AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Traffic Capabilities of Two Rearrangeably Nonblocking Photonic Switching Modules

By R. A. THOMPSON*

(Manuscript received June 10, 1985)

The architectures of two small switching networks are compared as potential implementations of a 4 × 4 photonic switching module. Such a module would be made by interconnecting several 2 × 2 photonic directional couplers on a single $LiNbO_3$ substrate. While both networks are rearrangeably nonblocking, we investigate whether one network requires significantly more rearrangements than the other. The analysis includes transient, Monte Carlo simulation, and Markov steady-state techniques. We conclude that the traffic capabilities of the two structures are not significantly different, and that selection of an architecture can be based on other criteria, like loss, crosstalk, or ease of manufacture.

I. INTRODUCTION

The percentage of the world's voice and data communications that is carried by optical fibers increases daily. The importance of research in photonic switching increases with it. A promising element for the implementation of photonic switching systems is the photonic directional coupler, made from titanium-diffused lithium niobate. The current state of this technology allows a level of integration of tens of devices on a single substrate. We investigate two competing architectures for a 4 × 4 switching module, fabricated in this technology, that could be useful in building photonic systems.

In the next two sections of the paper, we review some basic concepts

*AT&T Bell Laboratories.

of switching networks and describe the two candidate photonic switching modules. In Sections IV and V, we present the results of transient analysis and Monte Carlo simulation, respectively, as applied to the two switching modules. In Sections VI through VIII, we present steady-state Markov analysis as applied to a generalized module and to each candidate switching module, respectively. In Section IX, we enumerate the network configurations in each of the candidate switching modules.

II. SWITCHING NETWORKS

The traffic-handling performance of a switching network depends on two entities:

- The topology of the internal elementary switching stages.
- The rule by which paths through the network are selected.

2.1 Topology

One topological classification applied to switching networks is the hierarchy of "blockingness." Networks are classified by their ability to establish connections, especially sequences of connections with intermediate disconnections. An excellent tutorial and summary of the state-of-the-art in switching network topologies is found in Ref. 1.

2.1.1 Strict- and wide-sense nonblocking networks

A switching network is *nonblocking* if any desired connection between two idle ports can be completed immediately without interference from connections that may be already established in the network. If this property is independent of any switching rule used to select paths through the network, then the topology is *nonblocking in the strict sense.* If the property is true, provided the paths assigned to the established connections were selected under some switching rule, then the topology is *nonblocking in the wide sense.*

Such topologies are *serial nonblocking* in that arbitrary sequences of input-output pairs can be connected and disconnected without blocking. Neither switching module described in this paper is nonblocking in either sense.

2.1.2 Rearrangeably nonblocking and blocking networks

A switching network is *rearrangeably nonblocking* if any desired connection between two idle ports, which might be temporarily blocked by connections already established in the network, can be completed possibly after some established connections are moved to different paths. Such topologies are *parallel nonblocking* in that any set of I/O pairs can be connected without blocking if the network is initially idle and the set is known in advance.

A switching network is *blocking* if there exist network configurations of established connections from which some connection between two idle ports can not be completed, even with rearrangement of the established connections. Both switching modules under consideration in this paper are rearrangeably nonblocking.

2.2 Rule

A switching network generally allows more than one path by which ports on either end of the network may be connected. The *switching rule* is the means of selecting one path. For certain special cases, such as the strict-sense nonblocking networks, the performance is independent of the switching rule. In general, however, the switching rule is an important factor in the performance of a switching network.

The optimal switching rule follows: Route a new connection through the network in a way that least affects the routing of any future connections.

Depending on the network topology, such a switching rule may be difficult to implement because of tedious look-ahead iterations. Therefore, rules that are simple to implement and are optimal, or almost optimal, are sought. Beneš describes several switching rules:

> ... route a call through the most heavily loaded part of the network that will still take the call.
>
> Do not use a *fresh* middle switch (in a 3-stage network) unless you have to.[2]

A similar switching rule, used in the simulation program described in Section V, follows: Select the network path that minimizes the count of additional switches whose transmission state must be set.

Such switching rules are called *packing rules* because of their analogy to similar rules used in the problem of packing spheres into boxes. For one module under consideration here, the four switching rules above appear equivalent, and the rule is called *prudent* in the paper. For the other module under consideration here, a case will be demonstrated that represents a counter example to the general adoption of switching rules that recommend close packing as the primary consideration.

2.3 Performance measures

Three measures of the quality of switching networks are transmission, topology, and traffic capability.

2.3.1 Transmission

Two measures of transmission quality are insertion loss and crosstalk. Either measure may be in terms of average or worse-case value. Under a uniform wiring scheme, insertion loss and crosstalk worsen as a network becomes deeper. In electrical implementations, crosstalk usually worsens as interchip, interboard, and especially interframe wiring increase. This should be much less noticeable in photonic systems. Loss and crosstalk measures will not necessarily recommend the same switching networks.

2.3.2 Topology

Topological complexity is not so much a performance issue as it is a manufacturing and economic issue. Quantitative measures that correlate with manufacturing cost are not known, nor is the importance of this issue relative to transmission or traffic measures. It is an important open question.

2.3.3 Traffic capability

Traffic capability is a well-known analytical measure applied to blocking networks.[2] The probability of blocking is that weighted proportion of ⟨new connection, network configurations⟩ in which the given new connection cannot be completed through the network in the given configuration.

A corresponding measure for rearrangeably nonblocking networks is developed here. The *probability of requiring rearrangement* P_{rr} is that weighted proportion of ⟨new connection, network configurations⟩ in which the given new connection can be completed through the network in the given configuration, but only after some established connections in the network are first rearranged.

A dichotomy appears. If the malevolence is the CPU real time used in effecting the rearrangement, then the demerit is that any rearrangement is required, regardless of the count of switches or connections involved, because they are probably all rearranged in parallel. If the malevolence is the count of established connections disturbed by the rearrangement, then the demerit may not be binary and one choice of rearrangement may cost less than another. We will show that the count of established connections disturbed by a rearrangement is different for the two modules under consideration, so the dichotomy is relevant. We will derive expressions for how many established connections must be rearranged, on the average, for each of the competing architectures.

2.4 Traffic analysis techniques

In transient analysis, a sequence of connections and disconnections is applied to an idle module. Over a set of such sequences, P_{rr} is the proportion of those sequences in the set in which a rearrangement was required to complete a connection. The inadequacy of this analysis is that it does not produce a single number or expression, because many sets of sequences must be investigated. But the advantages over the classical analyses are that the results are not dependent on potentially unrealistic assumptions about traffic statistics, and that this analysis entails a level of detail (and corresponding tedium) not required in the other analyses. This detail uncovered an optimal switching rule for one module that probably would have been overlooked had the analysis been confined to traditional steady-state techniques.

In Monte Carlo simulation, a sequence of randomly generated events is applied to an initially idle module. The inadequacies of this analysis are that repeated simulations with identical event statistics yield different results and that no closed-form solution is obtained, giving P_{rr} as a function of those statistics. The benefit of this analysis is that both transient and steady-state behavior may be observed, and in fine detail. If properly recorded, the events leading up to anomalies may be studied.

In steady-state Markov analysis, the module is assumed to be in some random network configuration and then a new random connection or disconnection occurs. P_{rr} is a weighted sum over all network configurations, and all possible new connections from those network configurations, of those cases in which the module requires rearrangement to complete the given connection from the given configuration. The weighting is the steady-state probability distribution over the set of network configurations or states of equivalence classes. Both the steady-state probability distribution and the interstate transition probabilities are dependent on the traffic load. This analysis yields a closed-form result, but it is only valid in the steady state and it is dependent on potentially unrealistic statistical assumptions.

III. PHOTONIC SWITCHING MODULES

We review the technology of photonic switching and present the topologies of two proposed implementations of a 4 × 4 switching module.

3.1 Photonic directional coupler

A photonic directional coupler is a two-input two-output device whose transmission state depends on the magnitude of an applied external voltage.[3] (See Fig. 1.) With nominal 0-volt applied, the transmission is such that the signals cross over from input to output (called the *crossed* state) and with a positive voltage applied, the transmission is straight through from input to output (called the *bar* state). This device is a photonic realization of the generic 2 × 2 *beta switching element*,[4] where the control signal is electronic and the switched data is photonic.

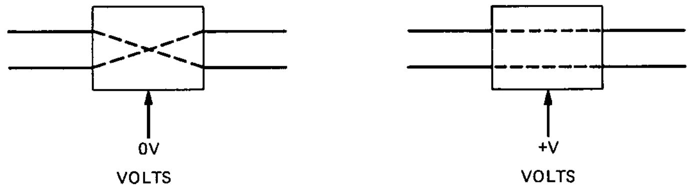

Fig. 1—Functional states of a photonic directional coupler.

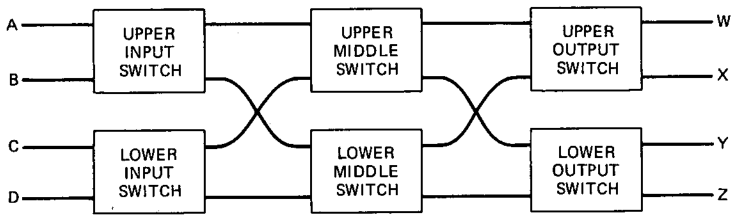

Fig. 2—Configuration and terminology of the 222 Module.

Switching speeds in the tens of picoseconds have been reported,[5] but speeds in the nanoseconds are more common.[6] The attainable switching speed is highly dependent on device packaging, on the quality of the electronic driver circuit that switches the applied voltage to the device, and on the magnitude of this required voltage. A 1:16 multiplexer/demultiplexer has been built as an integrated circuit module.[7] It is a simple tree topology of photonic directional couplers with one unused port at the input to each photonic directional coupler. Another integrated circuit is a 4 × 4 switching module built as a square array.[8,9] These circuits show the level of integration available today in this technology. Because each photonic directional coupler has length in the order of a centimeter, the expectation of increasing the level of integration by several orders of magnitude is not high, unless some major breakthrough changes the photonic coupling length by orders of magnitude.

Two waveguides, made by diffusing titanium in a lithium-niobate substrate, can be made to intersect or cross over. Crosstalk increases as the angle of intersection decreases, and becomes unpredictable at very small angles.[10] Thus a configuration, like that of Fig. 2, is feasible if the crossovers are truly at large angles, as the figure suggests. However, the photonic directional couplers have lengths in the order of a centimeter and widths measured in microns, so Fig. 2 is not drawn to true scale. Furthermore, the loss in a waveguide increases rapidly as its radius of curvature decreases. So, a configuration like that of

Fig. 2 would require a physically large chip to allow geometries with high crossover angles and high radii of curvature.

A major technological issue with today's photonic directional couplers is that they have significant insertion loss. While most of this loss is due to coupling between fiber and chip and would be alleviated with integrated circuits, the device loss is still high enough that multidevice systems, like large switching networks, will require signal amplification. Photonic gain can be simulated, expensively and with imposing a bit-rate constraint, by a circuit with a photodetector, an electronic amplifier, and a laser. However, direct photonic amplification is believed to be coming available soon.[11] The devices described in the reference are fabricated from gallium arsenide, while the photonic directional couplers are fabricated from lithium niobate. Thus, integration on the same chip is currently impossible. However, when such devices become practical, they could be integrated onto the same chip carrier as the chip containing the photonic directional couplers. If this form of integration is truly practical in the future, it allows small radius bends and alleviates many of the problems described in the preceding paragraph.

3.2 The 222 Module

We call one candidate 4 × 4 module the *222 Module* because it is a three-stage module, where each stage has two switches. (See Fig. 2.) This topology is in the class of networks called *Clos networks*, and this exact configuration is a frequently used example in the literature. The following terminology is used to identify symbolic inputs and outputs in the 222 Module:

- A and W represent an arbitrary input and output, respectively, each shown arbitrarily as an upper port on an upper switch.
- B represents the *other* port on the same input switch as A and X represents the *other* port on the same output switch as W.
- C represents either port on the input switch that A and B do *not* share, and Y represents either port on the output switch that W and X do *not* share, each shown arbitrarily as an upper port on a lower switch.
- D represents the *other* port on the same input switch as C, and Z represents the *other* port on the same output switch as Y.

Because of the symmetry of the 222 Module, there is no topological relationship between A and W.

