous uniformly most balanced assignments established in Section II, it is easy to obtain from $F(\tau; M, \alpha, c)$ the corresponding distribution function for uniformly most balanced *integral* assignments, so that in Sections IV and V we concentrate on $F(\tau; M, \alpha, c)$. In those sections we investigate the distribution function $F(\tau; M, \alpha, c)$ in the limit of large M for α and c fixed.

II. PROPERTIES OF UNIFORMLY MOST BALANCED ASSIGNMENTS

As we shall soon show, the optimization problems we consider can be viewed as network flow problems with convex cost structure. In this section we derive the necessary and sufficient conditions for optimality following standard procedures such as can be found in [9], [12]. These conditions imply special properties for our model, setting the stage for the analysis of balanced loads in the later sections of the paper. In particular, we establish the existence of *uniformly* most balanced assignments by noting that the optimality conditions do not depend on the choice of which convex cost function is used. We also expose a strong connection between the integral and continuous versions of the balancing problem.

A. Incremental Flows and Integral Assignments

Recall from Subsection I.A that the cost function $J_o(f)$ in the integral balance problem is defined for *integral* admissible assignment vectors, and is based on a convex function Φ_o defined on \mathbb{Z}_+ . Given Φ_o , let Φ'_o denote the function defined on \mathcal{R}_+ such that for each nonnegative integer k , $\Phi'_o(x) = \Phi_o(k) + (x - k)(\Phi_o(k + 1) - \Phi_o(k))$ for $k \leq x \leq k + 1$. Let J'_o be defined in the same way as J_o except with Φ_o replaced by Φ'_o , and consider the continuous assignment Problem P'_o defined as follows.

Problem (P'_o):

$$\min \{ J'_o(f) : f \text{ is an admissible assignment meeting demand } m \}.$$

Problem P'_o is an example of a problem of the type P considered in the introduction, though the cost function is piecewise affine in the way we have specified.

A consumer-resource network (U, V, C) corresponds naturally to a single commodity, single destination flow network \mathcal{N} , $\mathcal{N} = (S, L, C)$, and an admissible assignment vector f meeting demand m (that is, satisfying $0 \leq f_{u,v} \leq C_{u,v}$ for $u \in U$ and $v \in V$ and satisfying (1) and (2)) corresponds to a flow defined on \mathcal{N} as follows (see Fig. 1). The node set is $S = U \cup V \cup \{d\}$ where U is the set of consumers, V is the set of resource locations, and d is another node. The link set L is equal to $S \times S$, and the capacity assignment $C = (C_e : e \in L)$ is determined as follows. Let C_e for $e = (u, v)$ with $u \in U$ and $v \in V$ simply be the given upper bound on how much resource consumer u can obtain from resource location v , let $C_e = \infty$ for $e = (v, d)$ with $v \in V$, and let $C_e = 0$, otherwise. We

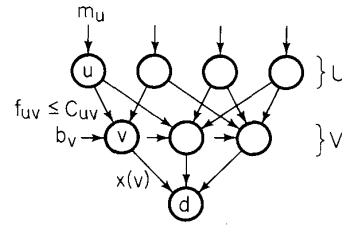


Fig. 1. Interpretation of (U, V, C) as flow network and assignment vector f as a directed flow.

can regard the demand vector m and base load vector b as input flow vectors, the coordinate $f_{u,v}$ of an admissible assignment vector f that meets demand m as the flow on link (u, v) for $u \in U$ and $v \in V$, and $x(v)$ as the flow on link (v, d) for $v \in V$. Equations (1) and (2) simply state that the net flow at nodes in U and V is zero. We allow the net flow into the destination node d to be nonzero. With the convention that all unspecified flow rates are zero, the pair (f, x) can thus be viewed as a flow on \mathcal{N} . We shall thus use $f_{v,d}$ as another way to denote the load, $x(v)$, on resource location v for assignment f .

Given a base load vector b , consumer demand vector m , and an admissible integral assignment f for (U, V, C) meeting demand m , we define a new flow network $\mathcal{N}(f)$ by $\mathcal{N}(f) = (S, L, C')$ where $S = U \cup V \cup \{d\}$, $L = S \times S$ and the capacity assignment C' is given by (only nonzero capacities are given):

$$C'_{u,v} = C_{u,v} - f_{u,v}$$

$$C'_{v,u} = f_{u,v}$$

$$C'_{d,v} = f_{v,d}$$

$$C'_{v,d} = \infty.$$

The network $\mathcal{N}(f)$ will be called the incremental network for f , since, as we will see, certain small flows for it correspond to flows near f in the original network (U, V, C) . See Fig. 2 for an example. We define a weight assignment w on the links of $\mathcal{N}(f)$ by (for $v \in V$)

$$w(v, d) = \Phi_o(x(v) + 1) - \Phi_o(x(v))$$

$$w(d, v) = -[\Phi_o(x(v)) - \Phi_o(x(v) - 1)]$$

and all other links have weight zero. A flow on $\mathcal{N}(f)$ is a vector $h = (h_e : e \in L)$ such that $h_e \geq 0$ for all $e \in L$, and the weight of such a flow is defined by

$$w(h) = \sum_{v \in V} (h_{d,v} w_{d,v} + h_{v,d} w_{v,d}).$$

A flow on $\mathcal{N}(f)$ is called a *circulation* if the net flow into any node is zero, and it is called *simple* if for any link (a, b) of $\mathcal{N}(f)$ either the flow assigned to (a, b) is zero or the flow assigned to (b, a) is zero.

Admissible assignments g for (U, V, C) meeting demand m are in one-to-one correspondence with simple circulations h on $\mathcal{N}(f)$ with $0 \leq h \leq C'$. The correspon-

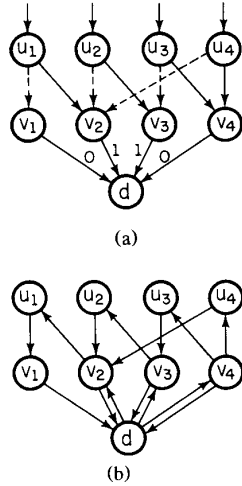


Fig. 2. (a) Flow f on network \mathcal{N} such that $C_{u,v}=1$ for $v \in N(u)$. Solid lines denote positive flow and dashed lines denote positive capacity but zero flow. (b) Corresponding network $\mathcal{N}(f)$. Improving elementary flow corresponds to sequence of nodes $(d, v_4, u_4, v_2, u_1, v_1)$.

dence is given by

$$\begin{aligned} h_{u,v} &= (g_{u,v} - f_{u,v})^+ \\ h_{v,u} &= (g_{u,v} - f_{u,v})^- \\ h_{v,d} &= (g_{v,d} - f_{v,d})^+ \\ h_{d,v} &= (g_{v,d} - f_{v,d})^- \end{aligned} \quad (3)$$

(where $a^+ = \max(a, 0)$ and $a^- = (-a)^+$) and conversely

$$\begin{aligned} g_{u,v} &= f_{u,v} + h_{u,v} - h_{v,u} \\ g_{v,d} &= f_{v,d} + h_{v,d} - h_{d,v}. \end{aligned} \quad (4)$$

Furthermore, if $|f - g| \leq 1$ where $|f - g| = \max\{|f_e - g_e| : e \in L\}$ then

$$J'_o(g) = J'_o(f) + w(h). \quad (5)$$

Equation (5) is the main reason for our interest in $\mathcal{N}(f)$.

An *elementary flow* on $\mathcal{N}(f)$ is a circulation h such that for some $p \geq 3$ there are distinct nodes r_1, r_2, \dots, r_{p-1} so that with $r_p = r_1$ it holds that: $(C'_e > 0$ and $h_e = 1)$ if $e = (r_i, r_{i+1})$ for $0 \leq i \leq p-1$, and $h_e = 0$, otherwise. Finally, an *improving elementary flow* on $\mathcal{N}(f)$ is defined to be an elementary flow corresponding to a sequence of nodes of the form

$$(d, v_1, u_1, v_2, \dots, u_{n-1}, v_n), \quad (6)$$

where $u_i \in U$ and $v_i \in V$ for each i , and such that $x(v_1) \geq x(v_n) + 2$.

Theorem 1: Suppose that Φ_o restricted to \mathbb{Z}_+ is strictly convex, and suppose that f is an admissible integral assignment. Then the following are equivalent:

- f is a solution to Problem P'_o .
- f is a solution to Problem P_o .
- There are no improving elementary flows on $\mathcal{N}(f)$.
- $w(h) \geq 0$ for all elementary flows h on $\mathcal{N}(f)$.

- $w(h) \geq 0$ for all simple circulations h on $\mathcal{N}(f)$ with $0 \leq h \leq C'$.
- $J'_o(f) \leq J'_o(g)$ for all admissible assignments g meeting the demand m with $|f - g| \leq 1$.

Proof: We prove that a) implies b), b) implies c), c) implies d), d) implies e), e) implies f) and f) implies a). First, a) implies b) since P_o is obtained from P'_o by reducing the set of assignments considered. Now suppose that b) is true and let h be an elementary flow on $\mathcal{N}(f)$ corresponding to a sequence of the form given in (6). Let g be the flow defined by (4). Then g is an admissible integral flow so that $J(f) \leq J(g)$, and (5) holds. Hence, $w(h) \geq 0$, and we also have

$$w(h) = \Phi_o(x(v_n) + 1) - \Phi_o(x(v_n)) - [\Phi_o(x(v_1)) - \Phi_o(x(v_1) - 1)]. \quad (7)$$

By the strict convexity of Φ_o this implies that $x(v_1) \leq x(v_n) + 1$, so that h is not an improving elementary flow. Since h was an arbitrary elementary flow corresponding to a sequence of the form in (6), there are no improving elementary flows, and the proof that b) implies c) is complete.

To prove that c) implies d), assume that c) is true and consider an elementary flow h on $\mathcal{N}(f)$. If $w(h) \neq 0$ then h must correspond to a sequence of the form in (6). In view of (7) and the fact that h is not an improving elementary flow, $w(h) \geq 0$. Thus c) implies d). The fact that d) implies e) follows from the fact that any circulation h on $\mathcal{N}(f)$ can be expressed as a linear combination with positive coefficients of elementary flows on $\mathcal{N}(f)$ [12, p. 104]. Equation (5) implies that e) and f) are equivalent. Finally, the cost function J'_o is a convex function, the feasible set for Problem P'_o is convex, and f) implies that f is a local minimum of J'_o . Thus, f) implies a). \square

The assumption that Φ_o is a *strictly* convex function was only used to prove that b) implies c). Thus, c) implies a) and b) even if Φ_o is convex but not strictly convex. The equivalence of b) and c) in the Theorem and the fact that condition c) does not depend on which strictly convex function Φ_o is used imply the following corollary.

Corollary 1: An integral assignment vector f is a solution to Problem P_o for a given strictly convex function Φ_o if and only if it is a solution to Problems P_o and P'_o simultaneously for all convex functions Φ_o .

Any solution f to Problem P_o for some strictly convex function Φ_o will be called a *uniformly most balanced assignment* (for demand m and base load b). Problem P_o might not have a unique solution, but the sample distribution function of the load, defined by

$$\frac{|\{v \in V : x(v) \leq k\}|}{M}, \quad \text{for } k \in \mathbb{Z}_+, \quad (8)$$

is unique.

