



Dynamic Bandwidth Allocation in a Network

Kurt Maly, C. Michael Overstreet, Xia-ping Qiu†, Deqing Tang‡

Computer Science Department
Old Dominion University
Norfolk, VA 23529-0162

Abstract

Recently protocols have been introduced which enable us to integrate periodic traffic (voice or video) and aperiodic traffic (data) and to extend the size of local area networks without any loss in speed and capacity. One of these, the DRAMA protocol, is based on broadband technology and allows for dynamic allocation of bandwidth to clusters of nodes in the total network. In this paper we propose a distributed algorithm to allocate bandwidth in a *fair* manner, where we have defined fair to mean that every node in the network, regardless of its location, should have the same expected access delay to the net. We show by means of simulation that the algorithm is stable and within a few percentage points of the optimal solution. Stability is measured in terms of the time it takes the system to return to near optimal distribution of the bandwidth after strong disturbances. The algorithm is shown to handle total reallocation of the resources within $O(100\text{ms})$.

1. Introduction

The concept of extending local area networks to a larger community followed quickly once local areal network (LAN) technology became available. An additional driving force in advancing the technology was the possibility of integrating periodic and aperiodic traffic into one medium. Local area networks based on CSMA/CD such as Ethernet [1, 2] have limitations in terms of data rate which depend on the length of the bus and the number of nodes in the net. For example, without active repeaters, 10 Mb/s can be achieved in Ethernet over a distance of 0.5 km for one hundred users. Other systems, such as the Token Bus [3] or Token Ring [4, 5] have different characteristics but similar limitations. A considerable effort has been made to adopt these protocols to integrate different types of traffics such as data, voice and video [5]. Similarly, efforts have been made to extend the protocols to larger networks. Several multichannel LAN protocols have been proposed which expand the original CSMA/CD scheme to improve performance [6, 7, 8]. However these protocols do not integrate different types of traffic, nor do they expand the size of the LAN, nor can bandwidths be dynamically allocated. In [9, 10] a protocol is introduced which combines both of these aspects. It produces a gatewayless network covering a metropolitan area allowing for dynamic sharing of bandwidth among different types of traffic, i.e. periodic and aperiodic, and different clusters of nodes, i.e. different LANs; we shall refer to it as the DRAMA (Dynamic Resource Allocation in a Metropolitan Area) protocol.

As efficiently as the protocol seems to work as seen from initial coarse grained simulations, the simulations raised more questions than it answered. In this paper we address one of these questions in detail. In

any distributed system having a fixed amount of resources which needs to be allocated to the nodes, the algorithm to handle this allocation is crucial to the overall performance. The metropolitan area network we investigate consists of a number of LANs communicating via one broadband cable. The broadband frequency is divided into bands which constitute the resources in the system, and the LANs are the nodes. At any instant bands are assigned to LANs in a mutually exclusive way. How to reassign these bands in a changing environment is the core of this paper.

We have introduced a concept of fairness at the workstation level as follows. A workstation should have the same expected message delay regardless of the LAN to which it belongs. We have constructed a distributed algorithm which will reassign resources so that the network behaves "fairly" on the average. The performance criterion we set ourselves is that even total reallocation should be achieved within one second.

The paper is organized as follows. In Sections 2 and 3 we give a brief overview of DRAMA and an overview of the resource allocation algorithm. In Section 4 we present the simulation model and the metrics for measuring success or the failure of the algorithm. Section 5 discusses the simulation results.

2. The DRAMA Protocol

The protocol is designed for dynamically sharing the bandwidth of a broadband cable among groups of nodes in a large, integrated voice/data/video network. The amount of bandwidth available is assumed to be large, say 350-500 MHz. The nodes are each capable of transmitting voice, data and video and are clustered by distance and function into *local areal network groups (LANGs)*. This type of clustering is typi-

† Applied Software Development Center, China State Shipbuilding Corporation, Shanghai, Peoples Republic of China

‡ Chengdu University of Science and Technology, Chengdu, Sichuan, Peoples Republic of China

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear,

and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

cal among various locations of a company within a particular city, installations on a large ship or military base, or among different departments within a university. For logical purposes we consider that a single broadband cable connects all the nodes. The topology of the network is assumed to contain no closed loops in cable layout and to contain a direct cable path between any two nodes in the network. The cable bandwidth is frequency-divided into bands dedicated to particular LANGs and a global pool of bands that may be acquired by any of the LANGs. For each LANG, requesting, acquiring, or releasing a band depends on the current distribution and amount of traffic within that LANG relative to the current traffic within the entire network. In fact, the combination of the topology and the operation features of the system combine to effectively provide a *gatewayless* metropolitan area network. This obviates the delays at bridge points in such interconnected LANGs as in [11]

In recently proposed broadband systems such as [12] and [13], as well as in commercial systems currently available, the broadband frequency spectrum is *statically* partitioned by user group and/or by traffic class. For example, CableNet [14], Sytek's LocalNet [15], and Mitre CableNet [4] partition the bandwidth into fixed bands for particular applications of specific groups of users; some bands are permanently reserved for analog video channels, some are reserved for TDM multiplexing for a set of closely-located users, and some bands are dedicated to specific functions such as process control. However, a common characteristic of a network that supports such diverse traffic classes as voice, data, and video is that the bandwidth requirements of an individual LANG fluctuate widely over time compared to the LANG's average requirement. For example, a typical voice call lasts for at least several minutes, generating an average of 10 Kb/s of data. Propagation delay, from voice sample generation at the source to actually hearing the voice sample at the receiver, should be less than several hundred milliseconds [16]. However, there are many periods in a voice circuit when silence occurs. Data traffic is often generated in much shorter, more irregular bursts and can tolerate a delay of several seconds. Analog video circuits, however, require up to 10 MHz and digitized video transmission requires 1.5-140 Mb/s, depending on the quality and type of application [17,18]. By 1984 several companies, e.g. Picture Telcorp, and Compression Labs Inc., had introduced 56 kb/s video conferencing. (Because frequency-division multiplexing on the cable inherently requires analog signal transmission anyway, analog video is assumed in the proposed system both for simplicity and to avoid the enormous bandwidth requirements of digitized video.)

In such diversified systems, static partitioning according to average requirements will often waste idle bandwidth and at other times be insufficient to satisfy a LANG's traffic requirements even if most of the bandwidth elsewhere in the network is not in use.

Therefore an important objective of the proposed network is to allow each LANG to acquire a greater amount of bandwidth than would be possible if the set of bands were statically partitioned among the LANGs.

Another desirable objective of a large system is to allow all nodes to communicate with one another in a uniform way regardless of distance. A related objective is that the communication delay between nodes should be made as independent as possible of the distance between them; the only additional delay between more distant nodes should be due to the inherently longer propagation delay between them and not due to any extra switching or connection set-up. We accomplish the goal of a uniform communication method by allowing both internet and intranet traffic on each band and by using the integrated voice/data protocol proposed in [9] on each voice/data band. This protocol is based on CSMA/CD. Normally CSMA/CD is not suitable as a transmission protocol between nodes separated by more than several kilometers because the interval during which a collision can occur is directly proportional to the propagation delay between the most distant nodes. We circumvent this problem in our system by restricting transmission privileges on a band to exactly one LANG at a time, while allowing all LANGs to receive the transmissions. In this way, the CSMA/CD-based protocol can be used over the entire set of LANGs with the same efficiency as in a single LANG.

