MARYLAND

TECHNICAL
RESEARCH
REPORT

SRC TR 88-96

Monotonicity of Throughput In Non-Markovian Networks

by

P. Tsoucas and J. Walrand

SYSTEMS RESEARCH CENTER
UNIVERSITY OF MARYLAND
COLLEGE PARK, MARYLAND 20742

Monotonicity of Throughput in non-Markovian Networks

by

Pantelis Tsoucas

Systems Research Center
University of Maryland, College Park MD 20742

and

Jean Walrand

Department of Electrical Engineering and Computer Sciences and
Electronics Research Laboratory
University of California, Berkeley CA 94720

ABSTRACT

Monotonicity of throughput is established in some non-Markovian queueing networks by means of path-wise comparisons. In a series of $\cdot/GI/s/N$ queues with loss at the first node it is proved that increasing the waiting room and/or the number of servers increases the throughput. For a closed network of $\cdot/GI/s$ queues it is shown that the throughput increases as the total number of jobs increases. The technique used for these results does not apply to blocking systems with finite buffers and feedback. Using a stronger coupling argument we prove throughput monotonicity as a function of buffer size for a series of two $\cdot/M/1/N$ queues with loss and feedback from the second to the first node.

Keywords: Blocking; closed queueing networks; monotonicity; throughput.

1. Introduction

This paper is concerned with some classes of non-Markovian networks. The aim is to establish the monotonicity of throughput in these networks as a function of some of their parameters. The proofs involve pathwise arguments.

In Section 2 we consider a loss system consisting of M $\cdot/GI/s/N$ queues in series. The queues are initially empty. The blocking between nodes is of the manufacturing type and the arrival process is arbitrary. It is shown that the number of customers accepted in the system in any time interval $[0,t]$, $t \geq 0$, is a stochasticaly increasing function of the buffer sizes and the number of servers, and that manufacturing-type blocking performs stochastically better than communication-type blocking.

In Section 3 we consider a closed network of $\cdot/GI/s$ queues. It is shown that the number of jobs processed by any node in the network in any time interval $[0,t]$ is a stochastically increasing function of the total number of customers in the network. The idea of the proof is in van Dijk et al [3]. Monotonicity with respect to the number of servers can be argued similarly. This extends results of Shantikumar and Yao [4] and van der Wal [6] who consider closed networks with exponential servers.

The technique of the previous sections does not apply to finite-buffer networks with feedback. In Section 4 we consider a simple system consisting of two initially empty $\cdot/M/1/N$ queues in tandem with communication blocking between the nodes and feedback from the second to the first node. Again, the arrival process is arbitrary. It is shown that the number of accepted customers in the system in any time interval $[0,t]$ is a stochastically monotone function of the buffer sizes of the nodes. The coupling construction employed here depends crucially on the fact that the service times are exponentially distributed. Monotonicity of throughput as discussed in Sections 2 and 4 is important for establishing properties of the Erlang fixed point approximation of these systems.

In this paper the throughput of a queueing node at time $t \geq 0$ is defined to be the number of jobs processed by the node by time t. In standard usage it is defined to be the time average of this quantity.

2. A series of $\cdot/GI/s/N$ queues

2.1 Throughput as a function of buffer size and the number of servers

Consider M queues $\{\cdot/(GI)_i/s_i/N_i\}_{i=1}^{M}$ in series. That is, node i has s_i servers with i.i.d. service times distributed according to $(GI)_i$ and waiting room of size N_i. An arriving job that finds the first node full is lost. A job that completes service in node i proceeds to node $i+1$ unless the latter is full. In this case it has to wait until there is an empty space in node $i+1$ while node i is blocked, i.e., while its server is idle. This discipline is called 'manufacturing blocking.' Let the arrival times in the first node be an arbitrary deterministic sequence $(a_k)_{k=1}^{\infty}$ and assume that the system starts empty.

Notation

For $i = 1, ..., M$, $j \geq 1$, we define the following quantities.

S_j^i: service requirement of the jth job through the ith node,

C_j^i: time when the jth job completes service in node i but does not necessarily join node $i+1$,

T_j^i: time of arrival of the jth job to be served in node i.