Corollary 2: The sample distribution function given in (8) is the same for all uniformly most balanced integral assignments.

Proof: For any $k \in \mathbb{Z}_+$, the function $(i-k)^+$ is convex in i , and we define $J_{o,k}(f)$ to equal $J_o(f)$ for the choice $\Phi_o(i) = (i-k)^+$. Since the sample distribution function for an admissible assignment vector f can be written as $1 - (J_{o,k}(f) - J_{o,k+1}(f))/|V|$, Corollary 2 follows from Corollary 1. \square

Corollary 3: Uniformly most balanced integral assignments minimize the maximum load, $x_{\max}(f) = \max\{x(v) : v \in V\}$, over all admissible integral assignments meeting the demand m .

Proof: Since $x_{\max}(f) \leq k$ if and only if $J_{o,k}(f) = 0$, Corollary 3 follows from Corollary 1. \square

B. Continuous Assignments

We will find conditions for an assignment to be a solution to the continuous assignment problem, Problem P , and find that the conditions are the same for any strictly convex function Φ .

Theorem 2: Suppose that Φ is a convex function on \mathbb{R}_+ and that f is an admissible assignment vector. Then condition a) next implies condition b), and if Φ is strictly convex then condition b) implies condition a).

- a) For all $u \in U$ and all $v, v' \in V$, $f_{u,v}(C_{u,v'} - f_{u,v'}) = 0$ whenever $x(v) > x(v')$.
- b) f is a solution to Problem P .

Proof: First, suppose that condition a) is true. For each $u \in U$ define $\hat{x}(u)$ and π_u by

$$\hat{x}(u) = \max\{x(v) : v \in V, f_{u,v} > 0\}$$

and $\pi_u = \Phi'_+(\hat{x}(u))$ where Φ'_+ is the right-hand derivative of Φ . Intuitively, the constants $(\pi_u : u \in U)$ are choices of Lagrange multipliers. Consider the function H defined for assignment vectors g by

$$H(g) = J(g) - \sum_{u \in U} \pi_u \left(\sum_{v \in V} g_{u,v} \right) - m_u. \quad (9)$$

We claim that f minimizes H over the set of all admissible assignment vectors, including those that do not satisfy the demand m . The claim implies condition b) since $J(g) = H(g)$ whenever g is an admissible assignment that does satisfy demand m .

To prove the claim we will show that $H(f+h) \geq H(f)$ whenever $f+h$ is admissible. Since $H(f+\epsilon h)$ is convex in ϵ , it is enough to show that the limit of $(H(f+\epsilon h) - H(f))/\epsilon$ as ϵ decreases to zero is nonnegative. The limit is equal to

$$\sum_{u,v: h_{u,v} > 0} h_{u,v} [\Phi'_+(x(v)) - \pi_u] + \sum_{u,v: h_{u,v} < 0} h_{u,v} [\Phi'_-(x(v)) - \pi_u] \quad (10)$$

where Φ'_- is the left-hand derivative of Φ . Consider first a pair u, v such that $h_{u,v} > 0$. We see that $f_{u,v} < C_{u,v}$ since $f+h$ is admissible, so that $\hat{x}(u) \leq x(v)$ by condition

a), which in turn implies that $\pi_u \leq \Phi'_+(x(v))$. Thus the first sum in (10) is nonnegative. Consider next a pair u, v such that $h_{u,v} < 0$. We see that $f_{u,v} > 0$ since $f+h$ is admissible, so that $\hat{x}(u) \geq x(v)$ by the definition of $\hat{x}(u)$, which in turn implies that $\pi_u \geq \Phi'_-(x(v))$. Thus the second sum in (10) is also nonnegative. We have thus established the claim, and hence we have shown that condition a) implies condition b).

We turn to the proof of the second assertion of the theorem, so suppose that Φ is strictly convex. If a) is not true then there are $u \in U$ and $v, v' \in V$ such that $f_{u,v} > 0$, $C_{u,v'} > f_{u,v'}$ and $x(v) > x(v')$. Hence decreasing $f_{u,v}$ by ϵ and increasing $f_{u,v'}$ by ϵ for a sufficiently small value of ϵ leads to an admissible assignment with strictly smaller cost, implying that b) is not true. Thus, b) implies a) under the assumption that Φ is strictly convex. \square

Theorem 2 and the fact that condition a) in the theorem does not depend on Φ imply the following corollary.

Corollary 4: An assignment vector f is a solution to Problem P for a given strictly convex function Φ if and only if it is a solution to Problem P simultaneously for all convex functions Φ .

Any solution f to Problem P for some strictly convex function Φ will be called a uniformly most balanced assignment (for demand m and base load b .) Problem P might not have a unique solution, but the load vector x is unique.

Corollary 5: The load vector $x = (x(v) : v \in V)$ is the same for all solutions to Problem P .

Proof: Suppose that Φ is strictly convex and continuous. The cost function J is then a strictly convex, continuous function of the load vector x , and as f ranges over the set of all admissible assignment vectors meeting demand m , the load vector x ranges over a compact, convex set. The corollary is thus a consequence of the fact that a strictly convex, continuous function defined on a compact, convex set is minimized at exactly one point. \square

Corollary 6: Uniformly most balanced assignments minimize the maximum load, $x_{\max}(f) = \max\{x(v) : v \in V\}$, over all admissible assignments meeting the demand m .

Proof: Let $J_x(f)$ equal $J(f)$ for the special case $\Phi(z) = (z-x)^+$. Since $x_{\max}(f) \leq x$ if and only if $J_x(f) = 0$, Corollary 6 follows from Corollary 4. \square

For the final corollary, we assume that $C_{u,v} = +\infty$ for $v \in N(u)$. Condition a) in Theorem 2 simplifies to the requirement that for all $u \in U$ and all $v, v' \in N(u)$, $f_{u,v} = 0$ whenever $x(v) > x(v')$. Define $\gamma(A)$ for nonempty subsets $A \subset V$ by

$$\gamma(A) = \sum_{u: N(u) \subset A} m_u + \sum_{v \in A} b_v. \quad (11)$$

Note that $\gamma(A)$ is the smallest total load that can be assigned to the resource locations in A . It is easy to deduce the following corollary from Theorem 2.

Corollary 7: Let x denote the load vector for a uniformly most balanced assignment f , and let $\hat{A} = \{v: x(v) = x_{\max}(f)\}$. Then

$$x_{\max}(f) = \frac{\gamma(\hat{A})}{|\hat{A}|} = \max \left\{ \frac{\gamma(A)}{|A|} : A \subset V, A \neq \emptyset \right\}. \quad (12)$$

Corollary 7 will be useful in Section V.

C. Relationship Between Integral and Continuous Assignments

The load distributions for the balance problem with and without the constraint that assignments be integral are closely related. Consider a network (U, V, C) with demand vector m and base load vector b such that C , m and b all have integral coordinates. Let f_o be a uniformly most balanced *integral* assignment meeting demand m and let f be a uniformly most balanced assignment meeting demand m . Let $x_o = (x_o(v): v \in V)$ and $x = (x(v): v \in V)$ denote the respective load vectors.

Theorem 3:

a) For any nonnegative integer k ,

$$\begin{aligned} & |\{v \in V: x_o(v) \leq k\}| \\ &= |\{v \in V: x(v) \leq k\}| \\ &+ \sum_{\tau: k < \tau < k+1} (k+1-\tau) |\{v: x(v) = \tau\}|. \quad (13) \end{aligned}$$

b) $\max\{x_o(v): v \in V\} = \max\{x(v): v \in V\}$.

c) $\min\{x_o(v): v \in V\} = \min\{x(v): v \in V\}$.

Proof: Recall that we defined $J_{o,k}$ to equal J_o for the choice $\Phi_o(i) = (i-k)^+$. Now let $J'_{o,k}$ denote the affine extension of $J_{o,k}$, which is the same as the function J for the special case that $\Phi(t) = (t-k)^+$ for $t \in \mathbb{R}_+$. Corollaries 1 and 4, respectively, imply that f_o and f both minimize $J'_{o,k}$ over all admissible assignments meeting demand m , so that $J'_{o,k}(f_o) = J'_{o,k}(f)$. The proof of a) is completed by noting that the left-hand side of (13) is equal to $|V| - (J'_{o,k}(f_o) - J'_{o,k+1}(f_o))$ while the right-hand side of (13) is equal to $|V| - (J'_{o,k}(f) - J'_{o,k+1}(f))$. Parts b) and c) of the theorem easily follow from part a). \square

III. LOAD BALANCING ALGORITHMS

A. Overview

Algorithms for computing uniformly most balanced assignments and uniformly most balanced integral assignments will be described in this section. Throughout the section we let $\Phi(x) = x^2$ and $\Phi_o(k) = k^2$, and we seek solutions to Problems P and P_o . Problem P can be viewed as a convex assignment problem or as a convex network flow problem. Many algorithms can potentially be used to solve Problem P , such as dual relaxation [2], projected gradient [1], method of multipliers [1], the Frank-Wolf method [12], the recursive conjugate gradient method [9], and others. Many of the algorithms presented

in the literature do not apply directly to Problem P due to the fact that the cost function $J(f)$ is not a *strictly* convex function of the assignment vector f . (A feasible direction method, called the "optimal distribution algorithm" in [12], is an exception). However, modifications of these algorithms based on the penalty method [1] can be devised. We will not attempt to survey the possible methods, for our goal in this paper is to study balanced assignments, not how to find them. However, in order to briefly illustrate some of the possibilities, in the next two subsections we give two methods for solving Problem P —one is iterative and the other is combinatorial. The algorithms are not necessarily the most efficient possible. In the final subsection we show how a solution to P_o can be found using a solution to P .

B. An Asynchronous Relaxation Algorithm for P

The well-known method of coordinate descent will be used in this section. Given an assignment vector f for (U, V, C) and given $u \in U$, we let $T_u f$ denote the new assignment that results by minimizing $J(f)$ with respect to $(f_{u,v}: v \in N(u))$ for u and $(f_{u',v}: u' \neq u, v \in V)$ fixed, subject to the constraints: $0 \leq f_{u,v} \leq C_{u,v}$ for $v \in N(u)$ and $\sum_v f_{u,v} = m_u$. It is straight-forward to implement the mapping T_u on a computer. Let $f^{(0)}$ be an arbitrary admissible assignment meeting demand m (one is easy to find if one exists) and let $(s_i)_{i \geq 1}$ be a sequence with $s_i \in U$ for all i . Define a sequence of assignments, $(f^{(i)})_{i \geq 0}$ recursively by $f^{(i+1)} = T_{s_i} f^{(i)}$. We assume that there is an integer Γ so that for any integer k and any $u \in U$, $s_i = u$ for some i with $k \leq i \leq k + \Gamma$.

Theorem 4: The cost $J(f^{(i)})$ is monotone nonincreasing in i and converges to the minimum cost for Problem P . Any limit point of the sequence $f^{(i)}$ is a solution to Problem P .

Before proving the theorem, we state and prove a lemma.