3.3 The 2121 Module

We call the other candidate 4 × 4 module the *2121 Module* because the topology has four stages, where the input and middle stages have two switches and the central and output stages have one switch. (See

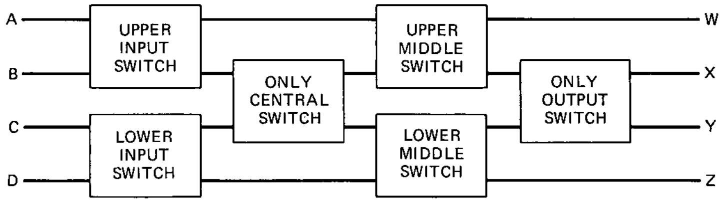

Fig. 3—Configuration and terminology of the 2121 Module.

Fig. 3.) This topology is also a special case of a general network structure.[12] Throughout the paper we will carefully distinguish the terms *middle* and *central* when applied to switching stages. The following terminology is used to identify symbolic inputs and outputs in the 2121 Module:

- A represents an arbitrary input and B represents the *other* port on the same input switch as A, both shown arbitrarily on the upper input switch with A above B.
- W represents the only output that A or B can reach through two stages (necessarily on the same horizontal level as A and B).
- C represents either port on the input switch that A and B do *not* share, and D represents the *other* port on the same input switch as C, both shown arbitrarily on the lower input switch with C above D.
- Z represents the only output that C or D can reach through two stages (necessarily on the same horizontal level as C and D).
- X represents either port on the only switch with two outputs on it, and Y represents the *other* port on the output switch with X, shown arbitrarily with X above Y.

Because the 2121 Module is not symmetric, there is a topological relationship between the inputs and the outputs. Accordingly, W or Z is on the upper or lower middle switches depending on whether A and B or C and D are on the upper or lower input switches, respectively.

3.4 A switching rule for the 2121 Module

An interesting and nonintuitive switching rule was discovered for the 2121 Module. Consider the following sequence of events applied to an idle 2121 Module. Let A to W be the first connection, and let the path that avoids the central switch be selected (intuitively, the best choice), setting the states of A's input switch appropriately, and W's middle switch to the bar state.

If the second connection is B to X, representing two of the nine second connections, the intuitively preferred path is not available because the A to W connection has already set the switches, and is using that junctor that B can reach that skips the central switch. Two other paths are available and the choice is extremely interesting. One path shares the state of A's input switch and W's middle switch, and assigns the central switch to the bar state and the output switch, appropriately. This path requires the assignment of *two* switches in addition to the ones already assigned, and it is intuitively appealing. After A to W is disconnected, A to Y, A to Z, and C to W, representing four of the nine third connections, require rearrangement, and the other five do not.

The other path for B to X still shares the state of A's input switch, but it assigns the central switch to the *crossed* state, the unused middle switch to the bar state, and the output switch appropriately. This path requires the assignment of three switches in addition to the ones already assigned, and is, therefore, intuitively less appealing than the previous path. However, after A to W is disconnected, only A to Z, requiring some of the junctors and switch connections used by B to X, requires a rearrangement of the network configuration.

I have two general interpretations that cover the essence of this switching rule: (1) If you must use the central switch, the crossed state is preferred, and (2) Minimize the count of switches that are shared with any other path. It is not clear yet whether either or both of these statements is applicable to a generalization of the 2121 Module to an $n \times n$ architecture. Even in the 2121 Module, I was afraid that the overall use of this switching rule, while improving the performance with some sequences of events, would degrade the performance with other sequences. In all the sequences investigated and all the simulations that were run, no such case arose.

3.5 Established connections disturbed in the 2121 Module

We give an example of a transient sequence of events applied to the 2121 Module in which the final connection in the sequence requires the disturbance of two established connections. We then argue that the worst-case number could not be greater than two.

The example begins, with an idle module, by connecting A to X. The optimal path avoids the central switch by assigning W's middle switch to the crossed state and A's input switch and the output switch, appropriately. Let the second connection be from B to Y. Since two of the paths are blocked, the remaining path must be used, by assigning the central switch to the crossed state and Z's middle switch to the bar state. Now disconnect the first connection, the one from A to X, freeing the state of W's middle switch.

Let a new second connection be from C to X. Since the output switch and Z's middle switch are in the wrong states to use two of the paths, the remaining path must be used, by assigning W's middle switch to the bar state, and using the idle link through the central switch. The states of all six switches are set, and two of four new connections will require rearrangement.

Consider a third connection from A to Z. Its only path requires the reconfiguration of the input switch shared by A and B and Z's middle switch, and also requires the link through the central switch now used by B to Y. So the B-to-Y connection must be moved to its other (optimal) path. But it can't be moved directly because W's middle switch and the output switch are in the wrong state. So, the C-to-X connection must also be moved to its other (optimal) path.

Having established, by example, that the worst-case number is at least two, the question remains whether it could be higher. Since there can only be four established connections, the only other number to consider is three. However, in any network configuration with three established connections, the unused fourth connection is always available. While this fact may not be immediately obvious, it can be proven exhaustively for the 2121 Module. A more elegant proof, applicable to the general $n \times n$ module, is desirable.

3.6 Comparing the modules

3.6.1 Blocking characteristic

The generalized Clos Network has the topology of Fig. 2, but with n inputs and m junctors on each of r rectangular input switches, with n outputs and m junctors on each of r rectangular output switches, and with m square middle switches each connecting to r junctors on each side. It is known[2] that such a switch is nonblocking if $m \geq 2n - 1$ and is rearrangeably nonblocking if $m \geq n$. The latter is satisfied in the 222 Module, in which $n = m = r = 2$.

The 2121 Module is also rearrangeably nonblocking. This result can be proven easily by exhaustion for the 2121 Module and is obvious by inspection of the state diagram in Section VIII. A general proof for a generalized $n \times n$ network is being developed.[12]

3.6.2 Symmetry and uniformity

In the 222 Module each I/O pair has exactly two connection paths. That is, there are exactly two paths through the module from any input to any output. The 2121 Module does not have this symmetry. The I/O pairs at opposite corners of the module (A to Z, B to Z, C to W, and D to W) have only one path through the module, requiring the central and the appropriate middle switches in the crossed state. The I/O pairs straight across the module (A to W, B to W, C to Z, and D to Z) have two paths through the module: one avoiding the central switch and the other using the central switch in the bar state. Any connection to an output on the only output switch (A to X, B to X, A to Y, B to Y, C to X, D to X, C to Y, and D to Y), has three paths through the module: one avoiding the central switch, the second using the central switch in the bar state, and the third using the central switch in the crossed state.

In the 222 Module each path through the module passes through exactly three photonic directional couplers. The 2121 Module does not have this symmetry, either. In the 2121 Module some paths pass through only two photonic directional couplers, some through three, and some through four. This variation will make deterministic gain difficult. If crossovers in the 222 Module require photonic directional couplers, or have similar loss and crosstalk as photonic directional couplers, then this module would not be symmetric in this sense either.

If crossovers in the 222 Module are insignificant, its inputs and outputs are uniform. That is, all inputs and outputs have the same properties of the following: count of paths per I/O pair, count of switches per path, and equal access to any port on the other side of the module. The last two properties do not hold true if crossovers are significant in the 222 Module, but none hold true in the 2121 Module.

3.6.3 Crossovers

The interconnection topology of the 222 Module has two crossovers: one between the input and central stages and one between the central and output stages. The interconnection topology of the 2121 Module has no crossovers.

An implementation is illustrated in Fig. 4 in which the two crossovers in the 222 Module are eliminated by off-chip fibers. Its practicality depends on the difficulty of off-chip fibering, the impracticality of crossovers on the photonic substrate, and the magnitude of the advantage of the 222 Module over the 2121 Module (if any).

3.6.4 Transmission

It is difficult to predict the difference in the crosstalk characteristics

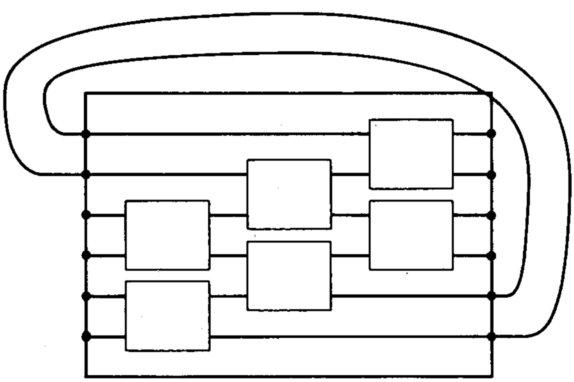

Fig. 4—Off-chip fibering to avoid crossovers in the 222 Module.

of the two modules. The crossovers in the 222 Module suggest that it may be worse than the 2121 Module, but the extra stage in the latter suggests otherwise.

Insertion loss is easier to predict. The worse-case count of switches in a network path is four in the 2121 Module and is three in the 222 Module. Thus, the 2121 Module would appear to have a poorer insertion loss characteristic. However, if crossovers must be implemented by photonic directional couplers, or if they have equivalent loss, then the worse-case count of (equivalent) switches in a network path is five in the 222 Module and it would have worse insertion loss than the 2121 Module.

3.6.5 Switching rule

The switching rule used in the 222 Module is the classical packing rule discussed in Section 2.2. The switching rule used in the 2121 Module is the unusual rule discussed in Section 3.4. Two implementations of the switching rule are discussed in Section IX and the network configurations for each module are enumerated.

3.6.6 The number of established connections disturbed

If some network configuration of the 222 Module must be rearranged before a new connection can be completed, only one established connection need ever be disturbed.[2] It was shown in Section 3.5 that there exist network configurations in the 2121 Module in which two established connections may need to be disturbed. Consequently, we not only compute P_{rr} for each module in the sections below, but we also compute, N_{rr}, how many established connections must be disturbed, on the average.

3.6.7 Familiarity

The 222 Module is intuitive, in so far as the theory of switching networks is intuitive. Its transient behavior is not surprising, except for one set of connect/disconnect sequences, as described in Section IV. The application of switching rules is logical and classical and the plethora of network configurations is easily partitioned into only 10 equivalence classes, or Markov states.

By contrast, the 2121 Module has been counter-intuitive. While its transient behavior is smooth, it degrades with simple connect/disconnect sequences. The switching rule and count of established connections disturbed has been startling to those familiar with such things. The plethora of network configurations is partitioned, with great difficulty, into 50 equivalence classes, making Markov analysis tedious.

IV. TRANSIENT ANALYSIS

In transient analysis, we assume the module is idle and study its behavior under different sets of connect/disconnect sequences. The network quality, the average probability of requiring rearrangement for a specific set of sequences, is the proportion of those sequences in the set that require rearrangement. Four sets are applied to each of the two modules, giving eight results. Computer programs applied all sequences in the four sets to each module and tabulated whether rearrangement was required. The program's output was studied and generalized into theorems whose formal proofs are exhaustive and tedious and not presented in this paper.[13] The proofs are available to the interested reader.

4.1 Sequences of switching events

A *template* is a set of fixed-length sequences of events applied to the switching module, where an *event* is a connection or a disconnection of an input-output pair. The nomenclature for templates is $x_1 \ldots x_n$, where x_i represents the ith event in a sequence; x_i has value C or D, depending on whether this ith event is a connection or disconnection, respectively; and n is the length of a sequence.

The following rules govern the determination of significant templates:

- Since the module is assumed to be idle before any sequence is applied, no sequence would begin with a D.
- Similarly, no sequence, nor initial subsequence, would have more Ds than Cs.
- Since a module can only support four connections, no sequence, nor interior subsequence, would have four more Cs than Ds.
- No significant sequences would have, nor would contain an initial subsequence that has, as many Cs as Ds (e.g., no significant sequence would begin with CD) because it would have the effect of restarting from an idle module.
- Since we expect no problems honoring disconnects in these modules, no significant sequence would end with a D.
- In no significant sequence would a D apply to an immediately preceding C because it would have the effect that the C never occurred.

Templates with length 3 or less exhibit no anomalous behavior, and templates with length 6 or greater are too complex to examine exhaustively. Fortunately, enough templates with lengths 4 and 5 are significant. Combining these length and content constraints, the significant templates are CCCC, CCDC, CCDCC, and CCCDC.

4.1.1 CCCC sequences

For sequences in a CCCC template, beginning with all switches idle, the assumed order of events is

1. One of four inputs connects to one of four outputs.
2. One of three idle inputs connects to one of three idle outputs.
3. One of two idle inputs connects to one of two idle outputs.
4. The last idle input connects to the last idle output.

With $4 \times 4 = 16$ cases of first connection, $3 \times 3 = 9$ cases of second connection, $2 \times 2 = 4$ cases of third connection, and $1 \times 1 = 1$ case of final connection, the CCCC template contains $16 \times 9 \times 4 \times 1 = 576$ sequences.