The *maximum* of x and y is denoted by $x \vee y$, $(x)^+ = x \vee 0$ and $x \wedge y$ is the *minimum* of x and y. By convention, $x \wedge y \vee z = (x \wedge y) \vee z$ and $x \vee y + z = (x \vee y) + z$. By ${}^{(s)}\wedge_k \, x_k$ we denote the sth smallest element in a set $\{x_k\}_k$. Lastly, $1\{\cdot\}$ is the indicator of an event.

Recursive relationships

We now establish recursive relationships for the times defined above. To this end note that in the ith node a server becomes available for the nth job to arrive in that node only after time $T^{i+1}_{n-s_i+1}$, i.e., only after $n - s_i + 1$ jobs have left node i. Similarly, note that the same job is blocked by node $i+1$ at most until time $T^{i+2}_{n-N_{i+1}+1}$.

For $j \geq 1$ and $i = 1, ..., M$, one verifies the following recursions. They are generalizations of the ones given in Wolff [7] for a single $\cdot/\cdot/s$ node.

$$C^i_j = T^i_j \vee T^{i+1}_{j-s_i} + S^i_j \tag{2.1}$$

$$T^{i+1}_j = {}^{(j)}\wedge^{j+s_i-1}_{k=1} \; C^i_k \vee T^{i+2}_{j-N_{i+1}} \qquad (2.2).$$

By convention, $T^{M+2}_j = T^i_0 = T^i_{-j} \equiv 0$ and note that $T^{M+1}_j = C^M_j$. Also, the minimum and the minimum over an empty set are taken to be zero.

We will make use of these relationships in establishing the main result in this section. By A_t denote the number of customers accepted in the system by time $t \geq 0$. Let $\overline{A}_t$ denote the same quantity for the same system but with buffer sizes increased to $\overline{N}_1 \geq N_1, ..., \overline{N}_M \geq N_M$ and with the number of servers at each node increased to $\overline{s}_1 \geq s_1, ..., \overline{s}_M \geq s_M$.

Theorem 2.1 : The following relationship holds. (Recall that $X \geq_{st} Y$ if $P(X \geq x) \geq P(Y \geq x)$ for $x \in \mathbf{R}$.)

$$A_t \leq_{st} \overline{A}_t, t \geq 0. \tag{2.3}$$

Proof : The proof consists in constructing a queueing process in a system with buffer sizes $\{\overline{N}_i\}$ and the same arrival sequence $(a_k)^\infty_{k=1}$, where (2.3) is satisfied almost surely. Variables $\overline{S}^i_j$, $\overline{C}^i_j$ and $\overline{T}^i_j$ denote the corresponding quantities for the larger system. Since service times are independent of the past of the arrival and the service processes we can require that $S^i_j = \overline{S}^i_j$, for $i = 1, ..., M$ and $j \geq 1$, almost surely, without altering the distributions of $\{(S^i_l)^\infty_{l=1}\}^M_{i=1}$, and $\{(\overline{S}^i_l)^\infty_{l=1}\}^M_{i=1}$.

We now show by induction on j that, for $i = 1, ..., M$ and $j = 1, 2, ...$,

$$C^i_j \geq \overline{C}^i_j, \text{ and} \tag{2.4}$$

$$T^i_j \geq \overline{T}^i_j, \text{ a.s.} \tag{2.5}$$

This is clear for $j = 1$ and assume it is true for $j = 2, ..., n$. We will prove that it remains true for $j = n + 1$.