Lemma 1: For any admissible assignment f and $u \in U$,

$$|f - T_u f|^2 \leq (J(f) - J(T_u f))$$

where

$$|f - g| = \max \{|f_{u,v} - g_{u,v}|: u \in U, v \in V\}.$$

Proof of Lemma 1: Let $(x(v): v \in V)$ denote the load vector corresponding to f and let $(\tilde{x}(v): v \in V)$ denote the load vector corresponding to \tilde{f} , where $\tilde{f} = T_u f$. Define a number π by $\pi = \max\{\tilde{x}(v): \tilde{f}_{u,v} > 0\}$. We claim that either $x(v) \geq \tilde{x}(v) \geq \pi$ or $x(v) \leq \tilde{x}(v) \leq \pi$ for $v \in N(u)$. Indeed, if $\tilde{x}(v) > \pi$ then $\tilde{f}_{u,v} = 0 < f_{u,v}$ so $x(v) \geq \tilde{x}(v)$. On the other hand, if $\tilde{x}(v) < \pi$ then, by the definition of \tilde{f} , $\tilde{f}_{u,v} = C_{u,v} \geq f_{u,v}$ so that $x(v) \leq \tilde{x}(v)$, and the claim is proved. The claim and the fact that $\sum_{v \in N(u)} (x(v) - \tilde{x}(v))$

= 0 yields that

$$\begin{aligned}
 J(f) - J(T_u f) &= \sum_{v \in N(u)} [x(v)^2 - \tilde{x}(v)^2] \\
 &= \sum_{v \in N(u)} [(x(v) - \pi)^2 - (\tilde{x}(v) - \pi)^2] \\
 &\geq \sum_{v \in N(u)} (x(v) - \tilde{x}(v))^2 \\
 &\geq \max_{v \in N(u)} |x(v) - \tilde{x}(v)|^2 = |f - T_u f|^2
 \end{aligned}$$

and the lemma is proved. \square

Proof of Theorem 4: The monotonicity is obvious, and since J is a nonnegative, continuous function it implies that $\lim_{i \rightarrow \infty} J(f^{(i+1)}) - J(f^{(i)}) = 0$ and that the limit $\lim_{i \rightarrow \infty} J(f^{(i)})$ exists and is equal to $J(f^*)$ for any limit point, f^* , of $(f^{(i)})_{i \geq 0}$. By Lemma 1 it follows that $\lim_{i \rightarrow \infty} |f^{(i+1)} - f^{(i)}| = 0$. Since T_u is a continuous mapping for each u we conclude that $T_u f^* = f^*$ for all $u \in U$ for any limit point f^* . By Theorem 2, that implies that f^* is a solution to Problem P . \square

Remark: A result similar to Theorem 4 is described for a load balancing problem by Bertsekas and Tsitsiklis [2], generalizing earlier work of Cybenko. They consider a computation model with delays, as could arise in distributed computation. They do not constrain where load can be moved, so that equal load is obtained at all resource locations in the limit.

C. An $O(N^4)$ Algorithm for P

The algorithm OPTFLO in [5] solves a certain network evacuation problem, and Problem P here can be viewed as a special case. The result is a method to solve Problem P in $O(N^4)$ computations, where $N = |U| + |V|$. We will briefly describe the method.

The average load per resource location for any assignment for (U, V, C) that satisfies the demand m is given by

$$a = \frac{\sum_{u \in U} m_u + \sum_{v \in V} b_v}{M}.$$

Suppose that f is a solution to Problem L , defined as follows. (The problem is especially simple if the base load vector b is zero.)

Problem (L): $\max_{\sum_{v \in V} x(v)}$ over f subject to (for all $u \in U$ and $v \in V$),

$$0 \leq f_{u,v} \leq C_{u,v}, \quad \sum_{v' \in N(u)} f_{u,v'} \leq m_u$$

and

$$x(v) \leq \max(a, b_v), \quad (14)$$

where $x(v) = b_v + \sum_{u' \in U} f_{u',v}$ as usual. Note that f is an

admissible assignment but may not meet demand m . Define

$$V_0 = \{v: x(v) < a\}$$

$$U_0 = \{u: f_{u,v} < C_{u,v}, \text{ for some } v \in V_0\}$$

$$V_1 = \{v: x(v) \geq a\}$$

$$U_1 = \{u: f_{u,v} = C_{u,v}, \text{ for all } v \in V_0\}$$

Clearly $f_{u,v} = 0$ for $(u, v) \in U_0 \times V_1$ and $f_{u,v} = C_{u,v}$ for $(u, v) \in U_1 \times V_0$.

Suppose that V_0 and V_1 are both not empty. Given a vector $(e_i: i \in I)$ and $I_0 \subset I$, we write $e|_{I_0}$ to denote the vector $(e_i: i \in I_0)$. Construct an assignment vector \tilde{f} for (U, V, C) as follows: $\tilde{f}_{u,v} = 0$ for $(u, v) \in U_0 \times V_1$; $\tilde{f}_{u,v} = C_{u,v}$ for $(u, v) \in U_1 \times V_0$; and for $i \in \{0, 1\}$, define $\tilde{f}|_{U_i \times V_i}$ to be a UMB assignment for the network $(U_i, V_i, C|_{U_i \times V_i})$ with demand vector $m|_{U_i}$ and base load vector $(b_v + \sum_{u \in U_1} C_{u,v}: v \in V_0)$ if $i = 0$ and $b|_{V_i}$ if $i = 1$. The following theorem is a special case of Theorem 3 in [5], to which we refer the reader for a proof. The main idea is to check that \tilde{f} satisfies the optimality conditions of Theorem 2.

Theorem 5: Suppose there exists a feasible assignment for (U, V, C) meeting demand m . Then Problem L has a solution f . If $V_0 = \emptyset$ then f is a UMB assignment. Otherwise \tilde{f} is well defined and is a UMB assignment.

Problem L is a maximum flow problem and can thus be solved in $O(N^3)$ steps [11]. The subproblems of finding $\tilde{f}|_{U_i \times V_i}$ for $i = 1, 2$ are again problems of the same form as P , and hence each can be either solved or reduced to two smaller problems by solving the corresponding versions of L . Continuing in this way provides a recursive method for solving problem P . The number of maximum flow problems that need to be solved can be shown [5] to equal $\{\sum_{v \in V} x(v)\}$ where x is the load vector for a UMB assignment. This number is at most M and is often much smaller. Hence, the recursive algorithm requires at most $O(N^4)$ computations.

D. An $O(N^3)$ Reduction of P_0 to P

Suppose that the constraint matrix C , the demand vector m , and the base load vector b are all integral vectors, suppose that \tilde{f} is a UMB assignment for (U, V, C) , m and b , and let \tilde{x} denote the corresponding load vector. Consider the following constraints on an assignment vector f (for all $u \in U$ and $v \in V$):

$$[\tilde{f}_{u,v}] \leq f_{u,v} \leq [\tilde{f}_{u,v}] \quad \sum_{v' \in V} f_{u,v'} = m_u \quad (15)$$

$$[\tilde{x}_v] \leq b_v + \sum_{u' \in U} f_{u',v} \leq [\tilde{x}_v]. \quad (16)$$

Theorem 6: An integral vector f satisfying the constraints in (15) and (16) can be found in $O(N^3)$ computations, and such f is a UMB integral assignment.

Proof: The constraints in (15) and (16) are integral constraints for a network flow problem, and a (not neces-

sarily integral) solution, namely \tilde{f} , to the constraints exists. Thus, the maximum flow algorithm [11] provides an $O(N^3)$ algorithm for finding an integral solution f . (This argument is the basis of the "integrality theorem" [12]). It remains to show that f solves problem P_o , and to do so we shall prove that $J'_o(\tilde{f}) = J'_o(f)$, or equivalently that

$$\sum_{v \in V} (\Phi'_o(\tilde{x}_v) - \Phi'_o(x_v)) = 0. \quad (17)$$

Given a real number c , let $B(c) = \{v: \tilde{x}_v = c\}$. Since the collection of sets $\{B(c): -\infty < c < +\infty\}$ is a partition of V , it suffices to prove (17) with V replaced by $B(c)$ for an arbitrary value of c . Furthermore, since $[c] \leq x_v, \tilde{x}_v \leq [c]$ for $v \in B(c)$ and since Φ'_o is affine over the interval $[[c], [c]]$, it suffices to prove for any c that $\sum_{v \in B(c)} (\tilde{x}_v - x_v) = 0$, or equivalently, that $\sum_{u \in U} \sum_{v \in B(c)} (\tilde{f}_{u,v} - f_{u,v}) = 0$. To finish the proof we will establish that for any $u \in U$,

$$\sum_{v \in B(c)} (\tilde{f}_{u,v} - f_{u,v}) = 0. \quad (18)$$

Consider two cases. First, if $\tilde{f}_{u,v} \in \{0, C_{u,v}\}$ (so $f_{u,v}$ is an integer) for all $v \in B(c)$, then $\tilde{f} = f_{u,v}$ for all $v \in B(c)$ and (18) is true. On the other hand, if $0 < \tilde{f}_{u,v} < C_{u,v}$ for some $v' \in B(c)$, then the optimality conditions in Theorem 2 applied to \tilde{f} yield that for all $v \notin B(c)$, $\tilde{f}_{u,v} \in \{0, C_{u,v}\}$ (so $f_{u,v}$ is an integer). Hence $\tilde{f}_{u,v} = f_{u,v}$ for all $v \notin B(c)$. Since f and \tilde{f} both meet demand m , (18) is true in this case as well. The theorem is established. \square

IV. ASYMPTOTIC ANALYSIS—THE TREE METHOD

A. Orientation

We will study the load distribution function $F(\tau; M, \alpha, c)$, defined in Section I-C, in the limit as M tends to infinity. The method we call the tree method will be used in this section, while the moment method will be used in the next section to derive different results. For simplicity, we will only consider the case $c=2$ in this section—thus $|N(u)|=2$ for each $u \in U$ —and we will write $F(\tau; M, \alpha)$ instead of $F(\tau; M, \alpha, 2)$. Also, we will think of the random network (U, V, N) as a multigraph (i.e., a graph in which there can be more than one edge between a pair of nodes) with the set V of resource locations being the set of nodes, and the set U of consumers indexing the edges. We thus say that nodes v and \hat{v} are neighbors if $N(u) = \{v, \hat{v}\}$ for some $u \in U$. The distance between nodes v and \hat{v} is the smallest integer p , $p \geq 0$, so that there is a sequence $v = v_0, v_1, \dots, v_p = \hat{v}$ such that v_i and v_{i+1} are neighbors for $0 \leq i < p$, with the provision that if no such p exists (i.e., if v and \hat{v} are in different components of the multigraph) then the distance between them is $+\infty$.

In the next subsection, we will exploit the fact that the load at a fixed node v_c after balancing is less than one if and only if the component of the multigraph containing v_o is a tree. For $\alpha \leq 0.5$ that will be sufficient to determine the limiting load distribution completely.

In order to determine the distribution function $F(\tau; M, \alpha)$ in the limit $M \rightarrow \infty$ for $\tau \geq 1$ and $\alpha > 0.5$ it is necessary to deal with the fact that the component containing a fixed node v_o is often not a tree. However, when M is large and α and an integer k are fixed, the subgraph of the multigraph induced by locations at distance k or less from v_o in the multigraph is, with high probability, a tree. Thus, while the component of the multigraph containing v_o may not be a tree, it is *locally* a tree with high probability. Moreover, the number of neighbors of a node in the tree is one plus an asymptotically Poisson distributed number. To get an upper bound on the load at node v_o we suppose that no node at distance k from v_o can be assigned any load. This restriction decouples the load at v from the part of the network at distance more than k from v_o . In the limit as M tends to infinity, the distribution of the load at v_o converges to the load at the root of a Poisson tree network with the same restriction on nodes at distance k from the root. This argument is made precise and the result summarized in Theorem 8. Theorem 9 provides a means for numerically evaluating the distribution of the load at the root of a Poisson tree when nodes at distance k from the root cannot be used, and Theorem 10 describes such distribution in the limit as $k \rightarrow \infty$. Comparison of numerical results and simulation suggest that a minor adjustment of the load distribution for a Poisson tree yields the exact limiting distribution for random networks.