4.1.2 CCDC(C) sequences

For sequences in a CCDC or CCDCC template, beginning with all switches idle, the assumed order of events is

1. One of four inputs connects to one of four outputs.
2. One of three idle inputs connects to one of three idle outputs.
3. The first input-output pair is disconnected.
4. One of three idle inputs connects to one of three idle outputs.
5. In a CCDCC sequence only, one of two idle inputs connects to one of two idle outputs.

With $4 \times 4 = 16$ cases of first connection, $3 \times 3 = 9$ cases of second connection, 1 case of disconnection, $3 \times 3 = 9$ cases of third connection, and, in the CCDCC sequences only, $2 \times 2 = 4$ cases of fourth connection, the CCDC template contains $16 \times 9 \times 1 \times 9 = 1296$ sequences and the CCDCC template contains $16 \times 9 \times 1 \times 9 \times 4 = 5184$ sequences.

4.1.3 CCCDC sequences

For sequences in a CCCDC template, beginning with all switches idle, the assumed order of events is

1. One of four inputs connects to one of four outputs.
2. One of three idle inputs connects to one of three idle outputs.
3. One of two idle inputs connects to one of two idle outputs.
4. Either of the first two input-output pairs is disconnected.
5. One of two idle inputs connects to one ot two idle outputs.

With $4 \times 4 = 16$ cases of first connection, $3 \times 3 = 9$ cases of second connection, $2 \times 2 = 4$ cases of third connection, 2 cases of disconnection (either the first or second connection), and $2 \times 2 = 4$ cases of final connection. The CCCDC template contains $16 \times 9 \times 4 \times 2 \times 4 = 4608$ sequences.

4.2 Results

In either module, with a prudent switching rule, no CCCC sequences require rearrangement. Since all combinations of idle I/O pairs can be simultaneously interconnected in both modules, they are both at least rearrangeably nonblocking.

The 222 Module outperforms the 2121 Module under CCDC and CCDCC sequences. In the 222 Module, with a prudent switching rule, no CCDC sequences, nor CCDCC sequences, require rearrangement. However, in the 2121 Module with its unusual switching rule, 2 percent of the CCDC sequences require rearrangement and 6 percent of the CCDCC sequences require rearrangement.

Conversely, the 2121 Module outperforms the 222 Module under CCCDC sequences. In the 222 Module, with a prudent switching rule, 8 percent of all CCCDC sequences require rearrangement. However, in the 2121 Module, with its unusual switching rule, only 6 percent of all CCCDC sequences require rearrangement. Since neither module is generally nonblocking but both are at least rearrangeably nonblocking, they are identically rearrangeably nonblocking.

Summarizing transient analysis, Fig. 5 illustrates module performance versus template complexity for the two modules. The scale of the X axis has no mathematical nor physical meaning. The 222 Module outperforms the 2121 Module under all tested templates up to and including CCDCC sequences, but is outperformed by the 2121 Module under CCCDC sequences. I expected the 222 Module to be consistently better, so I was surprised by this turn of events. A qualitative explanation of this unusual behavior is elusive, but I propose a conjecture.

The 2121 Module is distorted from its optimal connectivity by simple sequences of events, because of its asymmetric junctor pattern and nonuniform switch-count in alternate paths. These distortions usually

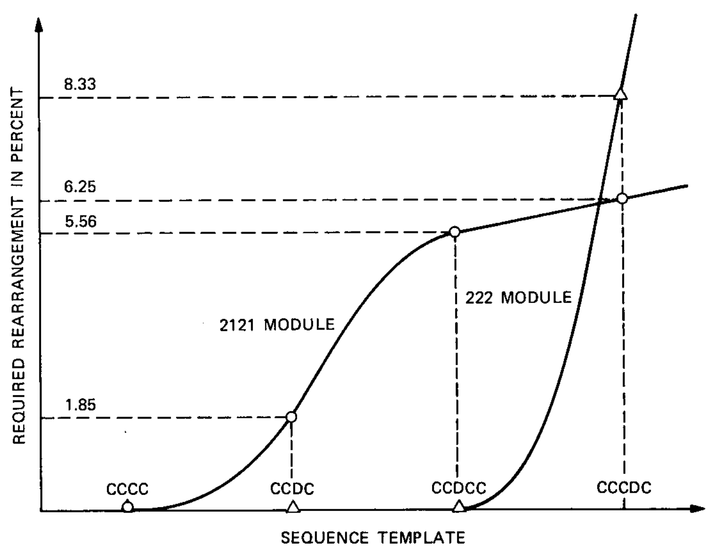

Fig. 5—Percentage of sequences requiring rearrangement versus template.

involve connections, like A to Z, that go diagonally across the module and have only one path. The severity of these distortions is softened by the presence of I/O pairs that have three paths, and by the opportunity to be clever in the choice of switching rule. The 222 Module, however, is not distorted from optimal connectivity until the event sequences are more complex. When it finally becomes distorted, it is more severely distorted than the 2121 Module.

V. MONTE CARLO SIMULATION

In a simulation, a sequence of randomly generated events is applied to an initially idle module. In this section, the results of simulation program are plotted in a scatter diagram and interpreted.

5.1 Program description

Two distinct programs were written—one for each module. The programs are, however, similar at a high level of description. Program variables record switch states, paths assigned to connections, interconnected parties, and other parameters and outputs.

The program begins by initializing a random number generator and reading module connectivity information from files. The program then enters a loop on traffic intensity: varying from 1.1 to 3.8 in increments of 0.3. Since the double exponential model of traffic is simulated, the average holding time and the average quiet time of each input are computed from the traffic intensity.

Rather than simulate the Poisson environment, by stepping through small intervals of time and simulating events, the program skips through time from one event to the next. Corresponding to each input i, is the time to the next event $tne[i]$, associated with that input. These values are initialized to an exponentially distributed random number whose mean is the quiet time. The number of rearrangements is initialized to zero and an interior loop, on the number of events in the simulation, is entered.

Time is advanced to the minimum value of the four $tne[i]$ and all $tne[i]$ are reduced by that value. The event that timed out is simulated. If it was a holding time that expired, the connection associated with the relevant input is disconnected and $tne[i]$ for that input is set to a random quiet time. If it was a quiet time that expired, a connection is made with the relevant input.

The connection is established by first locating a random idle input. A subroutine that implements the switching rule determines the best available path through the module. If there is no available path, the count of rearrangements is incremented, and established connections are disconnected and reconnected by their best paths. The connection associated with the event is established and *tne*[*i*] for that input is set to a random holding time.

The loop on events terminates, and after printing out results as a function of traffic intensity, the loop on traffic intensity terminates.

5.2 Results

Simulations were run with sequences of 1000 and 5000 events, for the 222 and 2121 Modules, under varying traffic load. These numbers of events should give statistically significant results, and should be large enough that the transient effects from starting idle would be damped out. The independent parameter is the mean of *traffic intensity*, the product of global rate-of-origination by per-call holding time, how many simultaneous connections exist in a module at any time. The dependent variable is the percentage of new connections requiring a rearrangement. Traffic intensity is varied from slightly over 1.0 to slightly under 4.0, and the result of each simulation is plotted as a scatter diagram in Fig. 6. Simulation results for the 222 Module are shown with × and for the 2121 Module with +. Results of simulations with 5000 events are circled and with 1000 events are not.

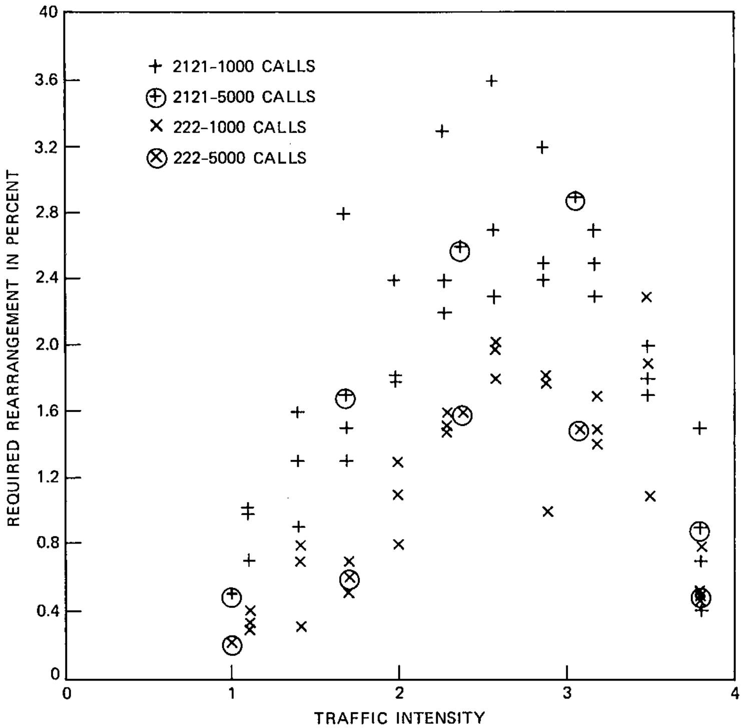

Fig. 6—Scatter diagram of P_{rr} versus traffic.

5.3 Interpretation

At low and moderate traffic intensities, it is noted that P_{rr} gets worse as the traffic intensity increases and that P_{rr} is uniformly, but slightly, worse in the 2121 Module than in 222 Module. In neither case is P_{rr} particularly bad, even at their maxima. That there is a maximum is somewhat surprising. A monotonically increasing P_{rr} might have been expected.

At high traffic load, there will be times when a 4 × 4 module is completely connected. The next event would have to be the disconnection of one I/O pair, and the event after that would be likely to be the reconnection of the same pair. It would be likely to be a connection because the traffic load is high, and it would have to be the same pair because they are the only idle input and output, the only connection that could be made. Such a connection would never require rearrangement because it was just disconnected. This behavior is consistent, whether the module is isolated as a simple 4 × 4 network or part of a large network.

A familiar result in classical statistics comes from applying Chebyshev's inequality to Bernoulli trials. This Bernoulli law of large numbers is used, for example in determining sample sizes of public opinion polls:

$$P(|f_n - p| > \epsilon) \leq p(1-p)/n\epsilon^2,$$

where f_n is the observed frequency after n trials, p is the given or assumed probability, and ϵ is an arbitrary tolerance. For simulations with $n = 1000$ events and $p = 0.03$, the inequality states that the variation in the outcome should be within ±3 percent for 97 percent of the simulations. The tolerance is even less with smaller values of p. A casual glance at Fig. 6 shows that the variation is much greater than this. A conjectured explanation is that law of large numbers is derived from the assumption that the Bernoulli trials are independent. Since P_{rr} in the ith event of a network simulation is highly dependent on previous events, this fundamental assumption is invalidated.

VI. MARKOV ANALYSIS OF A GENERALIZED MODULE

Three analyses were performed. The first, presented in the remainder of this section, is of a generalized nature and pertains to both modules. It establishes a general model to be used as a check for the models of the modules under investigation. The second analysis, in Section VII, is based on a Markov model of the operation of the 222 Module. The third analysis, in Section VIII, is based on a Markov model of the operation of the 2121 Module.

6.1 The generalized model

In this oversimplified model, any network configurations in which the same count of connections are established are deemed to be equivalent. There are five equivalence classes of network configurations, corresponding to zero to four established connections, inclusive. Hence, there are five states in the corresponding Markov process, shown in Fig. 7. The model is general enough to cover both modules.

The stochastic behavior of the model is based on the classical traffic assumptions: Poisson-distributed service arrivals and exponentially

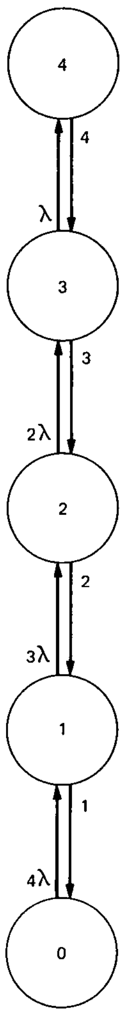

Fig. 7—Markov-queuing model.

distributed holding times. Since the Poisson and exponential distributions are mutual inverses, this model is equivalent to having Poisson-distributed arrivals of connections and disconnections or exponentially distributed off-hook and on-hook times (as used in the simulation program). Either way of looking at it, there are two random variables and there are two underlying parameters: the busyness and the true time scale. If results are considered per unit time or limited to steady-state behavior, then true time scale is irrelevant, and one random variable may be arbitrary and the other specifies busyness per unit time. We will use the double-Poisson process, for this and the two later models, and arbitrarily set the per-line disconnect rate to 1.