Relationships (2.1) and (2.2) are non-increasing for increasing $\{N_i\}^M_{i=2}$ and $\{s_i\}^M_{i=1}$. Also, again by (2.1) and (2.2), note that in order to establish (2.4) and (2.5) for $j = n+1$ it suffices to show that $T^1_{n+1} \geq \overline{T}^1_{n+1}$. Suppose on the contrary that $T^1_{n+1} < \overline{T}^1_{n+1}$ and let $T^1_{n+1} = a_k$

for some k. This implies that in node 1, at time a_k, there are $\overline{N}_1$ jobs in the larger system while there are strictly less than N_1 jobs in the smaller one. Therefore,

$$T^2_{n-\overline{N}_1+1} \le T^2_{n-N_1+1} \le a_k \; < \; \overline{T}^2_{n-\overline{N}_1+1},$$

which contradicts the induction hypothesis. This proves inequalities (2.4) and (2.5) and inequality (2.3) follows from that $T^1_j \ge \overline{T}^1_j$, $j \ge 1$. Also, from the fact that $T^M_j \ge \overline{T}^M_j$, $j \ge 1$, it follows that the departure process stochastically increases with increasing $\{N_i\}^M_{i=1}$ and/or $\{s_i\}^M_{i=1}$. □

Remark 2.1 : In the theorem above and in Theorem 4.1 the arrival times are arbitrary and deterministic. This implies that the results remain true for an arbitrary random sequence of arrival times as long as the future service requirements are independent of the past of the arrival process.

2.2 Manufacturing vs. communication blocking

In this subsection we consider the same series of $\{\cdot/(GI)_i/1/N_i\}^M_{i=1}$ queues but with communication-type blocking. This means that if a job completes service at node i and finds node $i+1$ full, then it has to repeat service in the former node. Starting the system empty, we prove that the throughput of this system is a lower bound on the throughput of the system with manufacturing-type blocking. This fact is useful because communication blocking is usually easier to analyze (see, e.g., Section 4 and Mitra and Tsoucas [3]). The issue was raised in Altiok and Stidham [1].

Variables $\tilde{A}_t$, $\tilde{S}^i_j$, $\tilde{C}^i_j$ and $\tilde{T}^i_j$ are defined in the same way as variables A_t, S^i_j, C^i_j, and T^i_j above. In addition, let p^i_j be the smallest integer such that

$$\sum_{l=1}^{p^i_j} S^i_{j,l} \ge (\tilde{T}^{i+2}_{j-N_{i+1}} - \tilde{C}^i_j)^+, \quad n \ge 1, \; i = 1, ..., M-1, \tag{2.6}$$

i.e., the additional number of times job j will have to be served at node i before it can join node $i+1$ ($p^i_j = 0$ if $\tilde{T}^{i+2}_{j-N_{i+1}} \le \tilde{C}^i_j$). The variables $(S^i_{j,l})_l$ are picked independently from the distribution $(GI)_i$. The recursions satisfied are

$$\tilde{C}^i_{j+1} = \tilde{C}^i_j \vee T^i_{j+1} + \tilde{S}^i_{j+1}, \tag{2.7}$$

$$\tilde{T}^{i+1}_{j+1} = \tilde{C}^i_{j+1} + \sum_{l=1}^{p^i_{j+1}} S^i_{j+1,l}, \; i = 1, ..., M, \; j \ge 1. \tag{2.8}$$

Theorem 2.2 : One has

$$A_t \ge_{st} \tilde{A}_t, \; t \ge 0.$$

Proof : The argument is essentially the same as the one in the proof of Theorem 2.1. By requiring that $S_j^i = \tilde{S}_j^i$ for all i and j, one proves inductively that

$$C_j^i \le \tilde{C}_j^i \text{ and } T_j^i \le \tilde{T}_j^i,\ j \ge 1,\ i = 1, ..., M.$$

The induction hypothesis propagates because, from (2.8), and (2.6),

$$\tilde{T}_{j+1}^{i+1} \ge \tilde{C}_{j+1}^i + (\tilde{T}_{j-N_{i+1}+1}^{i+2} - \tilde{C}_{j+1}^i)^+ = \tilde{C}_j^i \vee \tilde{T}_{j-N_{i+1}+1}^{i+2},$$

while

$$T_{j+1}^{i+1} = C_{j+1}^i \vee T_{j-N_{i+1}+1}^{i+2}.$$

□

Remark 2.2 : One can show the same result for a series of $\cdot/GI/s/N$ queues. We have restricted attention to single-server nodes in order to keep the notation simple.