B. The Load Distribution Function for $\tau < 1$

The tree method works especially well for finding $F(\tau; M, \alpha)$ in case $0 \leq \tau < 1$, as shown by the proof of the following theorem.

Theorem 7:

a) Suppose $0 \leq \tau < 1$. Then

$$F(\tau; M, \alpha) = \sum_{n=1}^{\lfloor 1/(1-\tau) \rfloor} b(n, M, \alpha)$$

where

$$b(n, M, \alpha) = \binom{M-1}{n-1} n^{n-2} \cdot \left[\binom{\alpha M}{n-1} q^{n-1} (1-q)^{\alpha M - n + 1} \right] \cdot \frac{(n-1)!}{H^{n-1}},$$

$$H = \binom{n}{2} + n(M-n), \text{ and } q = H / \binom{M}{2}. \quad (19)$$

b) Suppose $0 \leq \tau < 1$. The limit $\lim_{M \rightarrow \infty} F(\tau; M, \alpha)$ exists. Call the limit $F(\tau; \alpha)$. Then

$$F(\tau; \alpha) = \sum_{n=1}^{\lfloor 1/(1-\tau) \rfloor} b(n, \alpha)$$

where

$$b(n, \alpha) = \frac{(2\alpha n)^{n-1} \exp(-2\alpha n)}{n!}.$$

- c) Let $F(1-; \alpha)$ denote the limit of $F(\tau; \alpha)$ as τ increases to one. Then $F(1-; \alpha)$ is the smallest root of the equation $p = \exp(-2\alpha(1-p))$. In particular, $F(1-; \alpha) = 1$ if and only if $\alpha \leq 0.5$.

Proof: Consider a fixed resource location v_o , and the connected component of the random network (U, V, N) that contains v_o . If the component contains a cycle (this includes the case that two locations in the component have more than one edge between them) then it is easy to see that the load at location v_o when the network is balanced is at least one. Otherwise, the component containing v_o is a simple tree, so that for some $n \geq 1$ it contains n locations, there are $n-1$ consumers constrained to use resources from among the n locations, and no other consumer can use resources from the n locations. The load can be evenly distributed among the n locations so the load at location v_o is $(n-1)/n$. We can thus complete the proof of a) by showing that $b(n, M, \alpha)$ is the probability that the component containing v_o is a simple tree containing exactly n locations. To show that, note that $\binom{M-1}{n-1}$ is the number of ways to choose $n-1$ other locations to be in the component. Since n^{n-2} is the number of possible trees based on n distinct nodes ("Cayley's formula" [6], [10]), there are n^{n-2} ways to choose $n-1$ location pairs to make v_o and the other $n-1$ locations connected. The quantity in square brackets in (19) is the probability that exactly $n-1$ consumers are allowed to use at least one of the n locations to be in the component, and $(n-1)!/H^{n-1}$ is the probability that these $n-1$ consumers are constrained to use the $n-1$ specific pairs of nodes. This concludes our explanation that $b(n, M, \alpha)$ is the probability that the component containing location v_o is a tree with n locations, and part a) is proved.

We write $A_M \asymp B_M$ if $\lim_{M \rightarrow \infty} (A_M/B_M) = 1$. We then have

$$\binom{M-1}{n-1} \asymp \frac{M^{n-1}}{(n-1)!}, \quad \binom{\alpha M}{n-1} \asymp \frac{(\alpha M)^{n-1}}{(n-1)!},$$

$$H^{n-1} \asymp (nM)^{n-1}, \quad q^{n-1} \asymp \left(\frac{2n}{M}\right)^{n-1},$$

$$(1-q)^{\alpha M - n + 1} \asymp \exp(-2\alpha n),$$

which imply that $\lim_{M \rightarrow \infty} b(n, M, \alpha) = b(n, \alpha)$ for each n , which implies part b) of the theorem.

Clearly by part b) of the theorem, $F(1-; \alpha) = \sum_{n=1}^{\infty} b(n, \alpha)$. That this sum is the smallest positive root of the equation $p = \exp(-2\alpha(1-p))$ is a consequence of Lagrange's inversion formula [6, p. 23]. \square

C. Embedded Poisson Trees

1) *Rooted Labeled Trees and Degree Sequences:* Some notation will be introduced to help us describe the Poisson tree nature of the random multigraph near a fixed location. We define a rooted labeled tree (RLT) to be a simple, directed graph that is a rooted tree with edges directed away from the root node, such that the successors of any node are labeled from one up to the number of successors. See Fig. 3 for an example. The graph may have an infinite number of nodes. All nodes are labeled except for the root node. Under the provision that the successors of any node are to be ordered according to increasing labels, breadth-first search provides a unique ordering of the nodes in an RLT. We define the degree sequence of an RLT to be the sequence (d_1, d_2, \dots) where d_i for $i \geq 1$ is the outdegree of the i th node in the breadth-first ordering. For example, the degree sequence of the RLT in Fig. 3 is $(2, 2, 3, 0, 2, 0, 0, 0, 1, 0)$. The number L_k of nodes in an RLT that are at most distance k from the root is determined by the degree sequence as follows.

$$L_0 = 1, \quad L_1 = 1 + d_1,$$

$$L_i = L_{i-1} + \sum_{j: L_{i-2} < j \leq L_{i-1}} d_j, \quad \text{for } i \geq 2. \quad (20)$$

Given an RLT and a positive integer k , we define the k th partial degree sequence to be $(d_1, d_2, \dots, d_{L_{k-1}})$. For example, the third partial degree sequence of the tree in Fig. 3 is $(2, 2, 3, 0, 2, 0, 0)$. In general, the k th partial degree sequence determines (L_0, L_1, \dots, L_k) .

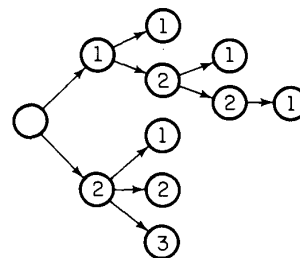


Fig. 3. Example of rooted labeled tree.

Given a sequence of nonnegative integers, (d_1, d_2, \dots) , we will construct an RLT satisfying the following: the degree sequence of the RLT is equal to either (d_1, d_2, \dots) or to a finite prefix of it. In the former case the RLT has infinitely many nodes. The nodes are a subset of \mathbb{Z}_+ with 0 serving as the root node. The set of nodes at distance i from the root is $\{v: L_{i-1} < v \leq L_i\}$ where $(L_i; i \geq 0)$ is defined by (20). If j is in level i for some i then j has d_j successors, $\{v: d_1 + \dots + d_{j-1} < v \leq d_1 + \dots + d_j\}$, and these successors are labeled from 1 to d_j in increasing order. The construction is complete.

Let \mathcal{D}_k denote the set of sequences

$$\mathcal{D}_k = \{(d_1, \dots, d_{L_{k-1}}(d_1, d_2, \dots)); d_i \in \mathbb{Z}_+\}$$

where $L_{k-1}(d_1, d_2, \dots)$ is defined by (20). Define the radius of an RLT to be the maximum of the distances between the root node and other nodes. There is a one-to-one correspondence between \mathcal{D}_k and the set of k th partial degree sequences of RLT's of radius at most k .

2) *Embedded Trees in the Random Network:* We return our attention to the random network (U, V, N) with parameters M and αM , and we continue to view it as a multigraph with node set V . Let v_o be a fixed location and let $k \geq 1$. We will construct a random variable D with values in $\{\delta\} \cup \mathcal{D}_k$ where δ is a "null" value. If there is a cycle of length less than or equal to $2k + 1$ that contains node v_o , then set $D = \delta$. Otherwise, consider the set \mathcal{Y}_o of locations at distance at most $k - 1$ from v_o , including v_o itself, and the set \mathcal{Y} of locations at distance at most k from v_o . For each $v \in \mathcal{Y}_o$, let $H(v)$ denote the set of locations that neighbor v and that are farther from v_o than is v , and randomly label the nodes in the set $H(v)$ from one to $|H(v)|$, all $|H(v)|!$ permutations being equally likely. So labeled, the subgraph induced by \mathcal{Y} is an RLT. Set D equal to the k th partial degree sequence of the tree.

Define a probability distribution $P_{\lambda, k}$ on \mathcal{D}_k by

$$P_{\lambda, k}(d) = \prod_{i=1}^{m_o} \frac{\lambda^{d_i} \exp(-\lambda)}{d_i!}, \quad (21)$$

for $d = (d_1, \dots, d_{m_o}) \in \mathcal{D}_k$.

Lemma 2: Fix k and α and let $d = (d_1, \dots, d_{m_o}) \in \mathcal{D}_k$. Then $\lim_{M \rightarrow \infty} P[D = d] = P_{2\alpha, k}(d)$. Also, $\lim_{M \rightarrow \infty} P[D = \delta] = 0$.

Proof: Let $m = d_1 + \dots + d_{m_o}$, and define Γ and p by

$$\Gamma = \binom{m+1}{2} + m_o(M - (m+1)), \quad p = \frac{\Gamma}{\binom{M}{2}}.$$

Then

$$P[D = d] = (M-1)_m \cdot \left(\frac{\alpha M}{m}\right) p^m (1-p)^{\alpha M - m} \left(\frac{m!}{\Gamma^m}\right) \prod_{i=1}^{m_o} \frac{1}{d_i!}. \quad (22)$$

To verify (22), note that $D = d$ if and only if \mathcal{Y} contains exactly $m + 1$ locations including v_o , and certain other conditions hold. There are thus $(M-1)_m$ ways to choose the locations that will be in \mathcal{Y} and to number the locations in $\mathcal{Y} - \{v_o\}$ from 2 to $m + 1$. The numbering will correspond to the order of the nodes in breadth-first search, so that \mathcal{Y}_o must consist of v_o and those nodes numbered 2 through m_o . Note that $\Gamma = |\mathcal{U}|$, where \mathcal{U} is the set of pairs of locations that contain two locations in \mathcal{Y} or at least one location in \mathcal{Y}_o , and p is the probability that $N(u) \in \mathcal{U}$ for a fixed consumer u . There are now three conditions that together are equivalent to $D = d$ and the condition that the ordering of nodes in \mathcal{Y} corresponds to breadth-first search: 1) there are exactly m consumers

with $N(u) \in \mathcal{U}$, 2) the m consumers with $N(u) \in \mathcal{U}$ are the m pairs of locations uniquely determined by the numbering of the nodes in \mathcal{Y} and the partial degree sequence d , and 3) the labels are chosen to be those that are also uniquely determined by the numbering of the nodes in \mathcal{Y} and the partial degree sequence d . Equation (22) can now be seen to be true. The fact that $\lim_{M \rightarrow \infty} P[D = d] = P_{2\alpha, k}(d)$ can be readily deduced from (22) by using the same type of asymptotic equivalences used near the end of the proof of Theorem 7. The last assertion of the lemma is a consequence of the rest of the lemma and the fact that $P_{2\alpha, k}$ is a probability distribution. \square

3) *Poisson Tree Networks:* We define a Poisson tree network (U, V, N) with parameters λ and k to be a random RLT network with radius at most k , such that the distribution of the k th partial degree sequence is $P_{\lambda, k}$, defined by (21). One way to construct a Poisson tree network is to begin with a sequence of independent random variables such that each has the Poisson distribution with mean λ , and then use the construction at the beginning of this subsection to construct an RLT whose k th partial degree sequence is a prefix of the sequence of Poisson random variables.