Since the underlying protocol is CSMA/CD, it is particularly important to reallocate in a fair manner. In CSMA/CD the amount of nonmessage traffic due to collisions is not a linear function of load. Therefore reducing the load on heavily utilized channels at the cost of increasing the utilization at lower utilized channels will produce an overall decrease of nonmessage traffic on the net.

Dynamic Bandwidth Allocation

In the protocol the bandwidth is frequency-divided into $M + 1$ bands; one band is reserved for a slotted band-control channel used by all the LANGs to coordinate band-sharing and the remaining M fixed-size bands (say, 10MHz each) that are available for voice, data and video transmissions. The M bands are partitioned into a set of *dedicated bands* and a set of *available bands*. Dedicated bands guarantee that no LANG "starves" and that each LANG's performance is at least that of a normal, solitary LAN using a baseband cable.

The available bands are either assigned to a LANG or are in a global pool. Bands in the global pool are shared via a dynamic, fully-distributed, band-sharing policy, explained below, that allows each LANG to obtain global bands based on its current needs, the current needs of other LANGs, and the current availability of global bands. The algorithm attempts to maintain a specified number of bands in this pool. When a band is acquired by a LANG, that

LANG has *exclusive* transmission rights to the band as long as the band remains allocated to it although each node in the system listens to all bands for packets addressed to itself. Therefore a node has a receiver for each band and enough frequency-agile transmitters for the maximum number of simultaneous voice circuits and simultaneous video circuits supported by the node. Giving transmission rights on a particular band to exactly one LANG at a time reduces the contention period from a network-wide propagation delay to a single-LANG delay. Because all bands can be received at all nodes, there is a uniform communication mechanism among all nodes in all LANGs. Simple wide band repeaters may be required over a long distance network to overcome attenuation and/or signal distortion.

The LANGs coordinate the request, acquisition, and release of global bands via a slotted band-control channel in which each LANG owns a particular slot that is used for requesting, acquiring, and releasing global bands. The slotted scheme allows the band-sharing policy to be collision-free and guarantees that bandwidth reallocation can be performed in a fixed amount of time. Specifically, if the current number of available global bands is at least equal to the demand for additional global bands, then all acquisitions are accomplished in one band-control cycle. If the number of bands is insufficient for current needs so that possibly a band should be released by one LANG and acquired by another, then the request, release, and acquisition occur within at most two band control cycles.

When a node wishes to transmit (voice, data or video), it must choose among the bands on which its LANG currently has transmission privileges. We refer to the policy for choosing among the transmission bands as the *traffic-placement policy*. The traffic-placement policy has to place a LANG's voice/video traffic on as few of its bands as possible to allow release of underutilized bands, whereas to minimize delay, data traffic is spread as evenly as possible over the LANG's current voice/data bands. In this paper we shall not discuss this policy any further but shall assume that the policy will produce available bands not having any periodic traffic.

Voice/Data Transmission Protocol

In this section we describe the integrated voice/data transmission protocol originally proposed in [9] for a single-channel LANG. It is slightly modified here for the multi-channel system. (Each modification is noted below.) In this protocol, the fraction of a band's capacity allotted to each traffic type depends on the current voice and data loads. Time on each band is slotted into frames. Each band's frames are delimited by "frame-begin" markers broadcast by the band's current *band-leader*. In the single channel version the frame-begin marker consists of a few synchronization characters; in the multichannel version the frame

marker also contains the LANG-ID of the LANG to which the band is currently allocated (for reliability and robustness). At any time, exactly one node in the LANG is responsible for broadcasting the frame-begin markers for a particular band, although the identity of the node may vary over time. In the single channel system, this node is referred to as the *designated node* of the LANG. For clarity, in the multichannel system we refer to this node as a *band-leader*; note that a LANG may have multiple band leaders if it holds multiple voice/data bands.

The band-leader of the LANG's lowest-numbered *dedicated* band is also the current LANG-leader, transmitting the LANG's band-control slot in the band-sharing scheme, and establishing the "frame-begin" markers on any voice/data bands that the LANG obtains. This LANG-leader is initially the band-leader of each newly allocated band to the LANG; if a voice call is placed on that band, the node placing the first voice call becomes the band-leader. If this node fails, the LANG-leader again becomes the band-leader.

Each frame is partitioned into a voice region and a data region. The boundary between the two regions varies from frame to frame, depending on the number of voice calls in talk-spurts during that frame. The data region (voice region) may consume the entire frame, if there is an absence of voice traffic (data traffic). Call set-up and other data packets are transmitted in the data region using nonpersistent CSMA/CD. The voice region provides one virtual circuit for each established (one-way) voice call. Silent periods are not transmitted, hence the length of each slot varies. A node having an ongoing call has a logical position number for this call which is changed by calls which terminate and precede it in the logical order. Termination is broadcast by a node by means of a control byte at the end of each voice slot.

The overhead per frame is five bytes plus one byte per periodic slot (for voice and video).

3. The Band Allocation Algorithm

The algorithm assumes that LANGs can communicate through the band control channel depicted in Figure 1. The length of the control cycle is determined by the following formula: $control.cycle.length = \max(algorithm.execution.time + net.propagation.delay + information.broadcast.time, 3 * frame.length)$. For instance, if we assume a 1 Mb/s channel, 5 LANGs, distance between LANGs of 5 km, a maximum length of 1 km per LANG, a propagation delay of 5 μ sec/km, then the cycle length of this band is approximately 30 milliseconds. The quiescent period, labeled QP, is necessary for the algorithm below to perform all its calculation after it has read the preceding LANG slot information.

The LANG leader writes the information obtained from executing the band allocation algorithm into its slot as shown in Figure 1. The detailed struc-

ture of each slot is shown in Figure 2. Figure 3 describes the LANG band status component of Figure 2. The band status reports for each band in the net who owns it currently, whether it is dedicated or in the global pool and what type of traffic is on it.

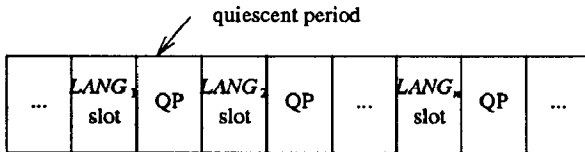


Figure 1: Band-Control Channel Structure

LANG ID	Network Average Utilization	Current Acquisition Threshold	Band Release	Band Acquisition	LANG Band Status
---------	-----------------------------	-------------------------------	--------------	------------------	------------------

Figure 2: Band-Control Slot Structure

band#, dedicated, voice/data global video/voice/data data	...	band#, dedicated, voice/data global video/voice/data data
---	-----	---

Figure 3: LANG Band Status Structure

The basic idea of the algorithm is to allocate bands to LANGs based on their weighted network utilization, where utilization is the ratio of band busy time over band time. If we define network utilization as the average utilization of all assigned bands and LANG utilization as the average utilization over all bands assigned to this LANG, then we can restate the original fairness objective as follows: Each LANG's utilization should be as close to the network utilization as possible. Ideally the difference should be zero at all times.