3. A closed network of $\cdot/GI/s$ queues

Consider a closed network $\mathcal{N}$ consisting of a collection of M nodes $\{\cdot/(GI)_i/s_i\}_{i=1}^M$. Routing between nodes is Bernoulli. Let K be the total number of jobs in the network. Service commences at $t = 0$ and the initial queue lengths are $(n_0^i)_{i=1}^M$. We will compare this network with one which consists of the same nodes and routing but with more jobs. The initial queue lengths in this network are assumed to be $\overline{n}_0^1 \ge n_0^1, ..., \overline{n}_0^M \ge n_0^M$. We denote this network by $\overline{\mathcal{N}}$. Finally, let P_t^i (respectively $\overline{P}_t^i$) be the number of jobs completed in node i by time $t \ge 0$ in network $\mathcal{N}$ (respectively $\overline{\mathcal{N}}$).

Theorem 3.1 : The following relationship holds.

$$P_t^i \le_{st} \overline{P}_t^i,\ t \ge 0,\ i = 1, ..., M. \tag{3.1}$$

Proof : Quantities S_j^i and T_j^i, $i = 1, ..., M$, $j \ge 1$ are defined as in the proof of Theorem 2.1. In addition we introduce

$R_{n,t}^i$: time remaining at time t until the jth accepted customer leaves node i. (by convention, $R_{n,t}^i = \infty$ if fewer than n customers have been processed by time t, $R_{n,t}^i = 0$ if the nth customer has departed by time t, and the paths of $R_{n,t}^i$ are taken to be right continuous),

B_n^i: time when a server in node i commences service on the jth job, i.e.,

$$B_n^i = \inf\{t \ge 0 \mid R_{n,t}^i \le S_n^i\}, \tag{3.2}$$

r_n^i: the nth routing decision of node i, i.e., $r_n^i = j$ means that the nth job through node i will be routed to node j upon completion.

Variables $\overline{S}_n^i$, $\overline{T}_n^i$, $\overline{R}_{n,t}^i$, $\overline{B}_n^i$, $\overline{r}_n^i$ denote the corresponding quantities for network $\overline{\mathcal{N}}$.

As in the proof of Theorem 2.1, we will construct processes in networks $\mathcal{N}$ and $\overline{\mathcal{N}}$ such that (3.1) holds almost surely. Since $(S_n^i)_n$ and $(r_n^i)_n$ are i.i.d. sequences we can require that

$S_n^i = \overline{S}_n^i$ and $r_n^i = \overline{r}_n^i$ a.s. for all i and n. Arguing by induction on t we show that, for $i = 1, ..., M$, $t \geq 0$ and $n \geq 1$,

$$R_{n,t}^i \geq \overline{R}_{n,t}^i. \tag{3.3}$$

Note that $T_n^i = \inf\{t \geq 0 \mid R_{n,t}^i < \infty\}$ and that for $t \geq T_n^i$, $R_{n,t}^i$ decreases at unit rate until it reaches 0 for all i and n (see Figure 1). Thus inequalities (3.3) need only be established for $t \in \{T_n^i\}$. We rewrite the set $\{T_n^i\}_{i,n}$ as $\{t_l\}_{l=1}^{\infty}$ with $t_l \leq t_{l+1}$. From the choice of initial conditions it is clear that in this construction one has

$$R_{n,t_1}^i \geq \overline{R}_{n,t_1}^i, \ i = 1, ..., M, \ n \geq 1.$$

Suppose (3.3) is true up to time t_l for all n and i. We show that it remains true up to time $t_l + 1$. Let $t_{l+1} = T_n^i$ for some i and n. It suffices to check that

$$T_n^i \geq \overline{T}_n^i \text{ and} \tag{3.4}$$

$$R_{n,T_n^i}^i \geq \overline{R}_{n,T_n^i}^i \ i = 1, ...M, \ n \geq 1. \tag{3.5}$$

To establish (3.4) suppose the contrary, i.e., that $T_n^i < \overline{T}_n^i$. By the induction hypothesis we only need to consider the kth job in node j where

$$T_n^i = B_k^i + S_k^j \text{ and for } t \geq T_n^i, \ r_k^j = i.$$

This implies that $B_k^j < \overline{B}_k^j$ and hence $R_{p,t}^j < \overline{R}_{p,t}^j$ for some $p \leq l$. This contradicts the induction hypothesis and shows (3.4).