For later use, we will describe another construction of a Poisson tree network. The construction works by induction. For $k = 0$ the network consists of a single node and no consumers. Let $k \geq 0$. To construct a Poisson tree network with parameters λ and $k + 1$, begin with a sequence $(U_i, V_i, N_i)_{i \geq 1}$ of independent Poisson tree networks with parameters λ and k , and let v_i be the root node of (U_i, V_i, N_i) . Suppose that v_o is a location and that $\{v_o\}, V_1, V_2, \dots$ are mutually disjoint sets of locations. Let u_1, u_2, \dots be consumers and suppose that $\{u_1, u_2, \dots\}, U_1, U_2, \dots$ are mutually disjoint. Let X be a Poisson random variable with mean λ that is independent of the sequence $(U_i, V_i, N_i)_{i \geq 1}$. Set

$$V = \{v_o\} \cup V_1 \cup \dots \cup V_X,$$

$$U = \{u_1, u_2, \dots, u_X\} \cup U_1 \cup \dots \cup U_X,$$

and for $u \in U$ set $N(u) = N_i(u)$ if $u \in U_i$, and $N(u) = \{v_o, v_i\}$, if $u = u_i$. See Fig. 4. It is then easy to check by

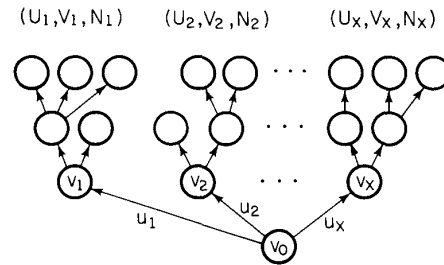


Fig. 4. Illustration of construction of Poisson tree with parameters $k + 1$ and λ from such trees with parameters k and λ .

induction that (U, V, N) is a Poisson tree network with parameters λ and $k+1$.

Remark: A Poisson tree network with parameters k and λ corresponds to the first k generations of a Galton–Watson branching process (see [7]) in which the number of children born to an individual has the Poisson distribution with mean λ . Fix n and consider the probability that a Poisson tree with parameters λ and k contains exactly n nodes. As long as $k \geq n$ it is easy to see that the probability is the same as the probability that there is a total of n individuals (summed over all generations) in a Galton–Watson branching process. Combining the fact $\lim_{M \rightarrow \infty} b(n, M, \alpha) = b(n, \alpha)$ (see the proof of Theorem 7) and Lemma 2 shows that this probability is equal to $b(n, \alpha)$, with $\lambda = 2\alpha$. Thus, we have proved the known fact that $(b(n, \alpha); n \geq 0)$, a special case of the Borel–Tanner distribution, is the distribution of the total number of individuals in a branching process with Poisson-distributed numbers of offspring per individual [8].

4) *The Poisson Tree Bound:* Define $\rho(d, k)$ for $k \geq 1$ and $d \in \mathcal{D}_k$ as follows. Consider an RLT of radius less than or equal to k and with k th partial degree sequence d . View the RLT as a consumer-location network (U, V, N) where the nodes of the RLT are resource locations and the edges are consumers with demand $m_u = 1$ for all consumers u . Suppose the base load at all nodes is zero. Finally, modify the network by requiring that no load be placed on nodes at distance k from the root. Equivalently, set $C_{u,v} = 0$ if v is distance k from the root. Thus, any edge that connects a node at distance $k-1$ from the root to a node at distance k from the root must assign all of its one unit of load to the node at distance $k-1$ from the root. Set $\rho(d, k)$ equal to the load at v_o under a UMB assignment.

Consider now the random variable D constructed for a random network in Section IV-C-2. If $D = d$, then the load at the fixed node v_o is clearly less than or equal to $\rho(d, k)$. Thus

$$F(\tau; M, \alpha) \geq \sum_{d \in \mathcal{D}_k} P[D = d] I_{\{\rho(d, k) \leq \tau\}}.$$

By Lemma 2, this implies that

$$\liminf_{M \rightarrow \infty} F(\tau; M, \alpha) \geq \sum_{d \in \mathcal{D}_k} P_{2\alpha, k}(d) I_{\{\rho(d, k) \leq \tau\}}. \quad (23)$$

Let $F_T(\tau; k, \lambda)$ denote the probability distribution function of the load at the root of a Poisson tree network with parameters $\lambda = 2\alpha$ and k for a UMB assignment when $m_u = 1$ for all consumers u , the base load is zero, and consumers are constrained not to use resource locations at distance k from the root.

Theorem 8: Define $F(\tau; \alpha)$ by

$$F(\tau; \alpha) = \liminf_{M \rightarrow \infty} F(\tau; M, \alpha).$$

Then

$$F(\tau; \alpha) \geq F_T(\tau; k, 2\alpha),$$

for $k \geq 1$.

Proof: By our definition of Poisson tree, the right-hand side of (23) is $F_T(\tau; 2\alpha, k)$, so (23) implies the theorem. \square

D. Computing the Load Distribution for Poisson Tree Networks

1) *The τ -Deficit of a Resource Location:* Motivated by Theorem 8, we will examine the probability distribution function, $F_T(\tau; k, \lambda)$, of the load at the root location of a Poisson random tree network. A key role is played by the τ -deficit for a location in a consumer-demand network (U, V, N) with demand m and base load b , which is defined as follows. Given $\tau \in \mathbb{R}$ and a resource location $v_o \in V$, define the τ -deficit of v_o to be the unique value y so that if the base load at v_o, b_{v_o} , is changed to $b_{v_o} + y$, then the load at v_o for a UMB assignment is τ . The load at v_o under a UMB assignment for the original base load vector b is less than (resp. greater than, equal to) τ if and only if the τ -deficit of v_o is greater than (resp. less than, equal to) zero. Hence, by calculating the τ -deficit of v_o for all τ , we can determine the load at v_o for a UMB assignment. The following lemma is useful for calculating the τ -deficit of nodes in a network with a tree structure, such as a Poisson tree network. As elsewhere in this section, we will assume that $|N(u)| = 2$ for all $u \in U$ and that $C_{u,v} = +\infty$ if $v \in N(u)$ and $C_{u,v} = 0$, otherwise, but we state the lemma for a general demand vector m and base load vector b for clarity.

Lemma 3: Suppose that $N(u_i) = \{v_o, v_i\}$ where v_1, v_2, \dots, v_p are distinct. Furthermore, suppose $\{\{v_o\}, V_1, V_2, \dots, V_p\}$ is a partition of V and $\{\{u_1, \dots, u_p\}, U_1, U_2, \dots, U_p\}$ is a partition of U such that $v_i \in V_i$ for all i , and $N(u) \subset V_i$ for all $u \in U_i$ and all i . Let $\tau \in \mathbb{R}$. Let y denote the τ -deficit of v_o relative to (U, V) , and for $1 \leq i \leq p$ let y_i denote the τ -deficit of the node v_i relative to the subnetwork (U_i, V_i) . For brevity, let $m_i = m_{u_i}$. Then

$$y = \tau - b_{v_o} - \sum_{i=1}^p [m_i - y_i]_0^{m_i} \quad (24)$$

where we use the notation $[x]_a^b$ for the number in $[a, b]$ closest to x .

Proof: We shall construct an assignment vector f for (U, V) . To begin, we require that for each i , the restriction of f to $U_i \times V_i$ be a uniformly most balanced assignment relative to $(U_i, V_i, (N(u): u \in U_i))$ for demand $(m_u: u \in U_i)$ and base load $(b_v: v \in V_i)$. We set $f_{u_i, v_i} = [y_i]_0^{m_i}$ and $f_{u_i, v_o} = m_i - [y_i]_0^{m_i}$ for $1 \leq i \leq p$. Finally, we set $f_{u, v} = 0$ if $v \notin N(u)$. It is easy to check that f is an admissible assignment vector. We claim that f is a UMB assignment for (U, V, N) , demand m and base load \bar{b} defined by $\bar{b}_{v_o} = b_{v_o} + y$ and $\bar{b}_v = b_v$ for $v \neq v_o$. To prove this claim it suffices to check that condition a) of Theorem 2 is satisfied, and since it is straightforward we leave it to the reader. Finally, we see that the load at node v_o for assignment f and base load \bar{b} , $b_{v_o} + y + \sum_{i=1}^p f_{u_i, v_o}$, is equal to τ . The proof of the lemma is complete. \square

2) *The τ -Deficit and Load Distribution in Poisson Trees:* Combining Lemma 3 and the recursive construction of Poisson tree networks given in Section IV-C-3 allows us to compute the probability distribution function $F_T(\tau; k, \lambda)$ of the load at the root location for a Poisson tree network with parameters k and λ , base load zero, and the constraint that nodes at distance k from the root cannot be assigned any load. First, if $k=1$, the τ -deficit of the root location, which is the only location in the network that can bear load, is equal to τ minus the degree of the root location. Now suppose $k \geq 1$. Let $(U_i, V_i, N_i)_{i \geq 1}$, $(v_i)_{i \geq 1}$, $(u_i)_{i \geq 1}$, v_o and X be as in the construction, in Section IV-C-3, of a Poisson tree network (U, V, N) with parameters λ and $k+1$. Let $Y_i^{(k)}$ denote the τ -deficit of node v_i relative to (U_i, V_i, N_i) for each $i \geq 1$, and let $Y^{(k+1)}$ denote the τ -deficit of v_o relative to (U, V, N) . Then by Lemma 3,

$$Y^{(k+1)} = \tau - \sum_{i=1}^X [1 - Y_i^{(k)}]_0^1 \quad (25)$$

Since the load at a node under a UMB assignment is less than (resp. greater than, equal to) τ if and only if the τ -deficit of the node is greater than (resp. less than, equal to) zero, we have the following relationships. A distribution function evaluated at $\tau -$ and at $\{\tau\}$ denotes the left limit and the jump of the distribution function at τ , respectively.

$$\begin{aligned} F_T(\tau; k, \lambda) &= P[Y^{(k)} \geq 0] \\ F_T(\{\tau\}; k, \lambda) &= P[Y^{(k)} = 0] \\ F_T(\tau -; k, \lambda) &= P[Y^{(k)} > 0]. \end{aligned} \quad (26)$$

Setting $Z^{(k)} = [1 - Y^{(k)}]_0^1$ and $Z_i^{(k)} = [1 - Y_i^{(k)}]_0^1$, we obtain from (25) that

$$Y^{(k+1)} = \tau - \sum_{i=1}^X Z_i^{(k)}. \quad (27)$$

Roughly speaking, $Z_i^{(k)}$ is how much load consumer u_i places at node v_o when the net load at v_o is brought to the value τ by imposition of base load $Y^{(k+1)}$ and balancing. Although we assumed that $k \geq 1$, we see that (27) also holds for $k=0$ if we define $Z_i^{(0)} = 0$. We can subtract each side of (27) from one and apply the projection operator $[\cdot]_0^1$ to express $Z^{(k+1)}$ in terms of X and $Z_i^{(k)}$, $1 \leq i \leq X$. Reexpressing the left-hand sides in (26) in terms of the Z 's then yields the following theorem. For random variables A and B we write $A \sim B$ to denote that A and B have the same distribution, and we write $A \sim \text{Poisson}(\lambda)$ to denote that A is a Poisson random variable with mean λ .