To reduce the frequency of band reallocations, we introduce a tolerance interval around the average network utilization. In Figure 4 we depict a situation where all LANGs' utilizations are within this tolerance interval. In this case the algorithm will do nothing if the global pool also has the desired size (steady.pool).

However, a number of things can change the situation. The global pool could be too large or too small or could be empty all together. A particular LANG's utilization could suddenly increase (or decrease) and move it outside the tolerance interval. Table 1 describes criteria under which the algorithm will release or acquire a band from the global pool. The term $adj.util$, computed as $(curr.util * lang.band\#) / (lang.band\# - 1)$, is the new LANG utilization if a band were released.

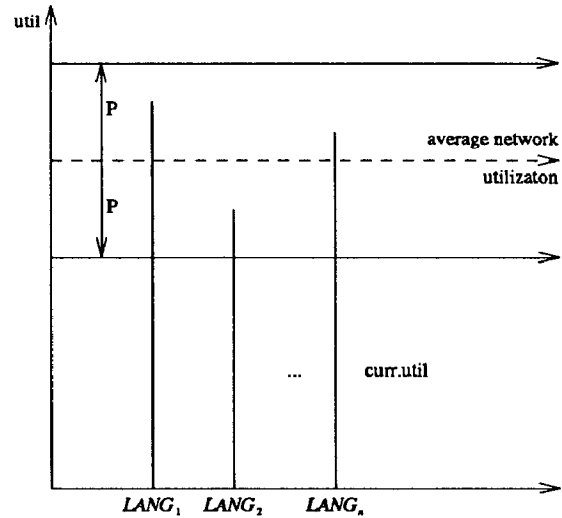


Figure 4: Steady State LANG Allocation

Criterion	Acquire	Release
pool.size > steady.pool	curr.util > avg.util	$(adj.util \leq avg.util - p)$ & num.data.bands > 0
pool.size = steady.pool	curr.util > avg.util + p	$(adj.util \leq avg.util - p)$ & num.data.bands > 0
pool.size < steady.pool	curr.util > avg.util + p	$(adj.util \leq avg.util + p)$ & num.data.bands > 0
pool.size = 0		$(adj.util \leq avg.util + p)$ & num.data.bands > 0

Table 1: Allocation Policy

The detailed algorithm is given in [19]. In particular it describes the steps necessary to insure that the LANG needing additional bands the most ($|curr.util - avg.util|$ is maximal) will get available bands first and that those LANGs needing bands least will give up bands when required. A current acquisition threshold is maintained in the control band; it is non-zero when the pool is empty and a LANG needs a band. This acquisition threshold is set by a needing LANG to the maximum of its own utilization or the old threshold. When a band becomes available a LANG can only take it if its utilization is greater than the latest acquisition threshold it has seen. We summarize the algorithm actions in Table 2.

Condition Causing Action	Action
if the pool is very big	Any LANG whose utilization is high acquires an additional band.
if the pool is too small	Any LANG whose utilization is low releases one band.
if a LANG sticks out over the tolerance interval	The LANG attempts to acquire a band.
if a LANG sticks out below the tolerance interval	The LANG attempts to release a band.

Table 2: Summary of Allocation Algorithm

In Table 2, we say "attempt" because in order to achieve steady state a band should not be added to (or deleted from) a LANG if that change causes the LANG to move outside the tolerance interval; the conditions of

Table 1 must be satisfied before a band can be reallocated.

4. Simulation Model

We have implemented a model of a network using the DRAMA protocol in SIMSCRIPT II.5. The simulation program is highly parameterized to allow experimentation with different loads and different thresholds for band reallocation. Use of a simulation language like SIMSCRIPT is appropriate for this study: our program includes both code for discrete event simulation components (using simulation features of the language to generate loads which fluctuate over time) and code implementing aspects of the DRAMA protocol (using procedural features of SIMSCRIPT to monitor that load and reallocate bands according to the DRAMA protocol).

Model Objectives

The purpose of the models developed for this experimentation is to create a partial implementation of the DRAMA protocol, to evaluate performance of the proposed system, to enhance our understanding of its behavior, and to better understand the effects of a variety of parameters on system performance. Factors which significantly influence system behavior are discussed in Section 5.

The experimentation is intended to evaluate the effectiveness of our band reallocation strategy using criteria discussed below. With this goal, our model incorporates the following assumptions.

Traffic:

- The traffic placement policy of each LANG is not relevant since reallocation is based on load disparities among LANGs.
- Different types of traffic on a band, that is, data, voice or video, are not distinguished since reallocation is based on relative loading. We assume that a traffic placement policy can be found so that bands can almost always be released as required. (Subsequent simulations not reported here indicate that this assumption is valid.)
- Time between traffic changes on each LANG is exponential. This represents placing a new message on the network.
- Since load values on a network are autocorrelated over time, the effect of individual traffic changes on a LANG can be represented satisfactorily by changing the existing load on that LANG either up or down.
- The amount of each change in load at discrete times is Poisson related to the length of time until the next change.

Protocol Simulation:

- The time required for protocol computations (such as the algorithm to compute band allocation) is small and can be performed in the time a slot passes between LANGs. Thus the simulation model does not explicitly represent this time.

Thus the granularity of the simulation is at the level of adjusting a utilization statistic (in effect, the amount of traffic) on the bands assigned to individual LANGs.

After initialization, the executing simulation model consists of two active components. One is a collection of identical traffic generators, one per LANG. Each generator changes the amount of traffic on each LANG by adjusting a variable representing utilization for each band assigned to the LANG. The assumptions which the traffic generator incorporates about stochastic behavior of the modeled system are outlined in Table 3. Since no other program components include representations of stochastic behavior, the table indicates that limited stochastic assumptions are incorporated into the model.

Behavior	How Generated	Parameters	How Defined
Time between traffic changes on each LANG	Exponential	mean time between changes	run parameter
Amount of traffic change at each adjustment	Poisson/(100 * number of bands in LANG) with minimum of 0 and maximum of 0.95	10 * Lambda * (time until next change)	Lambda defined at run time
Direction of traffic change	half up, half down	none	not applicable

Table 3: Stochastic Behavior of Traffic Generator

The second active component implements important aspects of the DRAMA protocol by monitoring band utilization and reallocating bands among LANGs according to the protocol. This program component is entirely procedural and includes no stochastic components.

In Figure 5, decisions to reallocate are made at discrete time values, t_1, t_2 , etc. The DRAMA protocol component examines utilization of each LANG and compares it to the average network utilization at these time values. For example, at time t_1 , all bands' utilization is within the tolerance interval, but the pool of unallocated bands is larger than the target pool size, so a band is allocated to LANG 2. The other parts of the figure at t_2 , etc. have similar interpretations.