For (3.5), note that because of the shape of $R_{n,t}^i$ the induction hypothesis implies that

$$\overline{R}_{l,t}^i \leq R_{l,t}^i, \text{ for } l < n. \tag{3.6}$$

Furthermore, note that we can write

$$B_n^i = T_n^i \vee \inf \{t \mid \sum_{l < n} 1\{R_{l,t}^i > 0\} < s_i\} \tag{3.7}$$

and

$$R_{n,t}^i = (B_n^i + S_n^i - t)^+. \tag{3.8}$$

Inequalities (3.6) and relation (3.7) imply that $\overline{B}_n^i \leq B_n^i$ and from (3.8) for $t = T_n^i$ we conclude (3.5). □

Remark 3.1 : Monotonicity of throughput with respect to the number of servers in a node can be established similarly.

4. Networks with finite buffers and feedback

In this section we consider the limitations of the coupling approach used so far. In the network of Section 2.1 take $M = 2$ and add Bernoulli feedback from the second to the first node with parameter p. The blocking of the fed-back jobs is also of the manufacturing type. We add an extra buffer space in the first node and consider the effect on the throughput. The construction of the previous sections can be shown to fail here, i.e., for some time t there are sample paths of positive probability for which the smaller system accepts strictly more customers (see Tsoucas [6]).

In what follows we establish the monotonicity property after simplifying the above system in two ways. The service time distributions are restricted to be exponential and the blocking is of the communication type. As before, let $(a_k)_{k=1}^{\infty}$ be an arbitrary deterministic sequence of arrivals, let μ_1, μ_2 be the service rates and let p be the feedback probability from node 2 to node 1. Denote by (X_t^1, X_t^2) the number of jobs in nodes 1 and 2 at time t, and by A_t the number of jobs accepted in the system by time t.

Next, consider the same system but with the buffer size in the first node increased to $N_1 + 1$. We can couple the two systems so as to have the same virtual service processes. The virtual service process in node 1 is a Poisson process with rate μ_1. A point in this process is a service completion time if node 1 is non-empty. The virtual service process in node 2 is a Poisson process with rate μ_2. A service completion occurs at each point of the process if node 2 is non-empty. The job completed gets fed back to node 1 with probability p and leaves the system otherwise. Its points carry additional routing information as to whether a job is to be fed back upon completion or not. Let $(\overline{X}_t^1,\ \overline{X}_t^2)$, $\overline{A}_t$ denote the corresponding quantities for this system. Finally, assume that $(X_0^1,\ X_0^2) = (\overline{X}_0^1,\ \overline{X}_0^2) = (0,0)$ and note that, in this construction, subsequent transition times are the same for both systems. Denote these times by $\{T_n\}_n \geq 1$.

Define $Y_t = \overline{X}_t - X_t$ and set $\Delta Y_t = Y_t - Y_{t-}$. The next two lemmas concern the process $\{Y_{T_n}\}$. Their proofs are straightforward (see Figure 2).

Lemma 4.1 : The transitions satisfy properties
(a) $Y_{T_n} \in \{(1,-1)\} \cup \{(n_1, n_2) \mid n_1 \geq 0,\ n_2 \geq 0\}$
(b) $\Delta Y_{T_n} = (-1,0)$ only if $Y_{T_n-}^1 \ > \ 1$, and$\Delta Y_{T_n} = (1,0)$ only if $Y_{T_n-}^1 = 0$.

In what follows we restrict attention to times T_n such that $Y_{T_n-} \not\in \{(0,0),\ (1,-1)\}$. We denote this set again by $\{T_n\}_n$.