Theorem 9: Given nonnegative constants λ and τ , let $(Z^{(k)})_{k \geq 0}$ denote a sequence of random variables with probability distributions uniquely determined by the re-

quirement that, for all $k \geq 0$,

$$Z^{(0)} = 1 \quad (28)$$

$$Z^{(k+1)} = \left[\left(\sum_{i=1}^{X^{(k+1)}} Z_i^{(k)} \right) - \tau + 1 \right]_0^1 \quad (29)$$

$X^{(k+1)}, Z_1^{(k)}, Z_2^{(k)}, \dots$, are mutually independent (30)

$$Z_i^{(k)} \sim Z^{(k)}, \quad \text{for } i \geq 1$$

$$\text{and } X^{(k+1)} \sim \text{Poisson}(\lambda), \quad \text{for } k \geq 0. \quad (31)$$

Then

$$F_T(\tau; k+1, \lambda) = P \left[\sum_{i=1}^{X^{(k+1)}} Z_i^{(k)} \leq \tau \right], \quad (32)$$

$$F_T(\{\tau\}; k+1, \lambda) = P \left[\sum_{i=1}^{X^{(k+1)}} Z_i^{(k)} = \tau \right], \quad (33)$$

$$F_T(\tau -; k, \lambda) = P[Z^{(k+1)} < 1]. \quad (34)$$

If τ can be expressed as a fraction with denominator n for some positive integer n , then by induction on k we see that $Z^{(k)} \in \{0, 1/n, \dots, 1\}$ with probability one. In that case the theorem provides a convenient way to numerically compute $F_T(\tau; k+1, \lambda)$ and $F_T(\{\tau\}; k+1, \lambda)$ to any specified degree of accuracy.

3) *Asymptotic Analysis of the Load Distribution in Poisson Trees:* Since Theorem 8 holds for any k and the lower bound $F_T(\tau; k, 2\alpha)$ is nondecreasing with k , we obtain our best bound by taking the limit as $k \rightarrow \infty$. We characterize the resulting limit in Theorem 10, and we then discuss a conjecture about the tightness of the bound. Trivially, $Z^{(1)}$ is stochastically smaller than $Z^{(0)}$. Since the right-hand side of (29) is a nondecreasing function of $Z_i^{(k)}$ for each i , we have by induction that the sequence $Z^{(k)}$ is stochastically decreasing in k . Hence the sequence converges in distribution to a random variable Z . Since the right-hand side of (29) is a continuous function of $Z_i^{(k)}$ for each k , we have that Z is the stochastically largest solution to the set of equations:

$$Z = \left[\left(\sum_{i=1}^X Z_i \right) - \tau + 1 \right]_0^1 \quad (35)$$

$$X, Z_1, Z_2, \dots, \text{ are mutually independent} \quad (36)$$

$$Z_i \sim Z \quad \text{for } i \geq 1 \text{ and } X \sim \text{Poisson}(\lambda). \quad (37)$$

We summarize some consequences in a theorem.

Theorem 10: a) The limits $\lim_{k \rightarrow \infty} F_T(\tau; k, \lambda)$ and $\lim_{k \rightarrow \infty} F_T(\{\tau\}; k, \lambda)$ exist. Call them $F_T(\tau; \lambda)$ and $a_T(\tau; \lambda)$, respectively. b) There is a stochastically maximal solution Z to (35)–(37). If τ is a rational number and X, Z_1, Z_2, \dots , correspond to Z as in (35)–(37), then

$$F_T(\tau; \lambda) = P \left[\sum_{i=1}^X Z_i \leq \tau \right] \quad (38)$$

$$a_T(\tau; \lambda) = P \left[\sum_{i=1}^X Z_i = \tau \right]. \quad (39)$$

Proof: Part a) of the theorem is a consequence of the fact that

$$F_T(\{\tau\}; k, \lambda) = F_T(\tau; k, \lambda) - F_T(\tau - ; k, \lambda)$$

and both $F_T(\tau; k, \lambda)$ and $F_T(\tau - ; k, \lambda)$ are nondecreasing in k , and the first statement in part b) was already proved. Finally, if $n\tau$ is an integer for an integer n , then $Z^{(k)}$, and also Z , is distributed on $\{0, 1/n, \dots, 1\}$ for each k . Equations (38) and (39) follow from (32) and (33). \square

We can immediately provide a bound for the limit infimum, $\underline{F}(\tau; \alpha)$, of the load distribution function for large random networks.

Corollary 8: a) For $\tau, \alpha \geq 0$, $\underline{F}(\tau; \alpha) \geq F_T(\tau; 2\alpha)$.

b) Let $\Delta(\alpha)$ be the minimum solution to the equation

$$\alpha = \int_0^{\Delta(\alpha)} 1 - F_T(\tau; 2\alpha) d\tau$$

if such a solution exists, and $\Delta(\alpha) = +\infty$ otherwise. Then $\underline{F}(\tau; \alpha) < 1$ for $\tau < \Delta(\alpha)$.

Proof: Part a) is immediate from Theorem 8 and a) of Theorem 10. Since the mean load at a resource location in the random network is α , $\alpha = \int_0^\infty 1 - F(\tau; \alpha, M) d\tau$ so by Fatou's lemma,

$$\alpha = \limsup_{M \rightarrow \infty} \int_0^\infty 1 - F(\tau; \alpha, M) d\tau \leq \int_0^\infty 1 - \underline{F}(\tau, \alpha) d\tau. \tag{40}$$

This inequality and part a) imply part b). \square

Some numerical calculations are indicated in Table I. These were made by using Theorem 9 and then numerically estimating the limit as $k \rightarrow \infty$. No numerical instabilities were encountered. We let τ range over low multiples of $\frac{1}{12}$. The last column of the table indicates the upper and lower bounds on $\int_0^1 1 - F_T(u; 2\alpha) du$ that are implied by the first two columns and the fact that $F_T(\tau; 2\alpha)$ is nondecreasing in τ and has a jump of magnitude at least $a(\tau; 2\alpha)$ at τ . These upper and lower bounds are not equal since $F_T(\tau; 2\alpha)$ increases in between the values of τ listed in the table, and the bounds can be made arbitrarily close by considering more values of τ . The rows in Table I for $\tau = 2.1667$ and $\tau = 2.5000$ imply that

$$2.1667 + \frac{2 - 1.97575}{1 - 0.34945} \leq \Delta(2) \leq 2.1667 + \frac{2 - 1.97449}{1 - 0.38899 + 0.02083}$$

so that $2.203 < \Delta(2) < 2.208$. Bounds on $\Delta(\alpha)$ computed in this fashion (but letting τ range over multiples of numbers smaller than $\frac{1}{12}$) are reported in Table II.

Note that $F_T(\tau; 2\alpha)$, as a function of τ , cannot equal $\lim_{M \rightarrow \infty} F(\tau; M, \alpha)$ since the integral indicated in the last column of Table I is larger than α for τ large. However, let $x = (x(1), \dots, x(M))$ denote the load vector for a uniformly most balanced assignment, let $x_{\max} = \max\{x(v) : v \in V\}$ and let $\hat{A} = \{v \in V : x(v) = x_{\max}\}$. We will prove in the next section that, for α not too small and M very

TABLE I
NUMERICAL CALCULATIONS BASED ON THEOREMS 9 AND 10 FOR $\alpha = 2$

τ	$a(\tau; 2\alpha)$	$F_T(\tau; 2\alpha)$	$\int_0^1 1 - F_T(u; 2\alpha) du$
0	0.01832	0.01832	0
0.0833	0.00000	0.01832	0.08181 - 0.08181
0.1667	0.00000	0.01832	0.16361 - 0.16361
0.2500	0.00000	0.01832	0.24542 - 0.24542
0.3333	0.00000	0.01832	0.32723 - 0.32723
0.4167	0.00000	0.01832	0.40904 - 0.40904
0.5000	0.00134	0.01966	0.49084 - 0.49084
0.5833	0.00000	0.01966	0.57254 - 0.57254
0.6667	0.00015	0.01980	0.65423 - 0.65423
0.7500	0.00002	0.01982	0.73592 - 0.73592
0.8333	0.00000	0.01983	0.81760 - 0.81760
0.9167	0.00000	0.01983	0.89928 - 0.89928
1.0000	0.07774	0.09756	0.98096 - 0.98096
1.0833	0.00000	0.09756	1.05616 - 1.05616
1.1667	0.00000	0.09757	1.13136 - 1.13136
1.2500	0.00026	0.09785	1.20657 - 1.20657
1.3333	0.00246	0.10031	1.28174 - 1.28175
1.4167	0.00000	0.10050	1.35670 - 1.35672
1.5000	0.02180	0.12232	1.43166 - 1.43168
1.5833	0.00000	0.12240	1.50479 - 1.50482
1.6667	0.00920	0.13236	1.57786 - 1.57795
1.7500	0.00462	0.13728	1.65014 - 1.65026
1.8333	0.00150	0.14151	1.72181 - 1.72215
1.9167	0.00012	0.14399	1.79315 - 1.79369
2.0000	0.19107	0.33533	1.86446 - 1.86502
2.0833	0.00024	0.33598	1.91982 - 1.92041
2.1667	0.00554	0.34945	1.97449 - 1.97575
2.2500	0.02083	0.38899	2.02715 - 2.02996
2.3333	0.04704	0.47013	2.07522 - 2.08088
2.4167	0.00000	1.00000	2.07522 - 2.12503
2.5000	0.00000	1.00000	2.07522 - 2.12503

TABLE II
SOME NUMERICAL VALUES OF $\Delta(\alpha)$ AND $1 - F_T(\Delta(\alpha) - ; 2\alpha)$

α	$\Delta(\alpha)$	$1 - F_T(\Delta(\alpha) - ; 2\alpha)$
0.2	1	0
0.4	1	0
0.6	1.07235 - 1.08333	0.00000 - 0.03520
0.8	1.22226 - 1.22794	0.11913 - 0.14205
1.0	1.37544 - 1.37757	0.25944 - 0.26825
1.2	1.52653 - 1.53595	0.35094 - 0.37209
1.4	1.68945 - 1.69542	0.46912 - 0.49171
1.6	1.84892 - 1.85339	0.58498 - 0.60966
1.8	2.00525 - 2.00764	0.50888 - 0.50975
2.0	2.20438 - 2.20593	0.63826 - 0.65055
4.0	4.07933 - 4.07937	0.87073 - 0.87074
6.0	6.03571 - 6.03575	0.94613 - 0.94615
8.0	8.01720 - 8.01721	0.97578 - 0.97578
10.0	10.00857 - 10.00857	0.98855 - 0.98855

large, the set \hat{A} tends to be large. This suggests that, rather than requiring that the nodes at the boundaries of the Poisson tree networks carry no load, we should allow a consumer to use such a node if the other node the consumer has available has load at least as large as the limiting value of x_{\max} . Figuratively speaking, we view the nodes at the boundaries of the tree networks as infinite sinks with load clamped at the limiting value of x_{\max} . The effect of this modification on the distribution of load at the root node is to truncate it at the limiting value of x_{\max} . The value of the truncation point in the limit of large trees must be $\Delta(\alpha)$ in order that the mean load at the root be α . We thus pose the following conjecture. The second part of the conjecture is motivated by the fact that

the function on the right-hand side of (41) has a jump of size $1 - F_T(\Delta(\alpha) - ; 2\alpha)$ at $\tau = \Delta(\alpha)$.