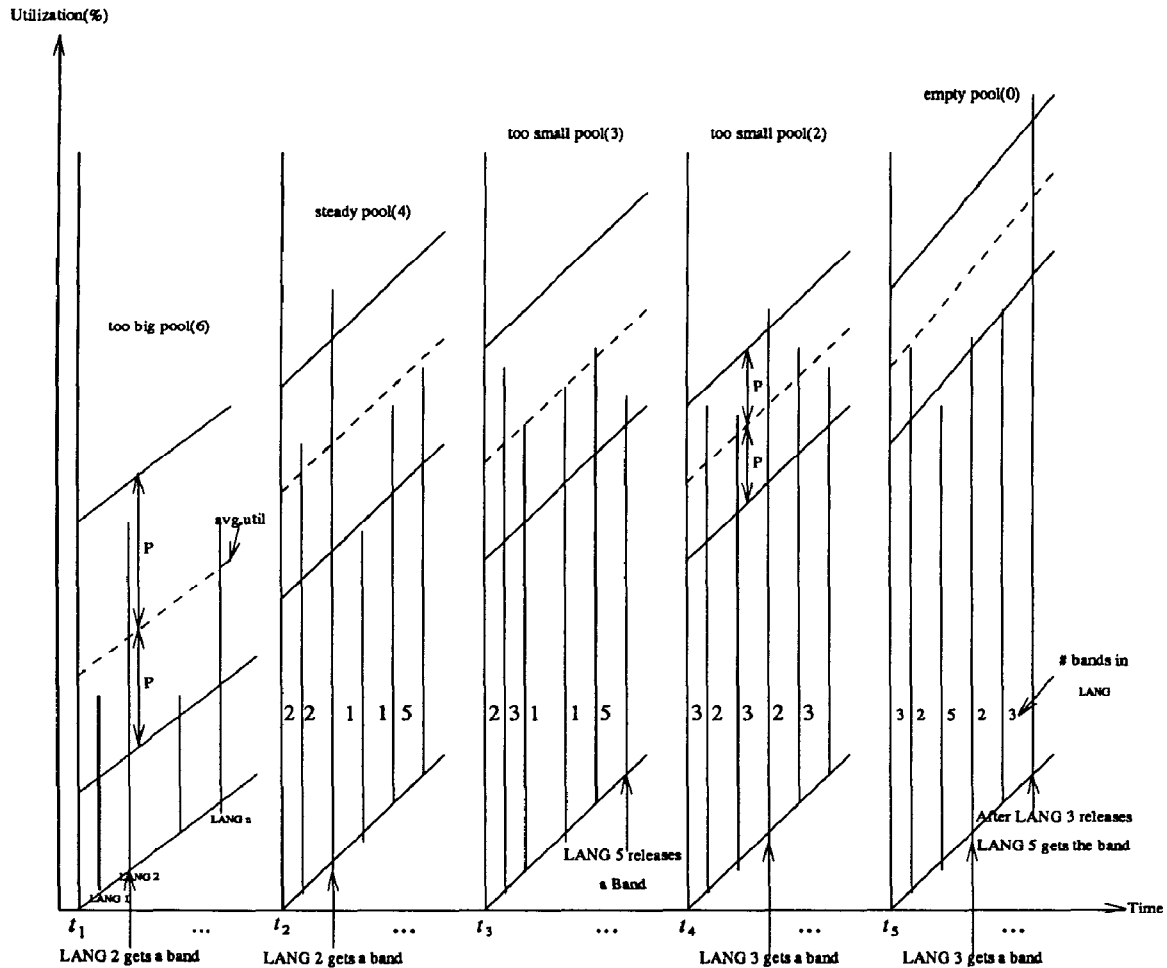


Figure 5: Sample DRAMA Protocol Behavior

Key variables used to parameterize simulations include: number of bands (N.BAND), number of LANGs (TOTAL.LANG.NUM), length of simulation (SIMUL.TIME), width of tolerance interval (P), and target global pool size (STD.POOL).

The model is started "idle and empty." Each LANG is initially allocated one band. Each traffic generator, one per LANG, is restrained to only generate increases in traffic until a specified load is achieved; after this, each traffic generator randomly generates half increases and half decreases. These steady increases in load will force allocation of additional bands to each LANG. Data collection begins after the specified load is achieved. In a typical simulation, this start up time requires 1.8 seconds of simulation time in a run of 12 seconds of simulation time.

Performance Statistics Generated

Each simulation run produces many statistics. Most are used for model validation; others are of direct interest in evaluating the effectiveness of the protocol.

These performance measures are defined here.

- **Fairness:** The average difference between individual LANG utilization and overall network utilization, i.e.

$$\text{Fairness} = \frac{\sum_{\text{LANG SAMPLES}} \sum_{\text{LANG}} |\text{OLD.UTIL}(\text{LANG}) - \text{AVR.UTIL}|}{\text{NO.OF.LANGS} \times \text{NO. OF LANG SAMPLES}}$$

The value is observed once per slot.

- **Responsiveness:** The LANGs' average excess deviation from the tolerance interval. Thus if a LANG's utilization is within the tolerance interval, the value is 0; if the LANG's utilization is out of the tolerance interval, the value is the excess. All these values are then averaged. Thus

$$\text{Responsiveness} = \frac{\sum_{\text{LANG SAMPLES}} \sum_{\text{LANG}} \text{NONTOLERANT.UTIL}}{\text{NO.OF.LANGS} \times \text{NO. OF LANG SAMPLES}}$$

where NONTOLERANT.UTIL is $\max(0, |\text{OLD.UTIL}(\text{LANG}) - \text{AVR.UTIL}| - P)$. The value is observed once per slot.

- **Deviation from optimal:** Average deviation of the band allocation generated by the DRAMA protocol and a band allocation which produces minimal deviation in utilization among the LANGs.

Deviation =

$$\frac{\sum_{\text{LANG SAMPLES}} \sum_{\text{LANG}} |\text{OPT.NO}(\text{LANG}) - \text{L.BAND.NUM}(\text{LANG})|}{\text{NO.OFLANGS} \times \text{NO. OF LANG SAMPLES}}$$

where OPT.NO(LANG) is the value computed by an algorithm presented in [19].

If all LANGs have identical utilization, the deviation is 0. Since bands must be assigned in integer values to LANGs, a 0 deviation is generally not achievable. The protocol we present does not necessarily produce an band allocation which results in minimal deviation. We present a statistic in Section 5 which is the average deviation of the band allocation generated by the DRAMA protocol and the solution as generated by the algorithm presented in [19]. We assert that this average deviation is an informative measure of the DRAMA protocol.

- **Recovery time from disruption:** Another measure of performance of the protocol the time required to recover from dramatic changes in load on an individual LANG, either resulting from large increases or large drops in traffic (such as might result of initiating or terminating a video hookup).

Experimentation

Using this simulation model a collection of experiments were performed which are reported in Section 5.

- **Variation in "P":** In this collection of experiments, the width of the tolerance interval was changed. The goal was to determine the sensitivity of the stated performance parameters to this parameter.
- **Variation of load:** The simulation model was run under a variety of loads to evaluate the performance of the DRAMA protocol in low to high traffic levels.
- **Variation of pool size:** A key concept of the DRAMA protocol is reserving a collection of bands to be used for load balancing. These experiments were designed to determine the optimal target pool size.
- **Response to drastic changes:** Experiments were performed in which both dramatic increases and dramatic decreases in load of an individual LANG were introduced exogenously. The goal was to measure the time required for recovery.