Lemma 4.2 : Given that $Y_t \not\in \{(0,0),\ (1,-1)\}$,

$$Y_t^1 + Y_t^2 = \sum_{T_n \leq t} (1\{\Delta Y_{T_n} = (1,0)\} - 1\{\Delta Y_{T_n} = (0,-1)\} - 1\{\Delta Y_{T_n} = (-1,0)\}) + 1 \geq 1.$$

Finally, we can compare the throughput in the two systems.

Theorem 4.1 : One has $\overline{A}_t - A_t \geq 0,\ t \geq 0$.

Proof : Note that

$$\overline{A}_t - A_t \geq \sum_{T_n \leq t} (1\{\Delta Y_{T_n} = (1,0)\} \ - \ 1\{\Delta Y_{T_n} = (-1,0)\}).$$

Then the result follows from Lemmas 4.1 (b) and 4.2. □

5. References

[1] T. M. Altiok and S. Stidham. 'A note on transfer lines with unreliable machines, random processing times, and finite buffers,' *IIE Trans.* 14, 125–127 (1982).

[2] N. M. van Dijk, P. Tsoucas and J. Walrand. 'Simple bounds and monotonicity of the call congestion of finite multi-server delay systems,' to appear in *Probability in the Engineering and Informational Sciences* (1987).

[3] D. Mitra and P. Tsoucas. 'Relaxations for the numerical solutions of some stochastic problems,' AT&T Bell Labs Technical Memo. (1987).

[4] J. G. Shantikumar and D. D. Yao. 'The effect of increasing service rates in a closed queueing network,' *Jour. Appl. Prob.* 23, 474–483 (1986).

[5] P. Tsoucas. Ph.D. Dissertation, Dept. of Electrical Engineering and Computer Sciences, University of California, Berkeley (1987).

[6] J. van der Wal. 'Monotonicity of the throughput of a closed exponential queueing network in the number of jobs,' Eindhoven University of Technology Memo. COSOR 85-21 (1985).

[7] R. W. Wolff. 'An upper bound for multi-channel queues, ' *Jour. Appl. Prob.* 14, 884–888 (1977).

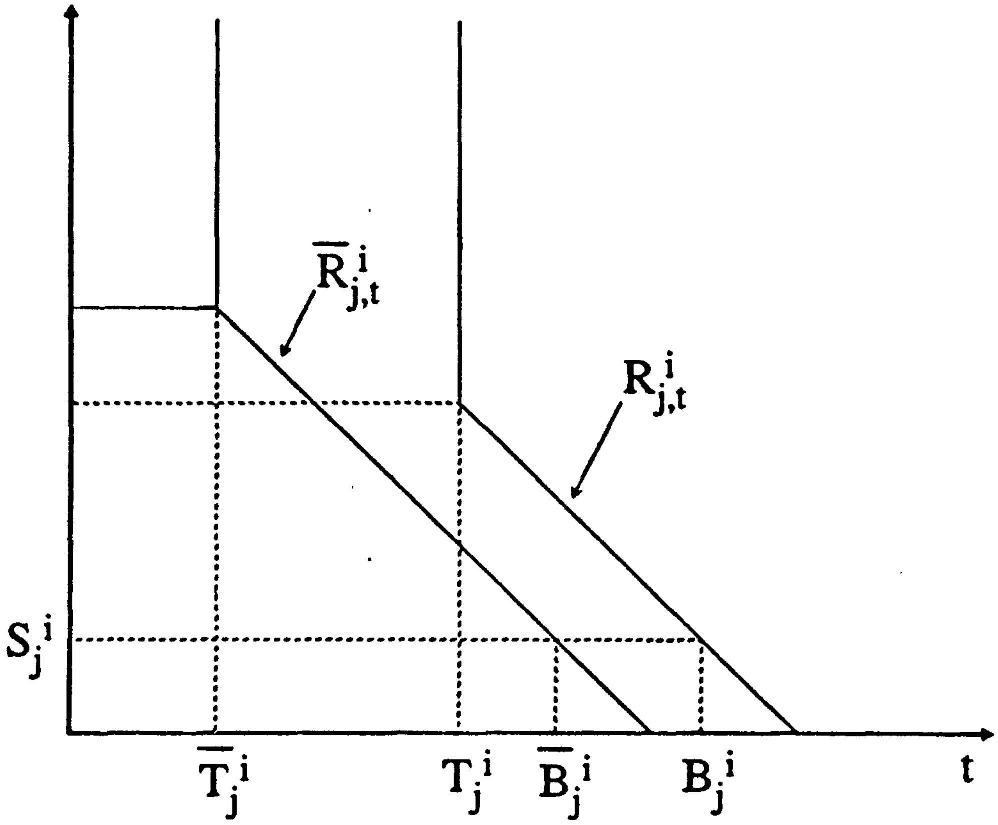

Figure 1.