Conjecture: For all $\tau, \alpha \geq 0$,

$$\lim_{M \rightarrow \infty} F(\tau; M, \alpha) = \begin{cases} F_T(\tau; 2\alpha), & \text{if } \tau < \Delta(\alpha) \\ 1, & \text{if } \tau \geq \Delta(\alpha). \end{cases} \quad (41)$$

Furthermore,

$$\lim_{M \rightarrow \infty} \text{in-prob.} \frac{|\hat{A}|}{M} = 1 - F_T(\Delta(\alpha) - ; 2\alpha).$$

The conjecture is true if either $0 \leq \tau < 1$ or $0 \leq \alpha \leq 0.5$, as can be seen with the help of Theorem 7.

V. ASYMPTOTIC ANALYSIS—THE MOMENT METHOD

Consider the random consumer-resource network (U, V, N) with parameters M, α and c defined in Section I-C. Let $x = (x(1), \dots, x(M))$ denote the load vector for a uniformly most balanced assignment, let $x_{\max} = \max\{x(v) : v \in V\}$ and let $\hat{A} = \{v \in V : x(v) = x_{\max}\}$. We will use the method of moments (in the sense of the theory of random graphs) in a fashion similar to that in [13] and [14] to bound x_{\max} above and $|\hat{A}|$ below, in a certain asymptotic sense. The key step in the moment method is given in (45) and (54).

Define, for $\alpha > 0$ and $0 \leq r \leq \min\{\alpha/\tau, 1\}$,

$$\phi(r, \alpha, \tau) = h(r) + \alpha h\left(\frac{r\tau}{\alpha}\right) + r\tau \log r^c + (\alpha - r\tau) \log(1 - r^c) \quad (42)$$

where \log 's are taken base e and h is defined by $h(u) = -u \log u - (1-u) \log(1-u)$. Also, let

$$\bar{\tau}(\alpha) = \inf \left\{ \tau \geq \alpha : \phi(r, \alpha, \tau) < 0, \quad \text{for } 0 < r \leq \frac{\alpha}{\tau} \right\}$$

and

$$\sigma(\alpha) = \inf \{r > 0 : \phi(r, \alpha, \alpha) \geq 0\}.$$

These functions are well defined and can be easily numerically computed to within specified accuracy. Consideration of $\phi(r, \alpha, \tau)$ for r near zero shows that $\bar{\tau}(\alpha) > 1/(c-1)$ for all $\alpha > 0$, and also that $\sigma(\alpha) \geq 1$, with equality if and only if $0 \leq \alpha \leq 1$. Some values are displayed in Table III.

Theorem 11:

- For $\alpha > 0$ and τ with $\tau > \bar{\tau}(\alpha)$,
 $\lim_{M \rightarrow \infty} P[x_{\max} < \tau] = 1$.
- For $\alpha > 1/(c-1)$ and $0 \leq \sigma < \sigma(\alpha)$,
 $\lim_{M \rightarrow \infty} P[|\hat{A}| \geq \sigma M] = 1$.
- $\lim_{\alpha \rightarrow \infty} \bar{\tau}(\alpha) - \alpha = 0$.
- $\lim_{\alpha \rightarrow \infty} \sigma(\alpha) = 1$.

Proof: The function $\phi(r, \alpha, \tau)$ is decreasing in τ for $\alpha \leq \tau \leq \alpha/r$, in fact for $\alpha r^{c-1} \leq \tau \leq \alpha/r$, because it is concave in τ and $\partial \phi(r, \alpha, \tau) / \partial \tau = 0$ for $\tau = \alpha r^{c-1}$. Thus,

TABLE III
VALUES OF $\bar{\tau}(\alpha)$ AND $\sigma(\alpha)$ FOR SOME VALUES OF α , AND $c = 2$

α	$\bar{\tau}(\alpha)$	$\sigma(\alpha)$
0.20	1.0577	0
0.40	1.2393	0
0.60	1.4119	0
0.80	1.5804	0
1.00	1.7472	0
1.20	1.9138	0.0000
1.40	2.0806	0.0025
1.60	2.2480	0.0140
1.80	2.4164	0.0346
2.00	2.5859	0.0614
4.00	4.3431	0.3870
6.00	6.1955	0.6326
8.00	8.1095	0.7878
10.00	10.0607	0.8805

if $\tau > \bar{\tau}(\alpha)$, as we now assume, then

$$\phi(r, \alpha, \tau) < 0, \quad \text{for } 0 < r \leq \frac{\alpha}{\tau}. \quad (43)$$

Define the random variable $W_{\tau, j}$ by

$$W_{\tau, j} = \left\{ A \subset V : |A| = j, \frac{\gamma(A)}{|A|} \geq \tau \right\} \quad (44)$$

and let $W_\tau = \sum_{j=c}^M W_{\tau, j}$. By Corollary 7, $x_{\max} \geq \tau$ if and only if $W_\tau \geq 1$. Consequently,

$$P[x_{\max} \geq \tau] = P[W_\tau \geq 1] \leq E[W_\tau]. \quad (45)$$

Thus, part a) of the theorem will be established once we prove that $\lim_{M \rightarrow \infty} E[W_\tau] = 0$. Introduce the following notation: $B(n, p)$ denotes a binomial random variable with parameters n and p , and n_c denotes $n(n-1)\cdots(n-c+1)$. We will verify that

$$\begin{aligned} E[W_{\tau, j}] &= \binom{M}{j} P \left[B \left(\alpha M, \frac{j_c}{M_c} \right) \geq j\tau \right] \\ &\leq \binom{M}{j} P \left[B \left(\alpha M, \left(\frac{j}{M} \right)^c \right) \geq j\tau \right] \end{aligned} \quad (46)$$

$$\leq \binom{M}{j} \binom{\alpha M}{\lceil j\tau \rceil} \left(\frac{j}{M} \right)^{\lceil j\tau \rceil c} \quad (47)$$

$$\leq \frac{M^j (\alpha M)^{\lceil j\tau \rceil}}{j! \lceil j\tau \rceil!} \left(\frac{j}{M} \right)^{\lceil j\tau \rceil c} \quad (48)$$

$$\leq \left(\frac{j}{M} \right)^{(c-1)\lceil j\tau \rceil - j} \left(\frac{\alpha}{\tau} \right)^{\lceil j\tau \rceil} e^{j + \lceil j\tau \rceil} \quad (49)$$

$$\leq e \left(\left(\frac{j}{M} \right)^{(c-1)\tau - 1} e^{\tau + 1} \right)^j. \quad (50)$$

The inequality in (46) follows from the fact that $j_c/M_c \leq (j/M)^c$. The inequality in (47) uses the fact that $P[B(n, p) \geq k] \leq p^k \binom{n}{k}$, which is a bound based on the union bound. The inequality in (49) is deduced using the fact $n! \geq (n/e)^n$.

Set $\delta = \epsilon^{(c-1)\tau-1}e^{\tau+1}$ where ϵ is a positive number so small that $\delta < 1$. Then by the inequalities just proved,

$$\sum_{j=c}^{\lfloor \epsilon M \rfloor} E[W_{\tau,j}] \leq e \sum_{j=c}^{\infty} \delta^j = \frac{e\delta^c}{1-\delta} \quad (51)$$

for all M , and the left-hand side of (51) can be made arbitrarily small by choosing ϵ small. Next we see that

$$\begin{aligned} & \sum_{j=\lfloor \epsilon M \rfloor}^M \binom{M}{j} P \left[B \left(\alpha M, \left(\frac{j}{M} \right)^c \right) \geq j\tau \right] \\ & \leq M \max_{\lfloor \epsilon M \rfloor \leq j \leq \lfloor \alpha M / \tau \rfloor} \binom{M}{j} P \left[B \left(\alpha M, \left(\frac{j}{M} \right)^c \right) \geq j\tau \right] \\ & \leq \alpha M^2 \max_{\epsilon \leq r \leq \alpha/\tau} \exp(M\phi(r, \alpha, \tau)) \quad (52) \\ & = \alpha M^2 \exp \left(M \max_{\epsilon \leq r \leq \alpha/\tau} \phi(r, \alpha, \tau) \right) \xrightarrow{M \rightarrow \infty} 0. \quad (53) \end{aligned}$$

The inequality in (52) comes from the bound $\binom{n}{k} \leq \exp(nh(k/n))$, the change of variable $j = Mr$, and the following inequality, valid for $x \geq np$:

$$\begin{aligned} P[B(n, p) \geq x] & \leq nP[B(n, p) = \lceil x \rceil] \\ & \leq n \exp \left(-n \left[\frac{x}{n} \log \frac{x}{np} + \left(1 - \frac{x}{n} \right) \log \frac{n-x}{n-np} \right] \right). \end{aligned}$$

The limiting value indicated in (53) is a consequence of (43). In view of (46), (51), and (53), $\lim_{M \rightarrow \infty} E[X_\tau] = 0$ so that part a) of the theorem is proved.

We turn next to the proof of part b), so suppose $\alpha > 1/(c-1)$ and that $0 < \sigma \leq \underline{\sigma}(\alpha)$. Define the random variable $S_{M,\sigma}$ by

$$S_{M,\sigma} = \sum_{j=c}^{\lfloor \sigma M \rfloor} W_{\alpha,j}.$$

By Corollary 7, $\gamma(\hat{A}) = |\hat{A}|x_{\max} \geq |\hat{A}|\alpha$, so that if $|\hat{A}| \leq M\sigma$ then $S_{M,\sigma} \geq 1$. Thus

$$P[|\hat{A}| \leq M\sigma] \leq P[S_{M,\sigma} \geq 1] \leq E[S_{M,\sigma}] \quad (54)$$

so that to prove b) it suffices to prove that $\lim_{M \rightarrow \infty} E[S_{M,\sigma}] = 0$.

A review of the proof of (51) shows it to be true for any τ with $\tau \geq \alpha$. In particular, it is true for $\tau = \alpha$ so that for some function η with $\lim_{\epsilon \rightarrow 0} \eta(\epsilon) = 0$, $E \sum_{j=c}^{\lfloor \epsilon M \rfloor} W_{\alpha,j} \leq \eta(\epsilon)$ for all M . By the same arguments leading to (53), we have that

$$\begin{aligned} & \lim_{M \rightarrow \infty} \sum_{j=\lfloor \epsilon M \rfloor}^{\sigma M} E[W_{\alpha,\tau}] \\ & \leq \lim_{M \rightarrow \infty} M^2 \exp \left(M \max_{\epsilon \leq r \leq \sigma} \phi(r, \alpha, \tau) \right) = 0. \end{aligned}$$

Thus, $\lim_{M \rightarrow \infty} E[S_{M,\sigma}] = 0$, and part b) is proved.