5. Results

In this section we present the results of our experimentations with the model introduced in Section 4. We shall elide results concerning the model validation since they simply show that performance is independent of such parameters as random number seeds, simulation time, and traffic distribution. We will

discuss the relation of the metrics we introduced — fairness, responsiveness, deviation from optimum — and four protocol parameters we identified as relevant — tolerance interval size, capacity, size of band, pool-size. In Figures 6 through 11 curves are obtained by applying a spline (done by PIC software) to the data points.

Figures 6 and 7 clearly indicate that the number of bands relates directly to performance as measured both by fairness and responsiveness (data points are at increments of 10 points). We varied poolsize and the magnitude of changes in traffic but neither produced any significant changes in performance. We can see from the graphs in Figures 6 and 7 that the curves tend to flatten out at about 50 bands, or about 10 bands per LANG (the number of LANGs is 5). The conclusion we draw from this experiment is that for good performance *the average number of bands per LANG should exceed 10*. For example, if a network has 20 LANGs and the total available bandwidth is 400 Mb/s then the bandwidth per band should not exceed 2 Mb/s.

Figures 8 and 9 illustrate two characteristics of the algorithms behavior (data points are at the given pool_size coordinates and for pool_size = 1, 2, 3). One, there is a significant difference between having no pool at all and having at least one band in the pool. However, as far as performance goes it does not matter very much whether the pool is bigger than one (we shall come back to this point when discussing the effect of load). We note that under certain settings — the tolerance interval being very narrow — the algorithm can not maintain the poolsize at the desired state.

A strong correlation between the size of the tolerance interval and both responsiveness and fairness is shown in Figures 8 and 9. Particularly interesting is the inverse performance relation between the two metrics. In a sense this is what can be expected because the narrower the interval the more difficult it will be to bring all LANGs utilizations back (responsiveness), but once achieved the closer to the network average (fairness) the LANGs will be. On the other hand we do not want to increase the number of bands in the pool arbitrarily because, being unused, this will reduce the capacity of the total network. The conclusion we draw from these sets of experiments is that we should *set the pool size to one and the interval size to 13%* (when measured at 60% of capacity).