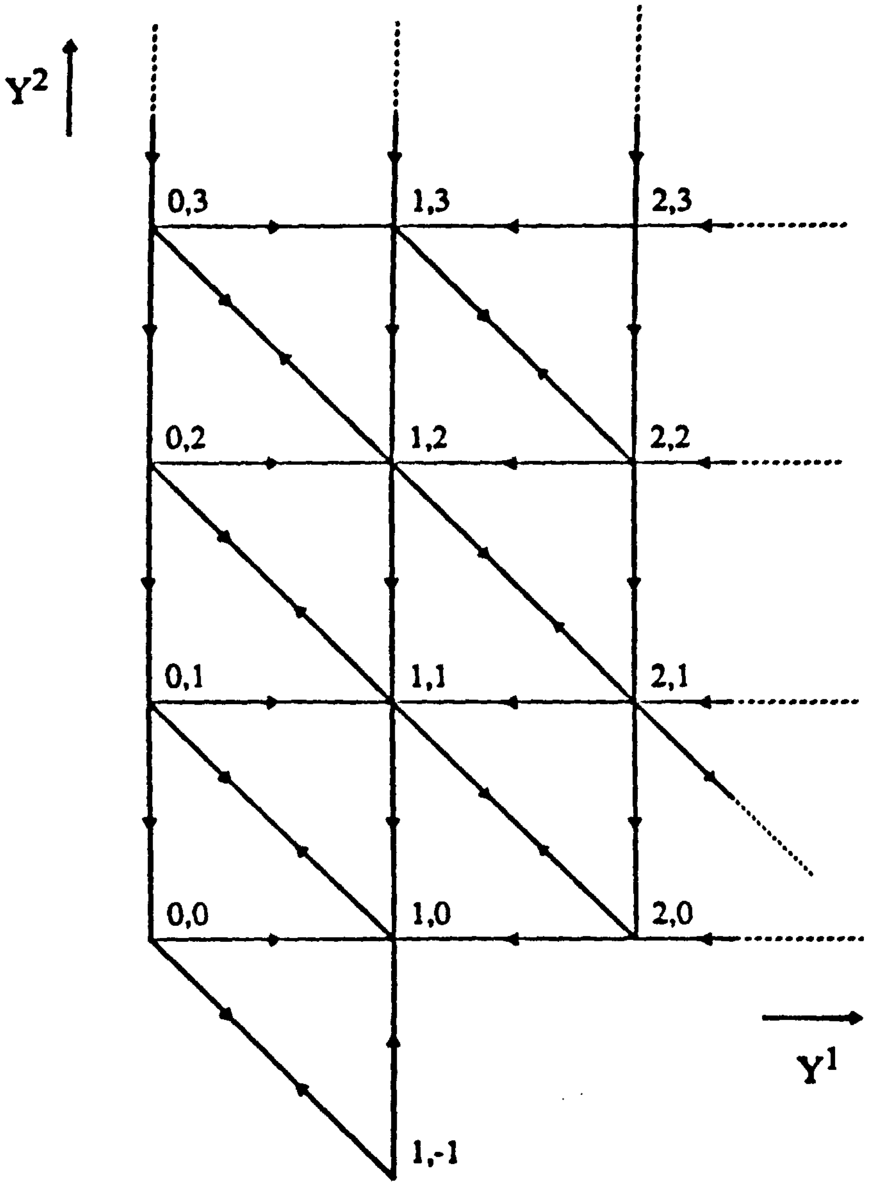

Figure 2.