We now turn to the proof of c). Let $\delta > 0$. We consider $\alpha \geq 1 + \delta$, $\tau = \alpha + \delta$ and r in the range $0 < r \leq \alpha/\tau$. Consider (42), which defines $\phi(r, \alpha, \tau)$. Applying the inequality $-(1-x)\log(1-x) \leq x$ to the first two terms and

using the fact that the last term is negative yields that

$$\phi(r, \alpha, \tau) \leq ((c-1)\tau - 1)r \log r + r \left[1 + \tau - \tau \log \frac{\tau}{\alpha} \right]. \quad (55)$$

From this we conclude that there is an $\epsilon > 0$ depending only on δ so that

$$\phi(r, \alpha, \tau) < 0, \quad \text{for } 0 < r \leq \epsilon. \quad (56)$$

Since h is a concave function,

$$\begin{aligned} \alpha h \left(\frac{r\tau}{\alpha} \right) & \leq \alpha h(r) + \alpha \left(\frac{r\tau}{\alpha} - r \right) h'(r) \\ & = \alpha h(r) + r\delta \log \frac{1-r}{r}. \end{aligned}$$

Using this inequality and the fact that $\tau = \alpha + \delta$ in (42) implies that $\phi(r, \alpha, \tau) \leq \alpha A + B$ where

$$\begin{aligned} A & = (1-r) \log \left(\frac{1-r^c}{1-r} \right) + (c-1)r \log r \\ & \leq (c-1)[(1-r) \log(1+r) + r \log r] \\ & \leq (c-1)[r(1-r) + r \log r] \leq \frac{-(c-1)(1-r)^2 r}{2} \quad (57) \end{aligned}$$

and

$$B = r\delta \log \left(\frac{1-r}{1-r^c} \right) + h(r) + (c-1)\delta r \log r.$$

The quantity B is bounded for $0 \leq r \leq 1$ and $\lim_{r \rightarrow 1} B = -\delta \log c < 0$. Combining these facts shows that for α sufficiently large, $\phi(r, \alpha, \tau) < 0$ for $\epsilon \leq r \leq \alpha/(\alpha + \delta)$. In view of (56), if α is sufficiently large then $\phi(r, \alpha, \alpha/\tau) < 0$ for $0 < r \leq \tau$ so that $\bar{\tau}(\alpha) \leq \tau$. Since δ was arbitrary, part c) is proved.

Finally, if we set $\delta = 0$ and require that $\alpha \geq 2$ (so α is still bounded away from 1) then in (56), the bound $\phi(r, \alpha, \tau) \leq \alpha A + B$ and (57) still hold. Thus there is an $\epsilon > 0$ so that $\phi(r, \alpha, \alpha) < 0$ for $\alpha \geq 2$ and $0 < r \leq \epsilon$, and $\phi(r, \alpha, \alpha) \leq \alpha A + h(r)$ for $0 \leq r \leq 1$, where A is defined as before. These facts and the bound on A in (57) imply part d) of the theorem. \square

VI. MONTE CARLO SIMULATIONS

We ran Monte Carlo simulations of a random network with the description in Section I-C. Our interest was in how balanced the load is for a UMB assignment, rather than how quickly a particular algorithm can find it. Each consumer demands one unit of resource and is constrained to obtain it from one of two (so $c = 2$) resource locations. Initially each consumer places 0.5 units of load on each of its two resource locations. Then we apply the balancing algorithm of Section III-B (which is especially simple when $c = 2$) to obtain a uniformly most balanced assignment. Data from a run with 10000 resource locations and $\alpha = 2.0$ is indicated in Tables IV and V. The tables indicate the number of locations with load less than or equal to, and the number exactly equal to the various

TABLE IV
SAMPLE LOAD DISTRIBUTION BEFORE BALANCING ($\alpha = 2, M = 10000$)

τ	Load $\leq \tau$	Load = τ
0.0	201	201
0.5	921	720
1.0	2382	1461
1.5	4299	1917
2.0	6291	1992
2.5	7896	1605
3.0	8899	1003
3.5	9472	573
4.0	9778	306
4.5	9912	134
5.0	9962	50
5.5	9987	25
6.0	10000	13

TABLE V
SAMPLE LOAD DISTRIBUTION AFTER BALANCING ($\alpha = 2, M = 10000$)

τ	Load $\leq \tau$	Load = τ	Product
0.0000000	201	201	0
0.5000000	223	22	11
1.0000000	992	769	769
1.2500000	996	4	5
1.3333333	1023	27	36
1.5000000	1239	216	324
1.6000000	1244	5	8
1.6666667	1313	69	115
1.7500000	1353	40	70
1.7777778	1362	9	16
1.8000000	1392	30	54
1.8333333	1398	6	11
1.85714286	1405	7	13
1.92307692	1418	13	25
2.0000000	3316	1898	3796
2.07692308	3329	13	27
2.11111111	3338	9	19
2.1250000	3362	24	51
2.14285714	3404	42	90
2.1666667	3440	36	78
2.18181818	3462	22	48
2.2000000	3562	100	220
2.20782852	10000	6438	14214

values. The load distribution in Table IV roughly matches that of 0.5 times a Poisson random variable with mean 2α , as expected. Comparison of Tables IV and V shows that the maximum load was reduced from 6.0 to 2.2078... by the balancing algorithm. The last column of Table V gives, for each τ , the product of τ and the number of resource locations that have load τ for a UMB assignment. These products are integer valued, as can be verified by the following simple consequence of Theorem 2: for $u \in U$ and $v, v' \in V$, $f_{u,v} \in \{0, 1\}$ whenever $x(v) \neq x(v')$.

We shall compare the data in Table V with the asymptotic bounds found in Sections IV and V. First, the fractions of nodes with load less than or equal to τ for various values of τ , computed from the second column of Table V, are generally larger than (and close to, for τ less than the maximum load) the asymptotic lower bound given in the third column of Table I. Secondly, the maximum load, 2.20782..., is less than the asymptotic upper bound, $\bar{\tau}(\alpha) = 2.5859$, given by the moment method in Section V, and it is close to the conjectured asymptotic maximum load $\Delta(\alpha)$, which is computed to be in the range 2.20438 - 2.20593 (see Table II). Thirdly, the frac-

TABLE VI
SAMPLE LOAD DISTRIBUTION AFTER BALANCING ($\alpha = 10, M = 10000$)

τ	Load $\leq \tau$	Load = τ	Product
6.00000000	2	2	12
7.00000000	6	4	28
8.00000000	17	11	88
9.00000000	51	34	306
9.33333333	54	3	28
9.50000000	56	2	19
10.00000000	114	58	580
10.00799110	10000	9886	98939

TABLE VII
SAMPLE LOAD DISTRIBUTION AFTER BALANCING ($\alpha = 0.5, M = 1000$)

τ	Load $\leq \tau$	Load = τ	Product
0.00000000	363	363	0
0.50000000	493	130	65
0.66666667	589	96	64
0.75000000	645	56	42
0.80000000	655	10	8
0.83333333	679	24	20
0.85714286	707	28	24
0.87500000	731	24	21
0.88888889	740	9	8
0.90909091	751	11	10
0.91666667	763	12	11
0.92857143	777	14	13
0.93333333	837	60	56
0.93750000	853	16	15
0.95454545	875	22	21
0.95833333	899	24	23
0.96551724	928	29	28
0.97435897	967	39	38
1.00000000	1000	33	33

tion of locations with the maximum load, about 64.4%, is (much) larger than the asymptotic lower bound $\underline{\sigma}(\alpha) = 6.14\%$ given by the moment method in Section V, and it is close to the conjectured asymptotic fraction of locations with maximum load, $1 - F_T(\Delta(\alpha) - ; 2\alpha)$, computed to be in the range 63.826-65.055% (see Table II).

Data similar to that in Table V is shown for $\alpha = 10$ and $\alpha = 0.5$ in Tables VI and VII, respectively. For the example with $\alpha = 10$ we see that the maximum load is just slightly larger than the average load α and the proportion of locations with load equal to the maximum is near one, as to be expected from part d) of Theorem 11. For all three examples we see that if the load at any resource location is less than one, then it is of the form $1 - (1/n)$ for some $n \geq 1$. By Theorem 7, when $c = 2$ the value 0.5 is a critical value of α . As M tends to infinity, the proportion of nodes with load strictly less than one should tend to one. Table VII reflects that fact.

VII. CONCLUSION

We have shown that certain balancing problems have, in a certain sense, uniformly best assignments. Algorithms were given for finding the assignments, and large random networks with uniformly most balanced loads were analyzed. Our analysis indicates that load balancing by local adjustments can be effective. It can cause a large number of resource locations to have the maximum load, where the maximum load is not much larger than the mean load.

The integral balancing problem studied in this paper as well as the existence of uniformly most balanced integral assignments were first given in [4] in the context of scheduling transmissions of packet radios. The terminology in terms of consumers and resource locations used in this paper was first given in [3], which also contains a variation of part a) of Theorem 11. The continuous assignment problem, which as we showed is closely connected to the integral assignment problem and is more amenable to analysis, is presented here for the first time.

Our discussion of algorithms (Section III) was brief. A more extensive study of algorithms would best be pursued in the context of a larger class of problems, such as the class of network flow problems with convex costs. It would be interesting to know if our performance analysis carries over to more general flow problems as well.

We considered a "static" balancing problem. There may be extensions to "dynamic" balancing problems in which consumers arrive at different times and require the use of resources for a finite amount of time.

ACKNOWLEDGMENT

The author wishes to thank Rene Cruz for many useful comments related to this paper. He is also grateful to three anonymous reviewers for valuable suggestions and to Frank Kelly, who pointed out [8].

REFERENCES

- [1] D. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. New York: Academic Press, 1982.
- [2] D. Bertsekas, and J. Tsitsiklis, *Parallel and Distributed Computation Numerical Methods*. Englewood Cliffs, NJ: Prentice Hall, 1989.
- [3] R. L. Cruz and B. Hajek, "Global load balancing by local adjustments," in *Proc. 1985 Conf. Inform. Sci. and Syst.*, Johns Hopkins Univ., Baltimore, MD, pp. 448-454, 1985.
- [4] B. Hajek, "Balanced scheduling in a packet synchronized spread spectrum network," in *Proc. IEEE Infocom*, pp. 56-65, 1983.
- [5] B. Hajek, and R. G. Ogier, "Optimal dynamic routing in communication networks with continuous traffic," *Networks*, vol. 14, pp. 457-487, 1984.
- [6] H. Harary and E. M. Palmer, *Graphical Enumeration*. New York: Academic Press, 1973.
- [7] T. E. Harris, *The Theory of Branching Processes*. New York: Springer-Verlag, 1963.
- [8] N. L. Johnson and S. Kotz, *Discrete Distributions*. Boston, MA: Houghton Mifflin, 1969.
- [9] D. G. Luenberger, *Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, 1984.
- [10] E. M. Palmer, *Graphical Evolution*. New York: Wiley, 1985.
- [11] C. H. Papadimitriou and K. Steigletz, *Combinatorial Optimization: Algorithms and Complexity*. Englewood Cliffs, NJ: Prentice Hall, 1982.
- [12] R. T. Rockafellar, *Network Flows and Monotropic Optimization*. New York: Wiley-Interscience, 1984.
- [13] E. Shamir and E. Upfal, "One factor in random graphs based on vertex choice," *Discrete Math.*, vol. 41, pp. 281-286, 1982.
- [14] D. Walkup, "Matchings in random regular bipartite digraphs," *Discrete Math.*, vol. 31, pp. 59-65, 1980.