The state transitions are computed from the individual connect and disconnect rates of the four inputs (or outputs, equivalently), which are assumed to be statistically identical. Let λ be the rate at which each individual input requests a connection through the module, and let 1.0 be the rate at which each individual established connection is disconnected. In other words, λ is the ratio of the connect rate to the disconnect rate of each individual input (or output). The global rate of new connections from state i is $(4 - i)\lambda$, because there are $4 - i$ idle inputs that could make such a request. The global rate of disconnections from state i is i, because there are i established connections, any of which could request disconnection.

The analysis is a special case of a queue with dependence on the state of the system,[14] but can also be viewed as a queue with finite customer population and infinite servers.[15]

6.2 Steady-state probabilities

The steady-state probabilities are calculated from a conservation law. In the steady state, the mean exit rate from state i is the sum of the rates on all exit arcs from state i times the steady-state probability of being in state i. The mean entry rate into state i is the sum of the products of a rate on each entry arc times the steady-state probability of being in the state from which the arc comes. Conservation of calls dictates that the mean exit rate must equal the mean entry rate in the steady-state for each state. The resulting simultaneous equations have the general solution:[14]

$$g_n = (\Lambda_0 \cdots \Lambda_{n-1})g_0/(\mu_1 \cdots \mu_n),$$

where $n > 0$, Λ_i is the global connect rate on the arc from state i to state $i + 1$, μ_i is the global disconnect rate on the arc from state i to state $i - 1$, and g_i is the steady-state probability of being in state i. Setting $\Lambda_i = (4 - i)\lambda$ and $\mu_i = i$ gives

$$g_1 = (4\lambda)g_0/1 = 4\lambda g_0$$

$$g_2 = (4\lambda)(3\lambda)g_0/2 = 6\lambda^2 g_0$$

$$g_3 = (4\lambda)(3\lambda)(2\lambda)g_0/(2 \times 3) = 4\lambda^3 g_0$$

$$g_4 = (4\lambda)(3\lambda)(2\lambda)\lambda g_0/(2 \times 3 \times 4) = \lambda^4 g_0.$$

Note that each g_i is expressed as a function of g_0. The probability g_0 that there are no connections is found by setting the sum of all steady-state probabilities to 1. This gives the neat result

$$\sum g_i = (1 + 4\lambda + 6\lambda^2 + 4\lambda^3 + \lambda^4) \times g_0 = 1$$

$$g_0 = 1/(1 + \lambda)^4.$$

The steady-state probability mass function is

$$G(\lambda) = [1, 4\lambda, 6\lambda^2, 4\lambda^3, \lambda^4]/(1 + \lambda)^4.$$

6.3 Traffic intensity

Traffic intensity is a random variable giving the count of established connections at any time. In the classical traffic model, having Poisson arrivals with rate r and exponential holding time with mean h, traffic intensity is known to be Poisson distributed with mean, $\tau = rh$.

One way of looking at the model of Fig. 7 is that the system has four sources, each generating one calling cycle per unit time. The calling cycle consists of an exponentially distributed off-hook interval with mean $\lambda/(1 + \lambda)$ and an exponentially distributed on-hook interval with mean $1/(1 + \lambda)$. Setting the arrival rate to 4 and the holding time to the mean off-hook interval, the Poisson-distributed traffic intensity has mean $\tau = 4 \times \lambda/(1 + \lambda)$. This intuitive argument is verified by computing the mean count of established connections in the steady state

$$\tau = \sum (i \times g_i) = (0 \times 1 + 1 \times 4\lambda + 2 \times 6\lambda^2 + 3 \times 4\lambda^3 + 4 \times \lambda^4)/(1 + \lambda)^4$$

$$= 4\lambda(1 + 3\lambda + 3\lambda^2 + \lambda^3)/(1 + \lambda)^4$$

$$= 4\lambda/(1 + \lambda).$$

The inverse of this expression will prove useful later on:

$$\lambda = \tau/(4 - \tau).$$

6.4 Examples

Consider the two extremes. If $\lambda = 0$, the steady-state probability mass function is

$$G(0) = [1, 0, 0, 0, 0],$$

and the mean traffic intensity $\tau = 0$. If $\lambda \rightarrow \infty$, the limit of the steady-state state probability mass function is

$$G(\infty) \rightarrow [0, 0, 0, 0, 1],$$

and the limit of the mean traffic intensity $\tau \rightarrow 4$. As a better example, let $\lambda = 1$, which means that the individual arrival rate equals the departure rate, or that each input is on-hook for the same average time as off-hook. The steady-state probability mass function is

$$G(1.0) = [1, 4, 6, 4, 1]/16,$$

showing a trend toward state 2, with decreasing probability in either direction away from state 2. The symmetry about state 2 suggests an average of two connections in the steady state and setting $\lambda = 1$ in the equation for traffic intensity gives $\tau = 2$.

As another example, let $\lambda = 2.0$, meaning that each input is off-hook twice the time that it is on-hook, that is, two-thirds of the off-hook/on-hook cycle. The steady-state probability mass function is

$$G(0.5) = [1, 8, 24, 32, 16]/81,$$

showing a trend slightly under state 3 and a lack of the symmetry observed in the case where $\lambda = 1$. If $\lambda = 0.5$, the steady-state state probabilities are reversed from the case where $\lambda = 2$. The mean traffic intensity in these cases is 8/3 and 4/3, respectively, representing the center of mass for each distribution.

VII. MARKOV ANALYSIS OF THE 222 MODULE

7.1 The model

The state model of the 222 Module is illustrated in Fig. 8.[16] Each bubble in the figure represents a state and contains the state's name and a representative network configuration from the state's equivalence class. State I represents the idle network configuration and is equivalent to state 0 in the generalized model. State J represents the 16 network configurations with one connection established. These 16 network configurations are all equivalent, for purposes of determining P_{rr}, and this state is equivalent to state 1 in the generalized model.

States S through V represent four equivalence classes of network configurations in which two connections are established. This set of states is equivalent to state 2 in the generalized model. In state S, the two established connections terminate on the same input switch and on the same output switch. In states T and U, the two established connections terminate on opposite input switches and on opposite output switches. In state T, both middle switches are used and in state

U, one middle switch is shared by both established connections and the other is idle. State T is the only malevolent state in the model and the only transition in which rearrangement is required is the one from state T to state X. In state V, either the two established connections terminate on the same input switch and on opposite output switches, or the two established connections terminate on opposite input switches and on the same output switch. These conditions are equivalent for purposes of computing P_{rr}.

States W and X represent two equivalence classes of network configurations in which three connections are established. These two states are equivalent to state 3 in the generalized model. In state W, two of the established connections terminate on the same input switch and on the same output switch, and the third established connection terminates on the other input switch and the other output switch. In state X, the two established connections that terminate on the same input switch terminate on opposite output switches, and the two established connections that terminate on the same output switch terminate on opposite input switches. These are the only distinctions that need be made among all network configurations with three connections established.

States Y and Z represent two equivalence classes of network configurations in which four connections are established. These two states are equivalent to state 4 in the generalized model. In state Y, two established connections that terminate on the same input switch terminate on the same output switch. In state Z, two established connections that terminate on the same input switch terminate on opposite output switches. These are the only distinctions that need be made among all network configurations with four connections established.

7.2 State transitions

Corresponding to the transition from state I to J is a connection through the module in which two paths are possible. The choice of path is arbitrary and in no way affects the results. Corresponding to the transitions from state J to S or V is a second connection that has only one available path. Corresponding to the transition from state J to U, is a connection in which the selected path shares a middle switch with the established connection. Selecting the other path would correspond to an imprudent transition to state T.[16] Corresponding to transitions from states T, U, and V to states W and X are third connections that have only one available path, even with the rearrangement required before the connection corresponding to the transition from state T to X. Corresponding to the transition from state S to W, is a third connection in which two paths are possible. The choice is arbitrary, not affecting the results, but the choice determines that established connection, which if later disconnected, would correspond to the undesired transition to state T. Corresponding to the transitions from states W or X to states Y or Z, respectively, are fourth connections in which only one path is available. The transition from state J to U represents the only connection where the switching rule is relevant.

The rates on the arcs connecting the states in Fig. 8 are similar to those in the generalized model, but several need further explanation.

- The sum of the connect rates on transitions from state J to states S through V must be 3λ, corresponding to the connect rate on the transition from state 1 to 2 in the generalized model. With one established connection, there are nine possible second connections: one that terminates on the same input and output switches as the established connection, four that terminate on the opposite input and output switches as the established connection, and four that terminate on one same switch and one opposite switch as the established connection. Thus the rates on the transitions to states S, U, and V are $3\lambda \times 1/9$, $3\lambda \times 4/9$, and $3\lambda \times 4/9$, respectively. The switching rule would prevent a direct transition from state J to state T; state U always being preferred over state T.
- The sum of the connect rates on transitions from each of states S through V must be 2λ, corresponding to the connect rate on the transition from state 2 to 3 in the generalized model. In states S and V, all third connections lead to the same next state, state W or X, respectively, and so each single transition is labeled with 2λ. In states T and U, however, half the third connections terminate on the same input and output switches as an existing connection, and half terminate on the same input switch as one connection and the same output switch as the other connection. Thus, states T and U each transit to both states W and X, and the rates on those transitions are all λ. All third connections associated with transitions from state T to state X require rearrangement, and these are the only connections requiring rearrangement in the entire model.[16] Either of the two established connections may be rearranged so that one middle switch is shared by both connections and one is idle—an intermediate network configuration that conforms to state U.
- The sum of the disconnect rates on transitions from each of states W and X must be 3, corresponding to the disconnect rate on the transition from state 3 to 2 in the generalized model. In any network configuration belonging to state X, two of the three single disconnections results in a configuration belonging to state V, so that arc is labeled with a rate of 2. The other single disconnection results in a configuration belonging to state U, so that arc is labeled with a

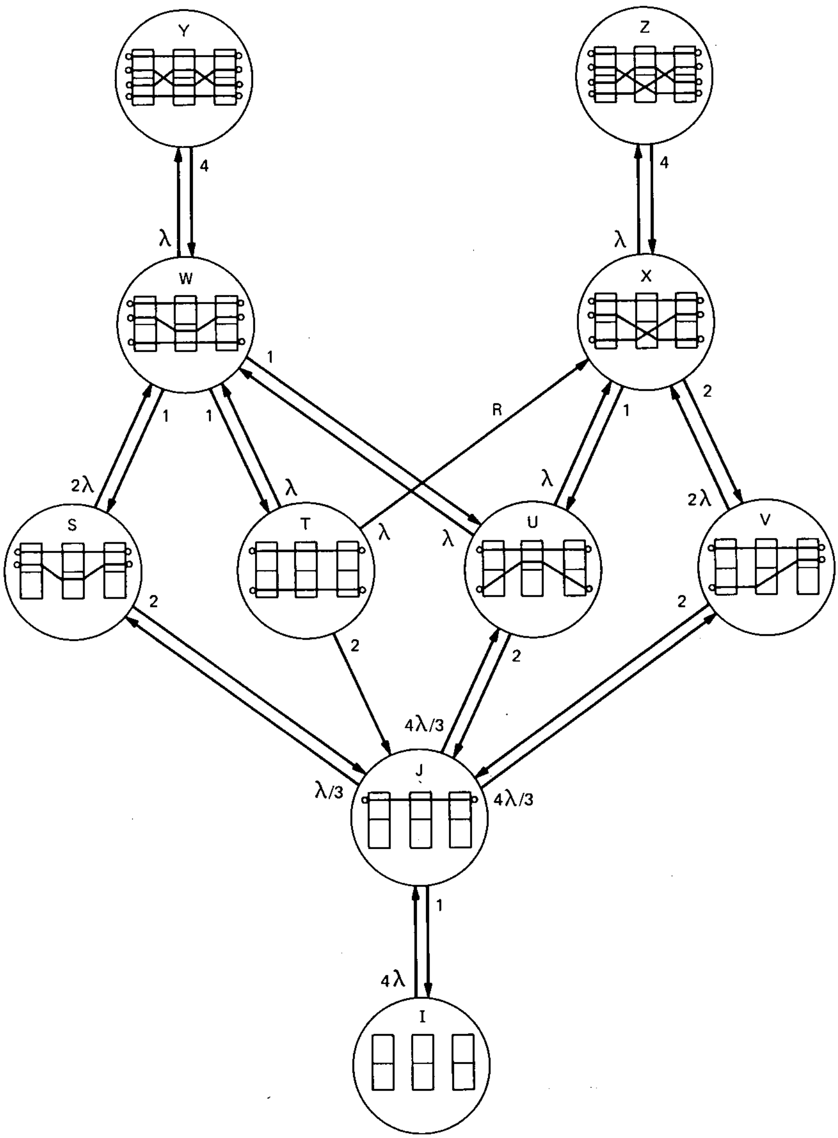

Fig. 8—Markov model for the 222 Module.

rate of 1. In any network configuration belonging to state W, a transition to a configuration belonging to state V is impossible, but each of the three single disconnections results in a configuration belonging to state S, T, or U, respectively. Thus, each of these transitions is labeled with a rate of 1. The transition from state W to T is the only entry into that malevolent state, and it is unavoidable under any prudent switching rule.