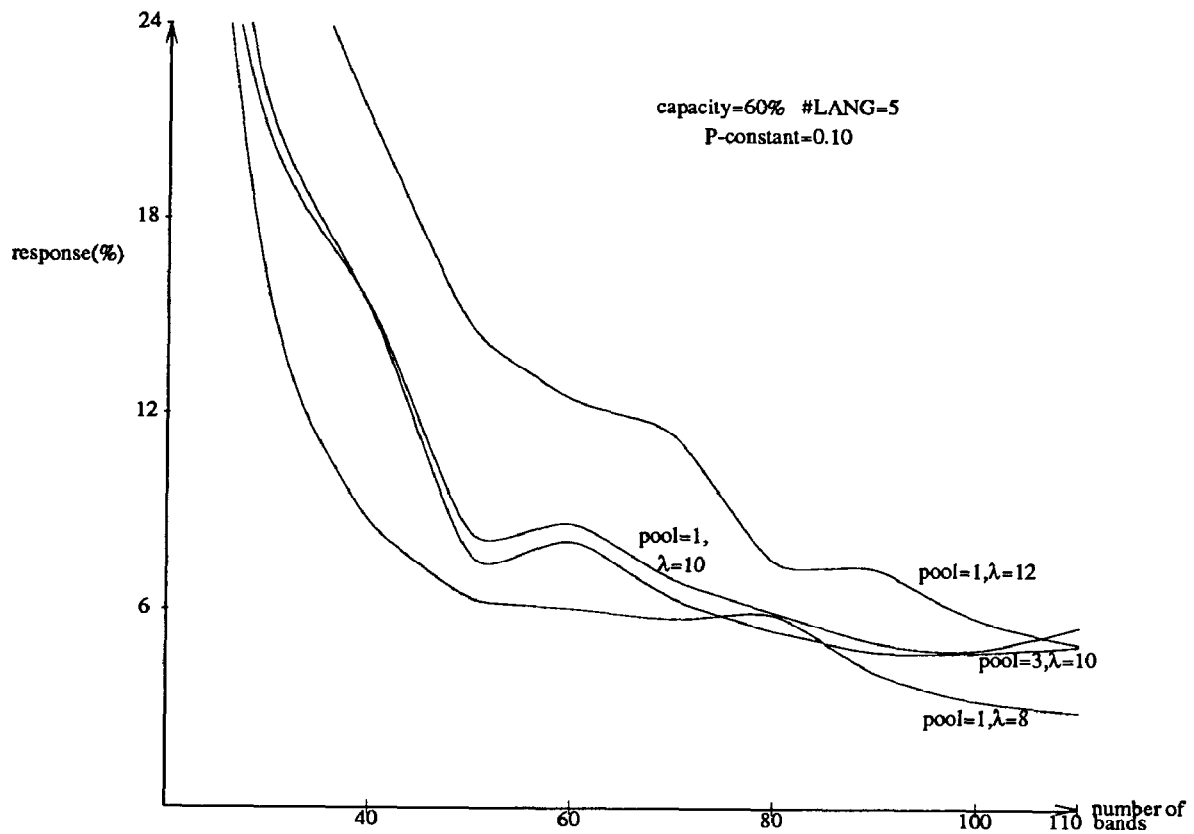


Figure 6: Protocol Performance - Band Size

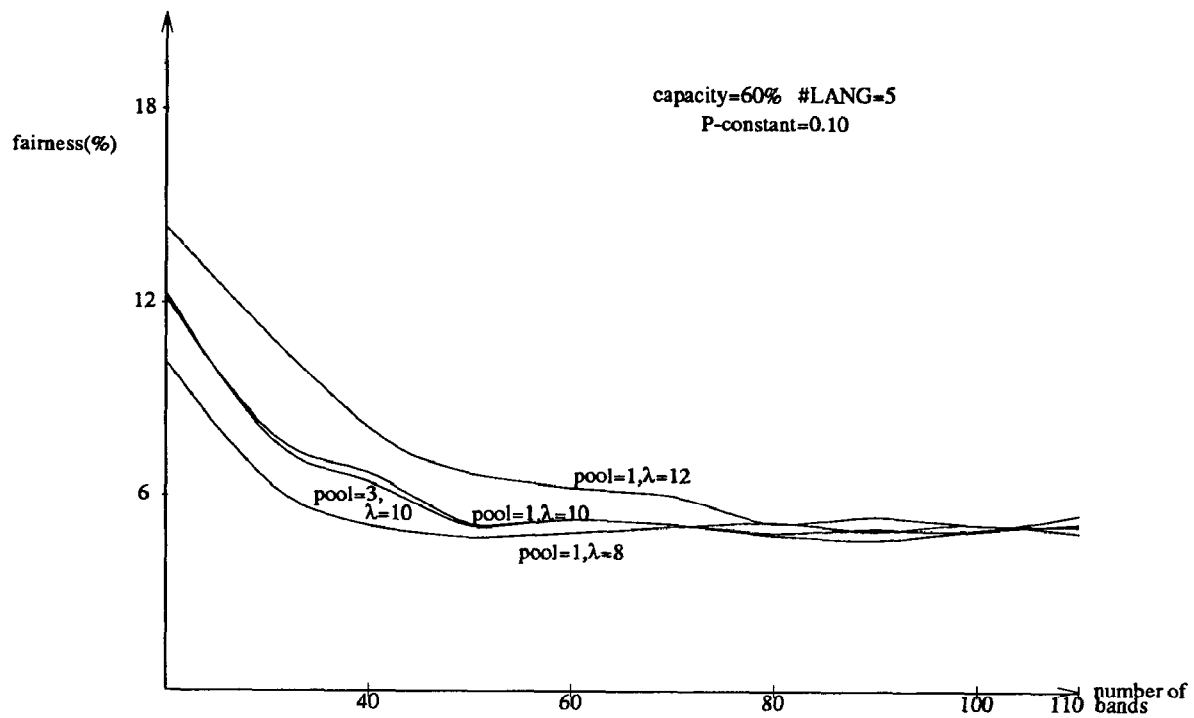


Figure 7: Protocol Performance - Band Size

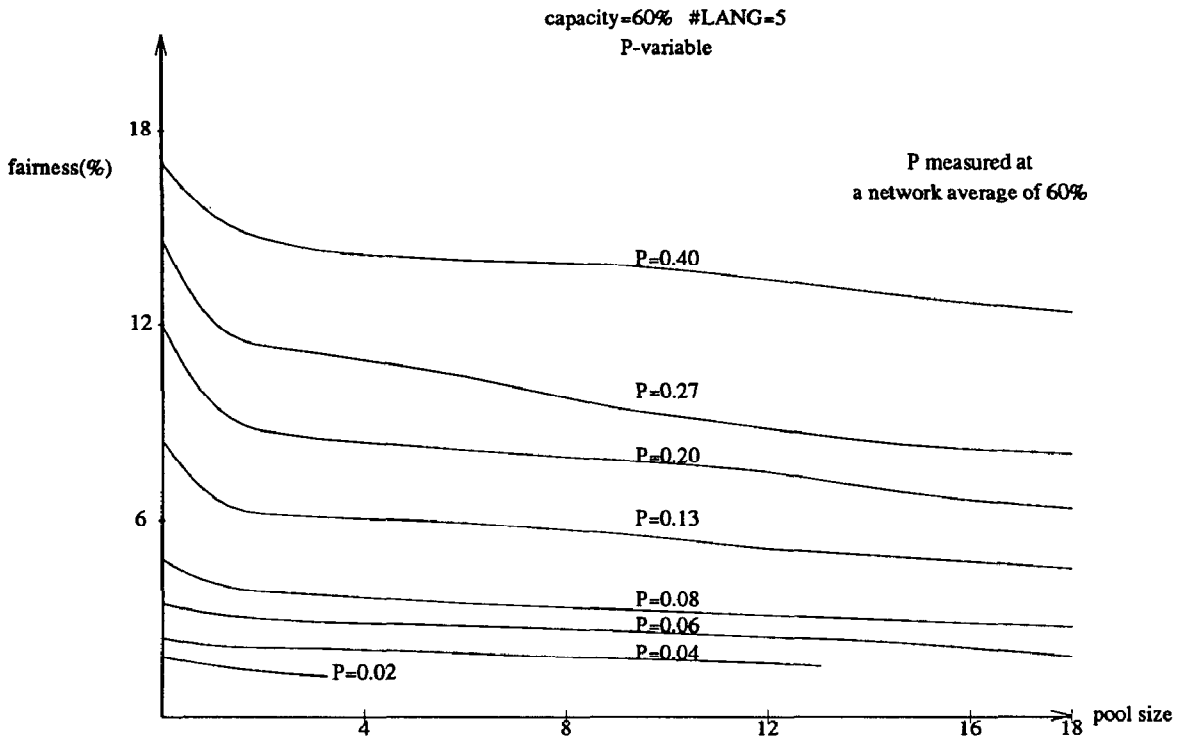


Figure 8: Protocol Performance - Pool and Variable Interval Size

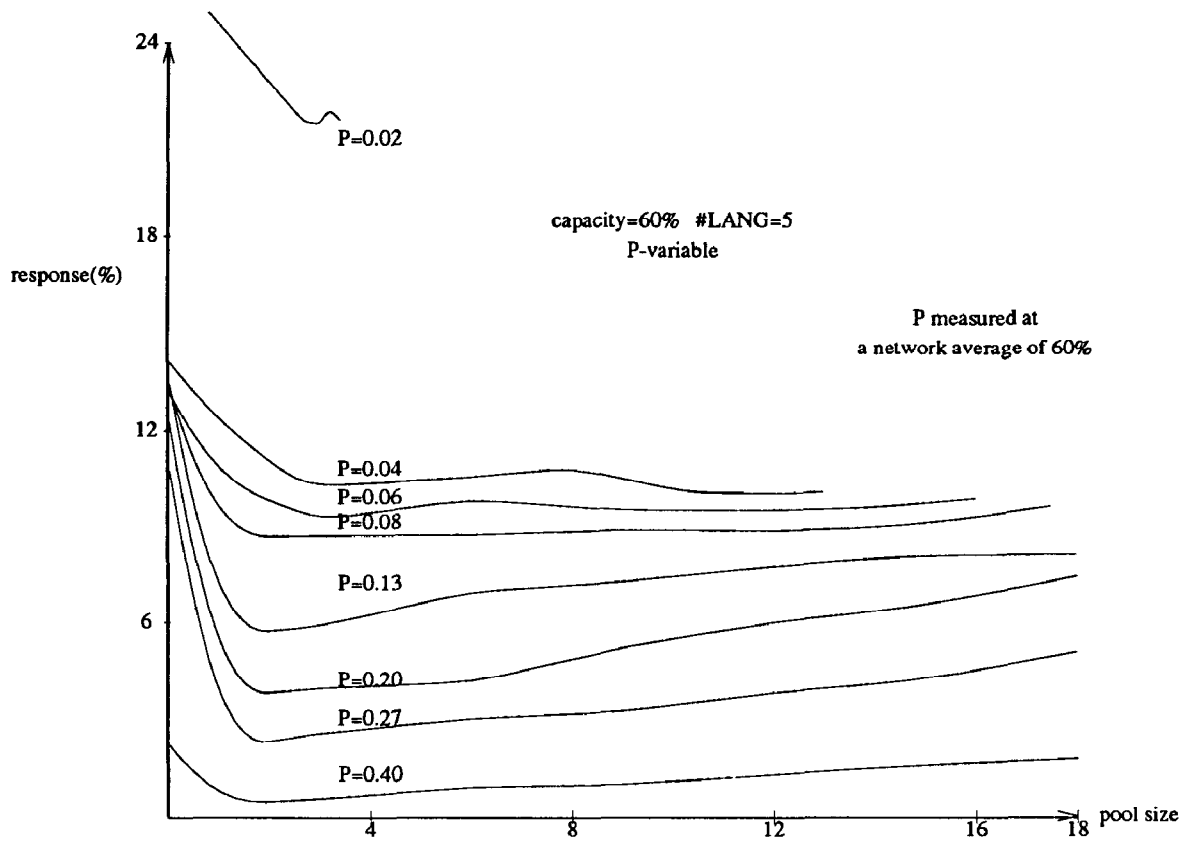


Figure 9: Protocol Performance - Pool and Variable Interval Size

Additional experiments were performed similar to those depicted in Figures 8 and 9, where the tolerance interval is a function of the average network utilization. In these experiments the tolerance interval size was kept constant regardless of the network average. Results indicate a slight performance degradation and our conclusion is that *the variable interval size is the preferred method*.