The rates on the state transitions and the notion of λ in this development are slightly different from that of the reference.[16] In that paper, λ represented the rate of origination of connections between specific inlet and outlet pairs. In this development, traffic is assumed to originate at an inlet, and terminate on any random idle outlet.

7.3 Steady-state probabilities

The rate equations for each of the ten states are

$$4\lambda p_{\rm I} = p_{\rm J}$$

$$(3\lambda + 1)p_{\rm J} = 4\lambda p_{\rm I} + 2(p_{\rm S} + p_{\rm T} + p_{\rm U} + p_{\rm V})$$

$$(2\lambda + 2)p_{\rm S} = (\lambda/3)p_{\rm J} + p_{\rm W}$$

$$(2\lambda + 2)p_{\rm T} = p_{\rm W}$$

$$(2\lambda + 2)p_{\rm U} = (4\lambda/3)p_{\rm J} + p_{\rm W} + p_{\rm X}$$

$$(2\lambda + 2)p_{\rm V} = (4\lambda/3)p_{\rm J} + 2p_{\rm X}$$

$$(\lambda + 3)p_{\rm W} = 2\lambda p_{\rm S} + \lambda p_{\rm T} + \lambda p_{\rm U} + 4p_{\rm Y}$$

$$(\lambda + 3)p_{\rm X} = \lambda p_{\rm T} + \lambda p_{\rm U} + 2\lambda p_{\rm V} + 4p_{\rm Z}$$

$$4p_{\rm Y} = \lambda p_{\rm W}$$

$$4p_{\rm Z} = \lambda p_{\rm X}.$$

As is customary with such processes, only $n - 1$ of the n equations are independent. Setting the sum of the n probabilities to 1 provides the nth independent equation. The solution, after several hours of manual algebra is

$$p_{\rm I} = 1/(1 + \lambda)^4$$

$$p_{\rm J} = 4\lambda/(1 + \lambda)^4$$

$$p_{\rm S} = 2\lambda^2/3(1 + \lambda)^4$$

$$p_{\rm T} = 2\lambda^3/3(1 + \lambda)^5$$

$$p_{\rm U} = \lambda^2(8 + 6\lambda)/3(1 + \lambda)^5$$

$$p_{\rm V} = 8\lambda^2/3(1 + \lambda)^4$$

$$p_{\rm W} = 4\lambda^3/3(1 + \lambda)^4$$

$$p_{\rm X} = 8\lambda^3/3(1 + \lambda)^4$$

$$p_{\rm Y} = \lambda^4/3(1 + \lambda)^4$$

$$p_{\rm Z} = 2\lambda^4/3(1 + \lambda)^4.$$

As a check, it is verified that p_I above equals g_0 from the generalized model, $p_J = g_1$, $p_S + p_T + p_U + p_V = g_2$, $p_W + p_X = g_3$, and $p_Y + p_Z = g_4$. Of particular interest, of course, is the steady-state probability of the malevolent state, p_T.

7.4 Probability of requiring a rearrangment

P_{rr} is the proportion of those new connections that require that an (one) established connection be rearranged before a new connection may be completed. The numerator is the sum over all states of ⟨the average count of new connections requiring rearrangement from state i⟩ × ⟨the steady-state probability of state i⟩. For this network, it is simply $1 \times p_T$. The denominator is the weighted average count of possible new connections, similar to the calculation of τ in the previous section

$$\sum(4 - i) \times g_i = (4 \times 1 + 3 \times 4\lambda + 2 \times 6\lambda^2 + 1 \times 4\lambda^3 + 0 \times \lambda^4)/(1 + \lambda)^4$$

$$= 4(1 + 3\lambda + 3\lambda^2 + \lambda^3)/(1 + \lambda)^4$$

$$= 4/(1 + \lambda).$$

The ratio is then

$$P_{rr} = [2\lambda^3/3(1 + \lambda)^5] \div [4/(1 + \lambda)] = \lambda^3/6(1 + \lambda)^4,$$

or, as a function of traffic intensity,

$$P_{rr} = \tau^3(4 - \tau)/1536.$$

The curve for this expression is plotted in Fig. 9 through the data

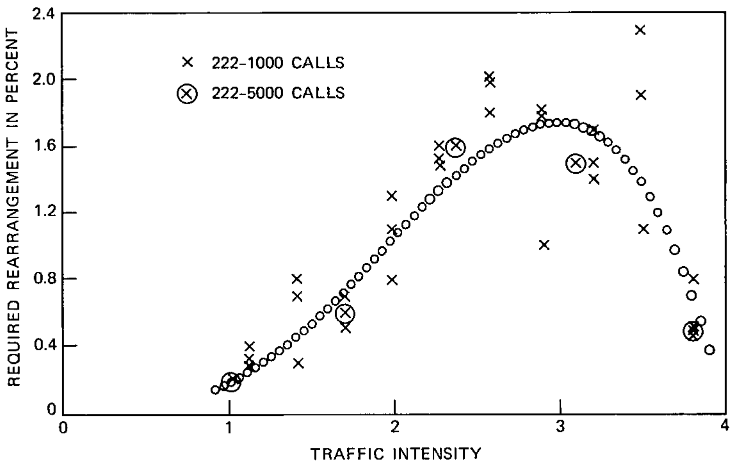

Fig. 9—Simulation results and derived curve for the 222 Module.

points for the 222 Module, taken from the scatter diagram of Fig. 6. Setting the derivative of the expression above to zero yields an extremum at $\lambda = \tau = 3$, and the value at that maximum is 1.76 percent.

Related to this figure is N_{rr}, the average count of established connections that must be rearranged when a new connection is completed. Since exactly one established connection is rearranged when the 222 Module requires rearrangement,

$$N_{rr} = P_{rr} = \lambda^3/6(1 + \lambda)^4 = \tau^3(4 - \tau)/1536.$$

VIII. MARKOV ANALYSIS OF THE 2121 MODULE

8.1 The model

Since the Markov model for the 2121 Module has 50 states, a natural nomenclature is a mapping to a familiar set with 50 elements, also called "states." The mapping is shown in Fig. 10, where each U. S. state represents a set of equivalent network configurations from the 2121 Module. A representative configuration is shown with each state in Fig. 10. The states are grouped according to the number of established connections, or their relationship to states in the generalized model. FL represents the idle Markov state and the six New England states represent six Markov states in which all four inputs connect to all four outputs. States in three intermediate east-to-west bands across the U. S. A. represent Markov states with one, two, and three established connections, respectively. To avoid clutter in Fig. 10, the transitions among the states are shown in later figures.

8.2 State transitions

The upward transitions in the model for the 2121 Module, in which no rearrangements are required, are shown in Fig. 11. Consider the three upward transitions from FL. The transition from FL to GA represents establishing, in an originally idle module, an A-to-Z (or C-to-W) connection diagonally across the module. This connection has only one path through the module, and that path requires the central switch in the crossed state. By contrast, A-to-W (or C-to-Z) connections have two paths through the module, represented by NM and AZ. A direct upward transition from FL to AZ, and none from FL to NM, demonstrates the preference for the path that avoids the central switch. Similarly, the direct upward transition from FL to AL, and lack of same to LA or TX, demonstrates the choice of the path for an A-to-X connection that avoids the central switch over the other two paths. Similar logic governs the other upward transitions on the graph. For simplicity, the weights on the arcs are not shown, but they are similar to those of the 222 Module.

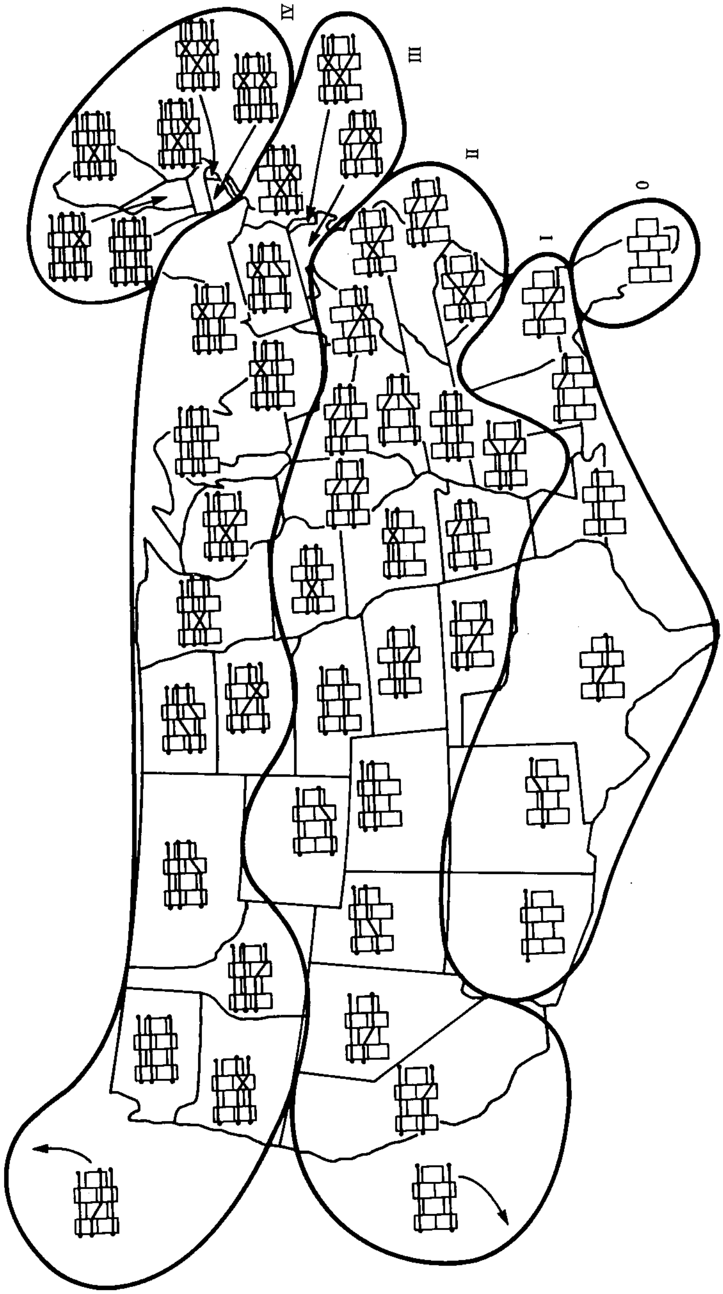

Fig. 10—Markov states and nomenclature for the 2121 Module.

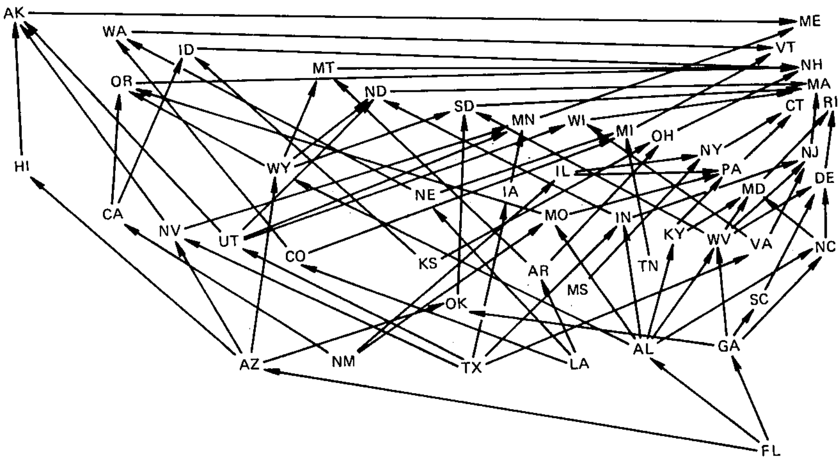

Fig. 11—Direct upward transitions in the 2121 Model.

The upward transitions, in which one or two established connections require rearrangement, are shown in Fig. 12. The three states with one established connection from which rearrangements may be required—LA, TX, and NM—have no direct upward transitions from FL, while the only three states that can be reached directly from FL—GA, AL, and AZ—have no upward transitions requiring rearrangement. Two of the states with two established connections from which any rearrangements may be required—TN and KS—have no upward transitions from any state with one established connection. The other five states with two established connections from which any rearrangements may be required—VA, IL, AR, IA, and NE—have upward transitions from states with one established connection, but only from LA, TX, and NM, the states that cannot be reached directly from FL. The transitions from IA to MD, KS to SD, and KS to ND are particularly interesting because both established connections require rearrangement before the new connection, corresponding to the transition, can be completed. Since the states from which rearrangement is required are reached only through complex sequences of connection and disconnection, we expect P_{rr} to be small.