In Figure 10 we have modeled the dependency of the algorithm on the load of the network. (Data points are at every 10% of capacity). We have plotted the results for a variety of traffic change intensities; the reason all graphs end at about 90% lies in the way our model is implemented. We do not model a changing level of utilization but rather model incremental changes and therefore can not maintain a level beyond 95%. Between 90% and 95% data points gyrate wildly but stay within the envelope of the extrapolated graphs. Fairness is practically independent of the load on the network. Responsiveness at low load levels (<40% of capacity) is higher due to the fact that the algorithm is invoked infrequently. The independency of the algorithm's performance on the load is also a partial explanation of the counterintuitive independence on the poolsize. Our conclusion is that *the algorithm will perform equally well under almost all load conditions of the network*.

Finally, Figure 11 illustrates the stability and the near optimality of the algorithms. (Points in the graph are at intervals of 33 ms). The lowest graph represents the best any algorithm could do in achieving fairness. Even there we have a small percentage deviation from zero due to the constraint of integer solutions. The horizontal line in Figure 11 gives the average deviation of our algorithm from this optimal solution as averaged over the total simulation time. The third graph shows how quickly the system returns to a stable state after a strong disturbance. We ran experiments with symmetrically opposite changes by suddenly dropping the load of the net; the results were equivalent to the ones for sudden increases in load. The conclusion we draw from this figure is that not only is *the distributed scheme near optimal but it is also a stable systems*.

6. Conclusions

The DRAMA protocol extends the advantages of LANGs in such a way that the protocol can be used for metropolitan areas for large number of users. One of the key concept behind the protocol is the use of broadband technology and the splitting of a large frequency channel (~500 Mh/s) into many small bands. Similarly all nodes in the net are grouped by proximity into, possibly changing, LANGs. Bands are then assigned on a mutually exclusive basis to individual LANGs; each LANG can only send on the band it owns but receives on all bands. Since currently DRAMA is based on CSMA/CD, a most crucial performance factor is the round-trip delay. By the device of a

mutually exclusive band assignment we have reduced the round-trip delay from one for the entire net to one for the constituent LANGs.

For this protocol to succeed we have to provide a mechanism whereby the assignment can be changed automatically as different LANGs exhibit different traffic utilizations and different mixes of different types of traffic at any point of time. This will allow for load balancing and also will reduce the amount of noise due to collisions because in CSMA/CD the ratio of noise and load is a non-linear function.

In this paper we have introduced a heuristic algorithm to solve the band assignment problem in a distributed fashion such that band assignment is handled on the $O(10ms)$. That is, it will take on the $O(10ms)$ for a LANG to acquire or release a band. The algorithm's objective is to provide a 'fair' assignment where fair is defined as each node having the same expected access delay to the net independent from location and time.

We have shown through simulation that the algorithm will produce an assignment which on the average is within 4% of the optimal assignment. The optimal assignment was computed as the exact solution of a non-linear integer programming problem. In addition we have shown that the algorithm is stable and will within 300 ms completely reassign resources within the net due to major disturbances such as an individual LANG going down. Lastly we have shown that the algorithm's performance is independent from the networks load.

In this work we have demonstrate the conceptual feasibility of the DRAMA protocol. Since the software aspects of the protocol appear functional, a cost/benefit analysis of hardware costs is appropriate; it is likely that the benefits of VLSI and large scale production will result in reasonable costs.

Due to the excellent performance of the algorithm we have not considered a number of further avenues of refinements. For example, at the moment the algorithm will reallocate one band at a time and one could investigate performance when bands are reallocated in blocks. However a number of questions still remain unanswered. We performed the simulations at the level of band utilizations and did not consider the impact of individual messages. It is important to evaluate various ways of placing data and periodic traffic on the individual bands to see how they effect band allocation and message delay. In current investigations we are experimenting with models which simulate various traffic placement policies and model the interaction of band allocation and traffic placement.

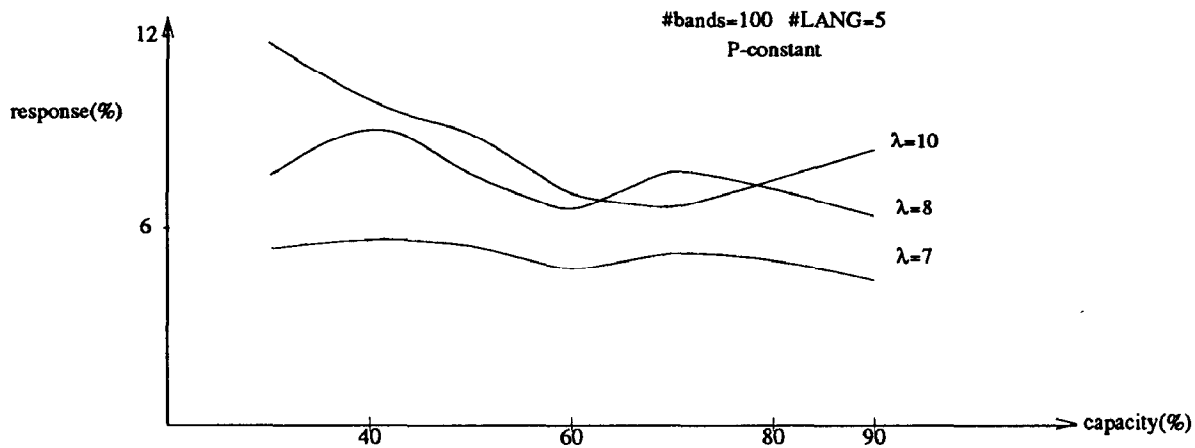


Figure 10: Protocol Performance - Capacity

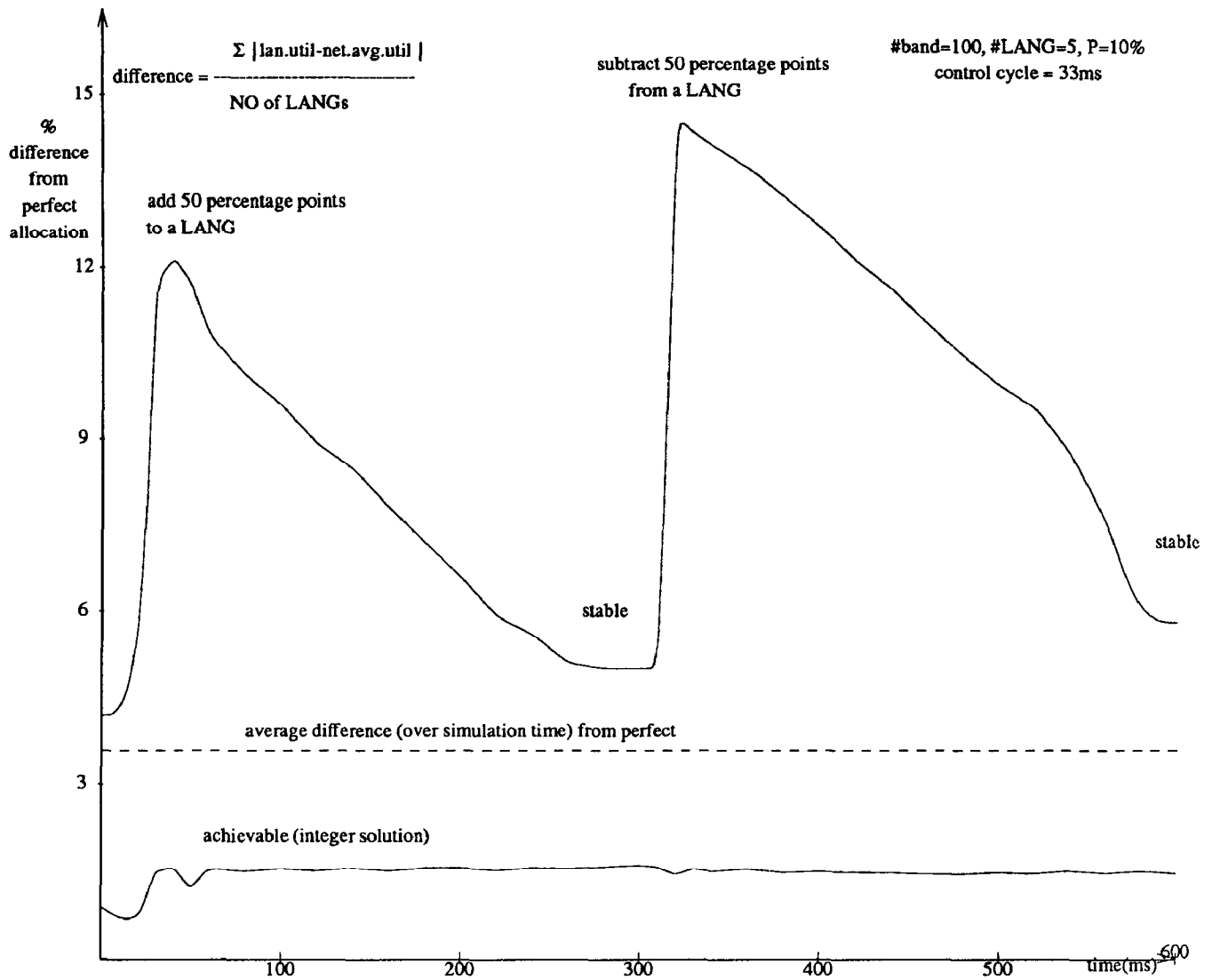


Figure 11: System Stability

References

1. R.M. Metcalfe and D. R. Boggs, "Ethernet: Distributed packet switching for local computer networks," *Commun. ACM*, vol. 19, ACM, July 1976.
2. Digital, Intel, and Xerox, *The Ethernet, a local area network. Data link layer and physical layer specifications*, Sept. 30. Version 1.0
3. F.A. Tobagi and M. Fine, "Performance of Unidirectional Broadcast Local Area Networks: Expressnet and Fasnnet," *IEEE Journal on Selected Areas in Comm.*, vol. SAC-1, pp. 913-925, November 1983.
4. T.B Fowler and Holmgren, S.F., "A wideband Cable Bus Local Area Network," *COMPCON*, pp. 405-414, Fall 1982.
5. D. Karvelas and A. Leon-Garcia, "Performance of Integrated Packet Voice/Data Token-Passing Rings," *IEEE Journal on Selected Areas in Communications*, vol. SAC-4, No 6, pp. 823-832, Sept. 1986.
6. Marco A. Marsan and D. Roffinella, "Multichannel Local Area Network Protocols," *IEEE J-SAIC*, pp. 885-897, Nov. 1983.
7. Imrich Chlamtac and Ora Gynz, "Performance Models of Asynchronous Multitrunk Hyperchannel Networks," *IEEE Transaction on Computers*, vol. C-36, no.2, pp. 138-146, Feb. 1987.
8. P.C. Wong and T.S. Yum, "Design and Analysis of a Contention - Based Lookahead Reservation Protocol on a Multichannel Local Area Network," *IEEE Transactions on Communications*, vol. 36, no.2, pp. 234-238, Feb. 1988.
9. S. Sharrock, K. Maly, S. Ghanta, and H. Du, "A Framed, Movable-Boundary Protocol for Integrated Voice/Data in a LAN," *Proceedings, SIGCOM '86*, p. 9 Pages, August 1986.
10. S. Sharrock, K. Maly, S. Ghanta, and H. Du, "A Broadband Integrated Voice/Data/Video Network of Multiple LANs with Dynamic Bandwidth Partitioning," *Proceedings, INFOCOM '87*, pp. 417-425, March 1987.
11. G.M. Exley and L.F. Merakos, "Throughput - Delay Performance of Interconnected CSMA Local Area Networks," *IEEE Journal on Selected Area in Communications*, pp. 1380-1390, Dec. 1987.
12. M. Hatamian and E.G. Bowen, "Homenet: A Broadband Voice/Data/Video Network on a CATV System," *AT&T Technical Journal*, vol. 64, pp. 347-367, February 1985.
13. A.N. Netravali and Z.L. Budrikis, "A Broadband Local Area Network," *Bell System Technical Journal*, vol. 64, no. 10, pp. 2449-2465, December 1985.
14. M.W. Rahm, "CableNet: A User Perspective," *COMPCON*, pp. 337-342, Fall 1982.
15. G. Ennis and P. Filice, "Overview of a Broad-Band Local Area Network Protocol Architecture," *IEEE Journal on Selected Areas in Communications*, vol. SAC-1, No.5, pp. 832-841, November 1983.
16. N.F. Maxemchuk, "A Variation on CSMA/CD That Yields Movable TDM Slots in Integrated Voice/Data Local Networks," *Bell Systems Technical Journal*, vol. 61, no 7, pp. 1527-1550, September 1982.
17. J.O. Limb, R.F.W. Pease, and K.A. Walsh, "Combining Intraframe and Frame-to-Frame Coding for Television," *Bell Systems Technical Journal*, vol. 53, No. 6, pp. 1137-1173, July-August 1974.
18. H. Yasuda, F. Kanaya, and H. Kawanishi, "1.5444-Mbits/s Transmission on TV Signals by Interframe Coding System," *IEEE Transactions on Communications*, pp. 1175-1180, October 1976.
19. Kurt Maly, C. Michael Overstreet, Xia-Ping Qiu, and Deqing Tang, "Dynamic Resource Allocation in a Metropolitan Area Network," Technical Report 88-03, Computer Science Department, Norfolk, VA 23529-0162, Feb. 16, 1988.