The downward transitions in the model for the 2121 Module are shown in Fig. 13. The rates on all the arcs connecting the states, as in the two previous figures, are not shown.

8.3 Steady-state probabilities

The rate equations for each of the 50 states are given in Appendix A. A closed-form solution to this set of equations has not been found, but approximations were derived by a simple, and tedious, iteration.

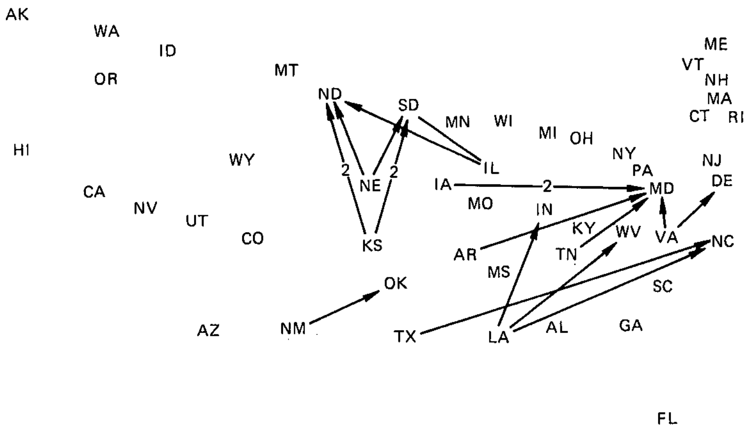

Fig. 12—Transitions requiring rearrangement in the 2121 Model.

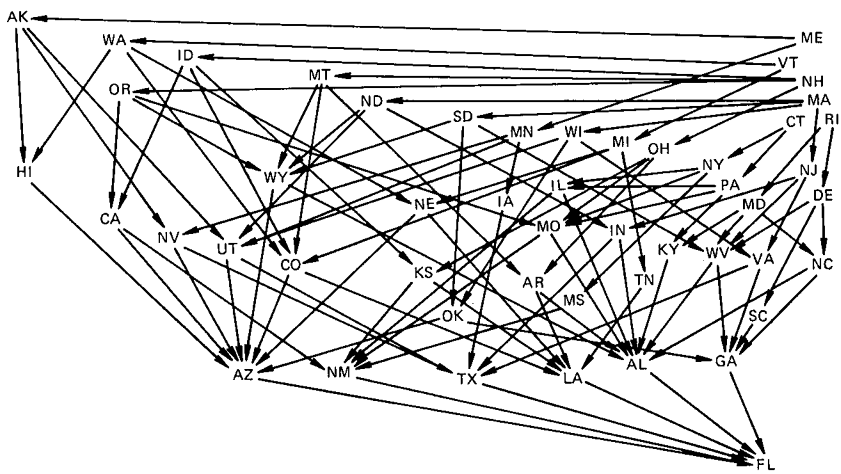

Fig. 13—Downward transitions in the 2121 Model.

The states are initially distributed equilikely within their equivalence class such that the sum of the state probabilities within each class equals the corresponding state probability in the generalized model. Thus,

$$q_{\mathrm{FL}} = 1/(1+\lambda)^4,$$

$q_x = 4\lambda/6(1+\lambda)^4$ for the 6 states: GA to AZ,

$q_x = 6\lambda^2/21(1+\lambda)^4$ for the 21 states: NC to HI,

$q_x = 4\lambda^3/16(1+\lambda)^4$ for the 16 states: DE to AK,

$q_x = \lambda^4/6(1+\lambda)^4$ for the 6 New England states.

Then, temporarily deleting $(1 + \lambda)^4$ from every denominator, these expressions were substituted into the right side of the rate equations in Appendix A. The resulting intermediate expressions for each q_x are polynomials in λ divided by the $(a + b\lambda)$ term on the left side of the corresponding rate equation. The long division was effected and the quotient was truncated to a simple polynomial in λ, a new expression for each q_x. Most of these new expressions had coefficients between double and half their analogs in the original expressions.

These new expressions were then substituted into the rate equations, and a similar simplification and approximation was effected. By the third iteration, the expressions were surprisingly close to those of the second iteration, and rapid convergence was observed. These approximations to the steady-state probabilities, without the long division and quotient truncation in the final iteration, are given in Appendix B. Of particular interest, again, are the steady-state probabilities of the ten malevolent states.

8.4 Probability of requiring a rearrangement

P_{rr} is the proportion of those new connections that require rearrangement of an (at least one) established connection before the new connection may be completed. The numerator is the sum over all states of ⟨the average count of new connections requiring rearrangement from state i⟩ × ⟨the steady-state probability of state i⟩. For the 2121 Module, the numerator is

$$(4/3) \times q_{LA} + (1/3) \times q_{TX} + (1/3) \times q_{NM} + 1 \times q_{VA} + 1 \times q_{TN}$$
$$+ 1 \times q_{IL} + 1 \times q_{AR} + 1 \times q_{IA} + 1 \times q_{KS} + 1 \times q_{NE}$$
$$= (2.1\lambda^4 + 4.8\lambda^3 + 1.0\lambda^2 - 1.1\lambda)/(1 + 3\lambda)(1 + \lambda)^5,$$

using the approximate expressions from Appendix B. The denominator, $4/(1 + \lambda)$, is the weighted average count of possible new connections, as in the calculation from the previous section. The ratio is then

$$P_{rr} = (0.5\lambda^4 + 1.2\lambda^3 + 0.2\lambda^2 - 0.3\lambda)/(1 + 3\lambda)(1 + \lambda)^4$$

or, as a function of traffic intensity,

$$P_{rr} = (0.5\tau^4(4 - \tau) + 1.2\tau^3(4 - \tau)^2 + 0.2\tau^2(4 - \tau)^3$$
$$- 0.3\tau(4 - \tau)^4)/512(2 + \tau)$$
$$= \tau(\tau - 4)(\tau^3 + 2\tau^2 - 88\tau + 96)/2560(2 + \tau).$$

The curve for this expression is plotted in Fig. 14 through the data points for the 2121 Module, taken from the scatter diagram of Fig. 6. Setting the derivative of the expression above to zero yields an extremum at $\tau = 2.6$, and the value at that maximum is 3.1 percent.

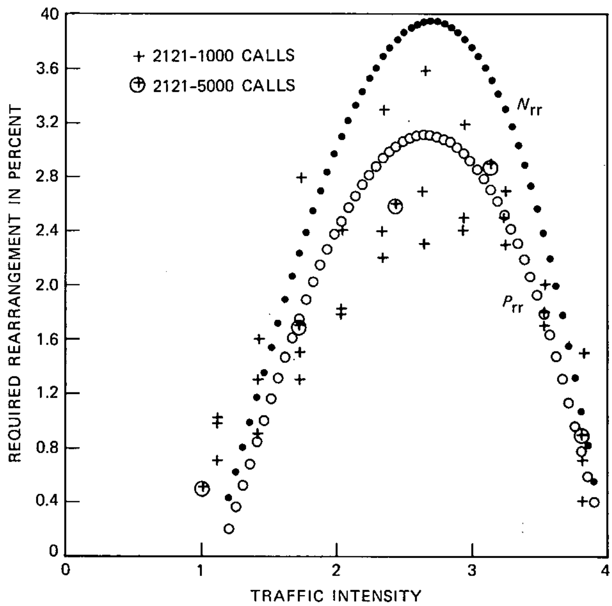

Fig. 14—Simulation results and derived curve for the 2121 Module.

Related to this figure is, N_{rr}, the average count of established connections that require rearrangement when a new connection is completed. Since this count is two on three of the state transitions, this figure is not equal to P_{rr}:

$$N_{rr} = [(4/3) \times q_{LA} + (1/3) \times q_{TX} + (1/3) \times q_{NM} + 1 \times q_{VA} + 1 \times q_{TN} + 1 \times q_{IL} + 1 \times q_{AR} + 2 \times q_{IA} + 2 \times q_{KS} + 1 \times q_{NE}]/[4/(1 + \lambda)]$$

$$\cdot N_{rr} = \tau(\tau - 4)(\tau^3 - 2\tau^2 - 88\tau + 96)/2560(2 + \tau).$$

This expression is also plotted in Fig. 14. We observe that the Markov analysis of such an innocent-looking network is surprisingly complex.

IX. NETWORK CONFIGURATIONS

Two algorithms for a switching rule are by direct calculation or by table look-up. The direct calculation of the switching rule for either module would be time-consuming, with the switching rule for the 2121 Module significantly more complicated than that for the 222 Module. The count of connection combinations in either module is also large, with little difference between the two modules, so a table look-up implementation of the switching rules would require a large ROM.

Two distinct realizations of such an algorithm are by a microcontroller per module or by a common controller governing all modules in a network. Direct calculation in a common controller of a large network suggests a real-time bottleneck and table look-up in a per-module algorithm suggests considerable replication of a large ROM. Therefore, the logical choices are distributed control by direct calculation in a per-module microcontroller, or centralized control by table look-up. The count of network configurations in a table look-up algorithm is discussed in this section.

9.1 Count of 222 Module configurations

The Markov diagram of the 222 Module is repeated in Fig. 15, except that an additional number is placed in each bubble, the count of unique network configurations that are represented by the Markov state. State I represents the only idle network configuration. Considering configurations with a single established connection, since any of four inputs can be connected to any of four outputs, and each connection has two paths through the network, state J represents $4 \times 4 \times 2 = 32$ network configurations.

States S through V represent all configurations with two established connections. A configuration in state S derives from a configuration in state J by connecting the only other input on the same input switch as the established connection to the only other output on the same output switch as the established connection by the only available path. Thus state S also represents 32 configurations. A configuration in state T derives from a configuration in state J by connecting either input on the opposite input switch as the established connection to either output on the opposite output switch as the established connection by the path using the unused middle switch. Thus state T represents $32 \times 2 \times 2 = 128$ configurations. A configuration in state U derives from a configuration in state J by connecting either input on the opposite input switch as the established connection to either output on the opposite output switch as the established connection by the path sharing the used middle switch. Thus state U represents $32 \times 2 \times 2 = 128$ configurations. A configuration in state V derives from a configuration in state J by connecting the only other input on the same input switch as the established connection to either output on the opposite output switch as the established connection, or either input on the opposite input switch as the established connection to the only other output on the same output switch as the established connection by the only available path. Thus state V represents $32 \times (2 + 2) = 128$ configurations.

States W and X represent all configurations with three established connections. A configuration in state W derives from a configuration in state S by connecting either input on the unused input switch to either output on the unused output switch by either available path, or

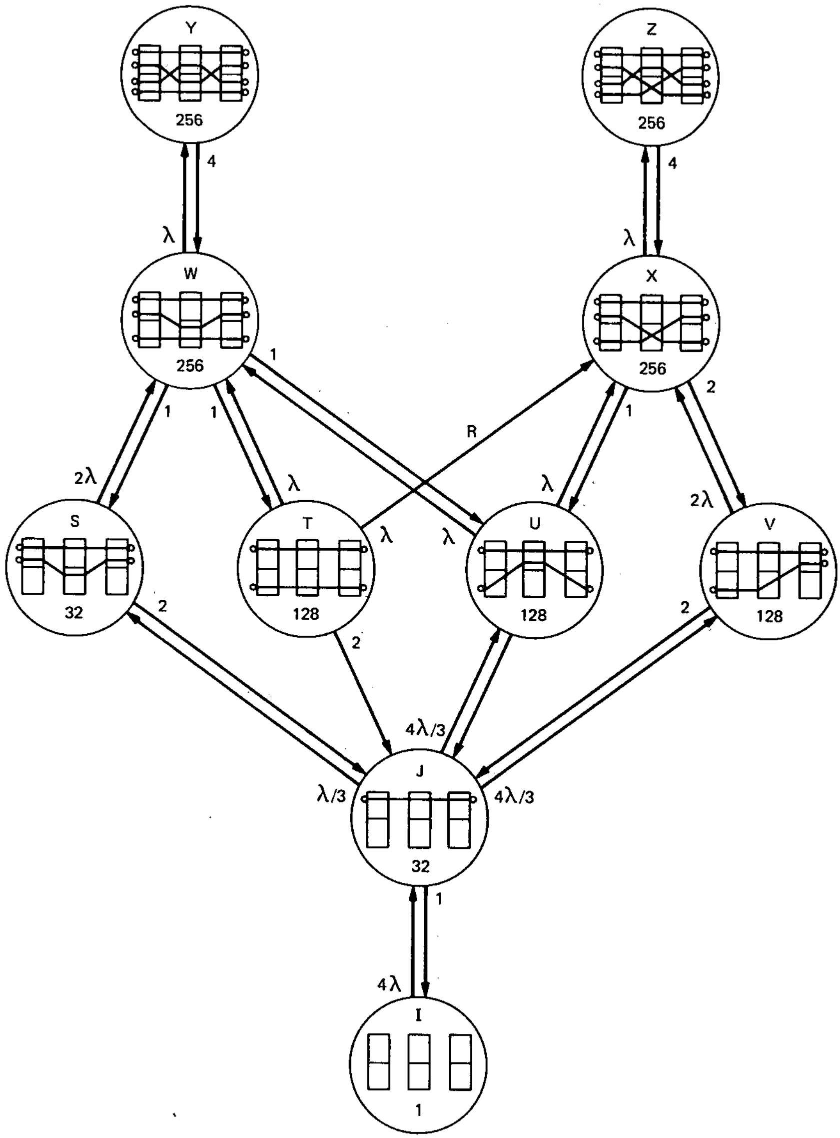

Fig. 15—Configurations per state in the 222 Module.

it derives from a configuration in state T or U by interconnecting either of two specific I/O pairs by the only available path. The specific pair share both an input switch and an output switch with either established connection. Thus state W represents $32 \times 2 \times 2 \times 2 = 128 \times 2 = 256$ configurations. A configuration in state X derives from a configuration in state U by connecting either of two specific I/O pairs, those that share an input switch with one established connection and an output switch with the other one, by the only available path. Each also derives from configurations in state V by connecting either idle input to either idle output by the only available path, but each derives from two different configurations in state V. Thus state X represents $128 \times 2 = 128 \times 2 \times 2/2 = 256$ configurations.

States Y and Z represent all configurations with four established connections. Each configuration in state Y derives from a configuration in state W by connecting the only idle I/O pair by the only available path. Each configuration in state Z derives from a configuration in state X by connecting the only idle I/O pair by the only available path. Thus state Y represents 256 configurations and state Z represents 256 configurations.

Summing, we count a total of

$$1 + 32 + 32 + 128 + 128 + 128 + 256 + 256 + 256 + 256 = 1473$$

distinct network configurations in the 222 Module.

9.2 Count of 2121 Module configurations

The Markov diagram of the 2121 Module, showing downward transitions, is repeated in Fig. 16, except that a number replaces the state name, the count of unique network configurations that are represented by the corresponding Markov state.

State FL represents the only idle network configuration. States GA through AZ represent all configurations with one established connection. In the four configurations in GA, any of the four inputs connects to the output on a middle switch diagonally across the module by the only path. In the configurations in AL, LA, and TX, any of the four inputs connects to either output on the output switch. Each state represents $4 \times 2 = 8$ configurations and the states are distinguished by which of three paths is used for the connection. In the configurations in NM and AZ, any of the four inputs connects to the output on a middle switch directly across the module. Each state represents four configurations and the states are distinguished by which of two paths is used for the connection. Summing it up, states GA through AZ represent $3 \times 4 + 3 \times 8 = 36$ configurations.

States NC through HI represent all configurations with two established connections. In one subset of these 21 states, the inputs, one from each input switch, connect to the two outputs on the middle switches. Each state in this subset—SC, MS, or HI—represents four configurations, and the states are distinguished by whether the connections are diagonally across the module or directly across by either of two similar paths. In OK, either output on a middle switch connects to either input directly across the module by the best path, and the

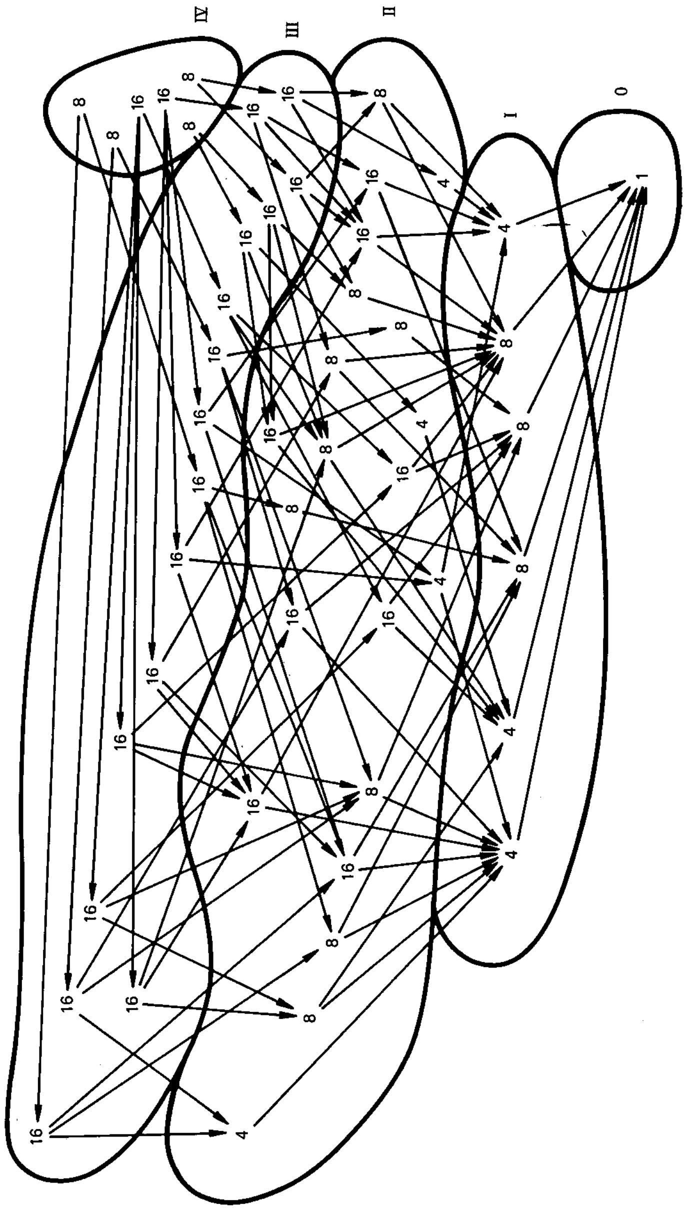

Fig. 16—Configurations per state for the 2121 Module.

other input on that same input switch connects diagonally across the module to the other output on a middle switch. In CA, either output on a middle switch connects to either input directly across the module by the best path and either input on the other input switch connects directly across the module to the other output on a middle switch by the worst path. OK represents $2 \times 2 = 4$ configurations and CA represents $2 \times 2 \times 2 = 8$ configurations.

In another subset of these 21 states, any of the four inputs connects to either output on the output switch and the other input on the same switch connects to any other output by the best remaining path. Each state in this subset—NC, IN, MO, CO, or NV—represents $4 \times 2 = 8$ configurations, and the states are distinguished by the path of the first connection and/or the output of the second connection. In another subset of these 21 states, either input on the upper input switch connects to either output on the output switch, and either input on the lower input switch connects to the other output on the output switch, and both connections use similar paths. Each state in this subset—KY, TN, or IA—represents $2 \times 2 \times 2 = 8$ configurations, and the states are distinguished by the three possible paths. In the final subset of these 21 states, any of the four inputs connects to either output on the output switch and either input on the other input switch connects to any other output. Excluding the cases, covered in the previous subset, where the paths are symmetric, each state in this subset—VA, WV, IL, AR, NE, KS, WY, and UT—represents $4 \times 2 \times 2 = 16$ configurations, and the states are distinguished by the path of the first connection and/or the output and/or the path of the second connection. Summing it up, states NC through HI represent $4 \times 4 + 9 \times 8 + 8 \times 16 = 216$ configurations.

States DE through AK represent all configurations with three established connections. In all 16 states, any of four inputs connects to either output on the output switch, and either input on the other input switch connects to some other output. Each state represents $4 \times 2 \times 2 = 16$ configurations, and the states are distinguished by the path of the first connection and/or the path and/or output of the second connection and/or the input and/or output and/or path of the third connection. Summing then, states DE through AK represent $16 \times 16 = 256$ configurations.

States RI through VT represent all configurations with four established connections. In the subset containing RI, CT, ME, and VT, either input on the upper input switch connects to either output on the output switch, either input on the lower input switch connects to the other output on the output switch, and both connections use similar paths. The remaining two inputs connect to the remaining two outputs, by the best remaining similar paths. Each state represents

$2 \times 2 \times 2 = 8$ configurations, and the states are distinguished by the path of the first two connections and/or the parties interconnected by the last two connections. In MA, both inputs on one input switch connect to the outputs on the middle switches by the best paths, and both inputs on the other input switch connect to the outputs on the output switch. In NH, both inputs on one input switch connect, one to the output straight across on a middle switch and the other to one of the outputs on the output switch, using paths like those in VT. Both inputs on the other input switch connect to similar outputs, but using paths like those in CT. Both MA and NH represent $2 \times 4 \times 2 = 16$ configurations. Summing them up, states RI through VT represent $4 \times 8 + 2 \times 16 = 64$ configurations.

Summing them up, we count a total of

$$1 + 36 + 216 + 256 + 64 = 573$$

distinct network configurations in the 2121 Module. The module supports more configurations, but no others are reached using the prudent rule. We see that the 222 Module has more than 2.5 times as many reachable configurations as the 2121 Module and, hence, a table-look-up implementation of a control algorithm would be more complex for the 222 Module than for the 2121 Module.

X. CONCLUSION

Two architectures for a 4×4 photonic switching network were compared by their traffic-handling capacity. Both networks are rearrangeably nonblocking. The percentage of sequences requiring rearrangement was found to be tolerable for both modules. Thus, both modules are judged acceptable, insofar as rearrangeably nonblocking modules are acceptable, and practically indistinguishable in their traffic performance. The selection of one module over the other may proceed based on criteria other than traffic capacity, like loss, crosstalk, or cost of manufacture.

XI. ACKNOWLEDGMENTS

Acknowledgments are extended to Vic Beneš, Nick Maxemchuk, and Krishnan Padmanabhan for their time, encouragement, and useful discussion and to Eric Grosse for his help with SMP.[17]

REFERENCES

1. G. Broomell and J. R. Heath, "Classification Categories and Historical Development of Circuit Switching Topologies," Comput. Surv., *15,* No. 2 (June 1983), pp. 95–133.
2. V. E. Beneš, *Mathematical Theory of Connecting Networks,* New York: Academic Press, 1965.

3. R. C. Alferness, R. V. Schmidt, and E. H. Turner, "Characteristics of Ti-Diffused Lithium Niobate Optical Directional Couplers," Appl. Opt., *18,* No. 23 (December 1979), pp. 4012–16.
4. A. E. Joel, "On Permutation Switching Networks, B.S.T.J., *47,* No. 5 (May 1968), pp. 813–22.
5. S. K. Korotky et al., "Fully Connectorized High-Speed Ti:LiNbO$_3$ Switch/Modulator for Time-Division Multiplexing and Data Encoding," J. Lightwave Technol., *LT-3,* No. 1 (February 1985).
6. L. McCaughan, "Design and Performance Limitation of Integrated Electro-Optic Cross Points," Proc. IEEE Globecom, Alanta, November 1984, pp. 878–9.
7. J. E. Watson, "Polarization-Independent 1 × 16 Optical Switch Using Ti:LiNbO$_3$ Waveguides," Conf. Optical Fiber Commun., San Diego, February 1985, p. 110.
8. H. S. Hinton, "A Nonblocking Optical Interconnection Network Using Directional Couplers," Proc. of IEEE Globecom, Atlanta, November 1984, pp. 885–9.
9. L. McCaughan and G. A. Bogert, "4 × 4 Strictly Nonblocking Integrated Ti:LiNbO$_3$ Switch Array," Conf. Optical Fiber Commun., San Diego, February 1985, p. 76.
10. E. E. Bergman, L. McCaughan, and J. E. Watson, "Coupling of Intersecting Ti:LiNbO$_3$ Diffused Waveguides," Appl. Opt., *23,* No. 17 (September 1984), pp. 3192–5.
11. S. Kobayashi and T. Kimura, "Semiconductor Optical Amplifiers," IEEE Spectrum, *21,* No. 5 (May 1984), pp. 26–33.
12. R. A. Spanke and V. E. Beneš, private communication.
13. R. A. Thompson, private communication.
14. M. Schwartz, Computer-Communication Network Design and Analysis, Englewood Cliffs, N.J.: Prentice-Hall, 1977.
15. L. Kleinrock, Queueing Systems, New York: Wiley, 1975.
16. V. E. Beneš, "Programming and Control Problems Arising from Optimal Routing in Telephone Networks," B.S.T. J., *45,* No. 9 (November 1966), pp. 1373–438.
17. SMP, A Symbolic Manipulation Program, Primer and Summary, Inference Corp., 1983.

APPENDIX A

Rate Equations for the 2121 Module

$$(4\lambda)\, q_{FL} = q_{AZ} + q_{NM} + q_{TX} + q_{LA} + q_{AL} + q_{GA}$$

$$(3\lambda + 1)\, q_{AZ} = \lambda q_{FL} + 2q_{HI} + q_{CA} + q_{NV} + q_{UT} + q_{WY} + q_{CO} + q_{NE} + q_{OK}$$
$$(3\lambda + 1)\, q_{NM} = q_{CA} + q_{KS} + q_{IL} + q_{MO} + 2q_{MS}$$
$$(3\lambda + 1)\, q_{TX} = q_{NV} + q_{UT} + 2q_{IA} + q_{IN} + q_{VA}$$
$$(3\lambda + 1)\, q_{LA} = q_{CO} + q_{KS} + q_{NE} + q_{AR} + 2q_{TN}$$
$$(3\lambda + 1)\, q_{AL} = 2\lambda q_{FL} + q_{WY} + q_{AR} + q_{MO} + q_{IL} + q_{IN} + 2q_{KY} + q_{WV} + q_{NC}$$
$$(3\lambda + 1)\, q_{GA} = \lambda q_{FL} + q_{OK} + q_{WV} + q_{VA} + 2q_{SC} + q_{NC}$$

$$(2\lambda + 2)\, q_{HI} = (2\lambda/3)\, q_{AZ} + q_{AK} + q_{WA}$$
$$(2\lambda + 2)\, q_{CA} = (2\lambda/3)\, q_{NM} + q_{OR} + q_{ID}$$
$$(2\lambda + 2)\, q_{NV} = (\lambda/3)[2q_{AZ} + q_{TX}] + q_{AK} + q_{MN}$$
$$(2\lambda + 2)\, q_{UT} = (2\lambda/3)\, q_{TX} + q_{AK} + q_{ND} + q_{MN} + q_{WI}$$
$$(2\lambda + 2)\, q_{WY} = (\lambda/3)[4q_{AZ} + 2q_{AL}] + q_{OR} + q_{MT} + q_{ND} + q_{SD}$$
$$(2\lambda + 2)\, q_{CO} = (\lambda/3)\, q_{LA} + q_{WA} + q_{ID} + q_{MT} + q_{MI}$$
$$(2\lambda + 2)\, q_{NE} = (2\lambda/3)\, q_{LA} + q_{WA} + q_{MI}$$
$$(2\lambda + 2)\, q_{KS} = q_{ID} + q_{OH}$$
$$(2\lambda + 2)\, q_{OK} = (\lambda/3)[q_{AZ} + q_{NM} + q_{GA}] + q_{SD} + q_{WI}$$

$$(2\lambda + 2)\, q_{IA} = (2\lambda/3)\, q_{TX} + \quad q_{MN}$$
$$(2\lambda + 2)\, q_{AR} = (2\lambda/3)\, q_{LA} + \quad q_{MT} + q_{OH}$$
$$(2\lambda + 2)\, q_{MO} = (\lambda/3)[2q_{NM} + q_{AL}] + \quad q_{OR} + q_{OH} + q_{NY} + q_{PA}$$
$$(2\lambda + 2)\, q_{IL} = (4\lambda/3)\, q_{NM} + \quad q_{NY} + q_{PA}$$
$$(2\lambda + 2)\, q_{MS} = \quad q_{NY}$$
$$(2\lambda + 2)\, q_{IN} = (\lambda/3)[q_{TX} + q_{LA} + q_{AL}] + \quad q_{ND} + q_{NJ}$$
$$(2\lambda + 2)\, q_{TN} = \quad q_{MI}$$
$$(2\lambda + 2)\, q_{KY} = (2\lambda/3)\, q_{AL} + \quad q_{PA} + q_{MD}$$
$$(2\lambda + 2)\, q_{WV} = (\lambda/3)[2q_{LA} + 2q_{AL} + 4q_{GA}] + \quad q_{SD} + q_{MD} + q_{NJ} + q_{DE}$$
$$(2\lambda + 2)\, q_{VA} = (2\lambda/3)\, q_{TX} + \quad q_{WI} + q_{NJ}$$
$$(2\lambda + 2)\, q_{SC} = (2\lambda/3)\, q_{GA} + \quad q_{DE}$$
$$(2\lambda + 2)\, q_{NC} = (\lambda/3)[q_{AL} + q_{TX} + q_{LA} + 2q_{GA}] + \quad q_{MD} + q_{DE}$$

$$(\lambda + 3)\, q_{AK} = (\lambda/2)[4q_{HI} + 2q_{NV} + q_{UT}] + \quad 2q_{ME}$$
$$(\lambda + 3)\, q_{OR} = (\lambda/2)[2q_{CA} + q_{WY} + 2q_{MO}] + \quad q_{NH}$$
$$(\lambda + 3)\, q_{WA} = (\lambda/2)[2q_{CO} + q_{NE}] + \quad 2q_{VT}$$
$$(\lambda + 3)\, q_{ID} = (\lambda/2)[2q_{CA} + q_{KS}] + \quad q_{NH}$$
$$(\lambda + 3)\, q_{MT} = (\lambda/2)[q_{WY} + q_{AR}] + \quad q_{NH}$$
$$(\lambda + 3)\, q_{ND} = (\lambda/2)[q_{WY} + q_{UT} + q_{KS} + q_{NE} + 2q_{IN} + q_{IL}] + \quad q_{MA}$$
$$(\lambda + 3)\, q_{SD} = (\lambda/2)[q_{WY} + q_{NE} + q_{KS} + 4q_{OK} + q_{IL} + q_{WV}] + \quad q_{MA}$$
$$(\lambda + 3)\, q_{MN} = (\lambda/2)[2q_{NV} + q_{UT} + 2q_{IA}] + \quad 2q_{ME}$$
$$(\lambda + 3)\, q_{WI} = (\lambda/2)[q_{UT} + q_{VA}] + \quad q_{MA}$$
$$(\lambda + 3)\, q_{MI} = (\lambda/2)[q_{NE} + 2q_{CO} + 2q_{TN}] + \quad 2q_{VT}$$
$$(\lambda + 3)\, q_{OH} = (\lambda/2)[q_{KS} + q_{AR}] + \quad q_{NH}$$
$$(\lambda + 3)\, q_{NY} = (\lambda/2)[q_{IL} + 4q_{MS}] + \quad 2q_{CT}$$
$$(\lambda + 3)\, q_{PA} = (\lambda/2)[q_{IL} + 2q_{MO} + 2q_{KY}] + \quad 2q_{CT}$$
$$(\lambda + 3)\, q_{MD} = (\lambda/2)[2q_{IA} + 2q_{AR} + 2q_{KY} + 2q_{TN} + q_{WV} + q_{VA} + 2q_{NC}] + \quad 2q_{RI}$$
$$(\lambda + 3)\, q_{NJ} = (\lambda/2)[2q_{IN} + q_{WV} + q_{VA}] + \quad q_{MA}$$
$$(\lambda + 3)\, q_{DE} = (\lambda/2)[q_{WV} + q_{VA} + 4q_{SC} + 2q_{NC}] + \quad 2q_{RI}$$

$$4q_{ME} = \lambda[q_{AK} + q_{MN}]$$
$$4q_{VT} = \lambda[q_{WA} + q_{MI}]$$
$$4q_{NH} = \lambda[q_{ID} + q_{MT} + q_{OR} + q_{OH}]$$
$$4q_{MA} = \lambda[q_{ND} + q_{SD} + q_{WI} + q_{NJ}]$$
$$4q_{CT} = \lambda[q_{NY} + q_{PA}]$$
$$4q_{RI} = \lambda[q_{MD} + q_{DE}]$$

APPENDIX B

Approximate Probabilities for the 2121 Module

$$q_{FL} = 1/(1 + \lambda)^4$$

$$q_{AZ} = (2.8\lambda^2 + 1.1\lambda)/(3\lambda + 1)(1 + \lambda)^4$$
$$q_{NM} = (1.2\lambda^2 - 0.3\lambda)/(3\lambda + 1)(1 + \lambda)^4$$

$$q_{TX} = (1.4\lambda^2) \quad /(3\lambda + 1)(1 + \lambda)^4$$
$$q_{LA} = (0.9\lambda^2 - 0.5\lambda) /(3\lambda + 1)(1 + \lambda)^4$$
$$q_{AL} = (3.6\lambda^2 + 2.4\lambda) /(3\lambda + 1)(1 + \lambda)^4$$
$$q_{GA} = (2.1\lambda^2 + 1.3\lambda) /(3\lambda + 1)(1 + \lambda)^4$$

$$q_{HI} = (0.5\lambda^3 + 0.6\lambda^2) \quad /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{CA} = (0.4\lambda^3 + 0.3\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{NV} = (0.6\lambda^3 + 0.8\lambda^2) \quad /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{UT} = (1.0\lambda^3 + 0.4\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{WY} = (1.4\lambda^3 + 2.0\lambda^2 + 0.3\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{CO} = (0.6\lambda^3 + 0.1\lambda^2) \quad /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{NE} = (0.3\lambda^3 + 0.2\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{KS} = (0.1\lambda^3) \quad /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{OK} = (0.6\lambda^3 + 0.7\lambda^2) \quad /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{IA} = (0.2\lambda^3 + 0.3\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{AR} = (0.2\lambda^3 + 0.2\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{MO} = (0.8\lambda^3 + 0.7\lambda^2) \quad /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{IL} = (0.3\lambda^3 + 0.5\lambda^2 - 0.2\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{MS} = (0.1\lambda^3) \quad /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{IN} = (0.6\lambda^3 + 0.7\lambda^2) \quad /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{TN} = (0.1\lambda^3) \quad /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{KY} = (0.8\lambda^3 + 0.8\lambda^2 + 0.2\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{WV} = (1.7\lambda^3 + 1.9\lambda^2 + 0.3\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{VA} = (0.3\lambda^3 + 0.3\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{SC} = (0.4\lambda^3 + 0.5\lambda^2 + 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{NC} = (0.9\lambda^3 + 1.1\lambda^2 + 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$

$$q_{AK} = (0.3\lambda^4 + 1.0\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{OR} = (0.1\lambda^4 + 0.9\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{WA} = (0.2\lambda^4 + 0.4\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{ID} = (0.1\lambda^4 + 0.2\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{MT} = (0.1\lambda^4 + 0.3\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{ND} = (0.3\lambda^4 + 1.1\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{SD} = (0.3\lambda^4 + 1.5\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{MN} = (0.3\lambda^4 + 0.6\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{WI} = (0.3\lambda^4 + 0.3\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{MI} = (0.2\lambda^4 + 0.5\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{OH} = (0.1\lambda^4 + 0.1\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{NY} = (0.2\lambda^4 + 0.4\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{PA} = (0.2\lambda^4 + 0.9\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{ND} = (0.5\lambda^4 + 1.7\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{NJ} = (0.3\lambda^4 + 0.7\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{DE} = (0.5\lambda^4 + 1.3\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$

$$q_{ME} = (0.1\lambda^4)/(1+\lambda)^4$$
$$q_{VT} = (0.1\lambda^4)/(1+\lambda)^4$$
$$q_{NH} = (0.2\lambda^4)/(1+\lambda)^4$$
$$q_{MA} = (0.3\lambda^4)/(1+\lambda)^4$$
$$q_{CT} = (0.1\lambda^4)/(1+\lambda)^4$$
$$q_{RI} = (0.2\lambda^4)/(1+\lambda)^4$$

AUTHOR

Richard A. Thompson, B. S. (Electrical Engineering), 1964, Lafayette College; M. S. (Electrical Engineering), 1966, Columbia University; Ph.D. (Computer Science), 1971, University of Connecticut; AT&T Bell Laboratories, 1963–1968, 1977—. From 1963 to 1968 Mr. Thompson was involved in switching systems development at AT&T Bell Laboratories. He was a member of the Electrical Engineering Department at Virginia Polytechnic Institute from 1971 to 1977, achieving the rank of Associate Professor. Since 1977 Mr. Thompson has been a member of the Digital Systems Research Department at AT&T Bell Laboratories, with a brief stint in ABI/ATTCP in 1983. His research interests are probabilistic formal languages, fault tolerance and cellular automata, terminals and the human-machine interface, communications switching systems, and, most recently, photonic switching. Mr. Thompson is an active participant in IEEE Computer and Communications Societies. Senior Member, IEEE.