Monotonicity of the throughput of a closed Erlang queueing network in the number of jobs


Document status and date:
Published: 01/01/1987

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 13. May. 2021
Monotonicity of the throughput of a closed Erlang queueing network in the number of jobs

by

Ivo Adan and Jan van der Wal

Eindhoven, The Netherlands
January 1987
Monotonicity of the throughput of a closed Erlang queueing network in the number of jobs.

by

Ivo Adan
Jan van der Wal

ABSTRACT

It is shown that the throughput of a closed Erlang queueing network is nondecreasing in the number of jobs.

1. Introduction.

Recently a number of papers appeared on the problem of establishing the intuitively obvious result that the throughput in a closed queueing network is nondecreasing in the number of jobs. All papers, see e.g. Robertazzi and Lazar [1985], Suri [1985] and Yao [1985], deal with productform networks and use this productform explicitly. In Van der Wal [1985] the exponential network was treated without using the productform character of the equilibrium distribution.

In this note we show that with some modifications the proof in Van der Wal [1985] can be extended to the closed network with Erlang service times. The proof is based on the relation between continuous and discrete time Markov chains and uses mathematical induction.

The paper is organized as follows. In section 2 the model and some notations are introduced, and the main theorem is formulated. Section 3 presents some well-known results on the relation between continuous and discrete time Markov chains with rewards. The proof of the main theorem is given in section 4. To complete this proof, we use a second theorem. Section 5 contains the proof of this theorem.

2. The model.

We consider a closed queueing network with \( N \) single server stations. The service time is Erlang distributed, i.e. the service time at station \( i \) consists of \( L_i \) independent exponential phases with mean \( \mu_i^{-1} \). The routing of the jobs is determined by the irreducible matrix \( P \) with elements \( p_{ij} \) indicating the probability that a job after its completion at queue \( i \) jumps to queue \( j \).

The state of the system can be characterized by the vector \( (k, l) = (k_1, ..., k_N, l_1, ..., l_N) \) with \( k_i \) the number of jobs in station \( i \) and \( l_i \) the phase of the job which is being served in station \( i \). We shall suppose that in an empty station the phase is 1. Consequently the phase is always 1 if a job has been completed, also if the station is empty afterwards. The set of all such state vectors with \( k_i \geq 0, \sum k_i = K, 1 \leq l \leq l_i \) and \( l_i = 1 \) if \( k_i = 0 \) will be denoted by \( S(K) \). So \( S(K) \) is the set of all possible states for the queueing network containing \( K \) jobs. The queueing network model gives rise to an irreducible continuous time Markov chain if \( P \) is irreducible (as we already assumed).

If phase \( j \) at station \( i \) has been completed one receives a reward \( R(i,j) \). The average reward per unit time will be denoted by \( G(K) \). So
\[ G(K) = \sum_{(k,l) \in S(K)} p(k,l) \sum_{i} \mu_i \epsilon(k_i) R(i,i_i), \]

where \( p(k,l) \) is the limiting probability of the network being in state \((k,l)\) and \( \epsilon(k) = 0 \) if \( k = 0 \) and 1 elsewhere.

The definition of throughput is more or less arbitrary. Due to the Markovian routing all throughput notions differ only by a multiplicative constant. If we define throughput as the total number of service completions in the network then we should define \( R(i,j) = 1 \) if \( j = I_i \) and 0 otherwise, \( i = 1, \ldots, N. \)

Now the monotonicity can be stated as

**Theorem 1.**

If

\[ R(i,j) \geq 0 \text{ for all } i \text{ and } j \]

then

\[ G(K+1) \geq G(K) \text{ for all } K = 0, 1, 2, \ldots. \]

3. Preliminaires.

Let \( Q \) be the generator of an irreducible finite state Markov chain with reward rate \( r(s) \) if the system is in state \( s \). We denote this chain by \((Q,r)\). With this continuous time chain one may associate a discrete time Markov chain with transition matrix \( R = I + \alpha Q \), where \( \alpha > 0 \) is a constant such that \( R \) is nonnegative, and with immediate reward per period \( \alpha r(s) \) if the system is in state \( s \).

The irreducible discrete Markov chain will be denoted by \((R, \alpha r)\). Then we have the following well-known results (see Van der Wal [1985])

**Lemma 1.**

The equilibrium distribution of the chains \((Q,r)\) and \((R, \alpha r)\) are identical. Hence the average reward per unit time for \((Q,r)\) and the average reward per period for \((R, \alpha r)\) differ only from a multiplicative constant \( \alpha \).

And

**Lemma 2.**

Let \((R,r)\) be an irreducible discrete time Markov chain with rewards, and let \( g \) be the average reward per period for this chain. Let further \( V^n \) be the \( n \)-period reward vector, so

\[ V^n = \sum_{t=0}^{n-1} R^t r \]

and also

\[ V^{n+1} = r + R V^n \]
Then for each state $i$

$$\lim_{n \to \infty} n^{-1}V^n(i) = g$$

An immediate consequence of this lemma is

**Lemma 3.**

Let $(P_1,r_1)$ and $(P_2,r_2)$ be two irreducible Markov chains with average reward per period $g_1$ and $g_2$ respectively. (The chains need not have the same number of states). Let $V_1$ and $V_2$ be the $n$-period reward vector for chain 1 and 2 respectively. If for some state $x$ in chain 1 and some state $y$ in chain 2 we have

$$V_1^x(x) \geq V_2^y(y) \text{ for all } n,$$

then

$$g_1 \geq g_2$$

So this lemma enables us to compare different chains, in our case networks with $K$ and $K+1$ jobs.


In order to prove $G(K+1) \geq G(K)$ we consider the two continuous time Markov chains with rewards $(Q_K, r_K)$ and $(Q_{K+1}, r_{K+1})$. Here $Q_L$ is the generator for the network with $L$ jobs and $r_L$ is the reward structure defined by

$$r_L(k, \bot) = \sum_{i} \mu_i \epsilon(k_i) R(i, l_i) (k, \bot) \in S(L).$$

Related to these continuous time chains we define the discrete time chains

$$(1 + \alpha Q_K, \alpha r_K) \text{ and } (1 + \alpha Q_{K+1}, \alpha r_{K+1})$$

with $\alpha > 0$, but sufficiently small for $1 + \alpha Q_K$ and $1 + \alpha Q_{K+1}$ to be nonnegative. Now define $V_K^m$ and $V_{K+1}^m$ to be the $n$-period reward vectors for the two chains. Then according to lemma 3 it suffices to show that for some state $x \in S(K)$ and $y \in S(K+1)$

$$V_{K+1}^m(y) \geq V_K^m(x) \text{ for all } m.$$

By induction we shall prove for all $m$

$$V_{K+1}^m(k + \epsilon_r, \bot) \geq V_K^m(k, \bot) \text{ for all } (k, \bot) \in S(K) \text{ and } r = 1, \ldots, N,$$

where $\epsilon_r$ denotes the $r$-th unit vector: $(0, \ldots, 0, 1, 0, \ldots, 0)$ with the 1 on place $r$. 
Since \( V^0 = 0 \) inequality (2) trivially holds for \( m = 0 \). Assuming that (2) holds for \( m = n \) we prove it for \( m = n+1 \). Using recursion (1) we can express \( V_{k+1}^{n+1}(k, \epsilon_r, \perp) \) and \( V_k^{n+1}(k, \perp) \) in terms of \( V_{k+1}^n \) and \( V_k^n \).

First \( V_{k+1}^{n+1}(k, \perp) \).

\[
V_{k+1}^{n+1}(k, \perp) = \alpha \sum_i \mu_i \epsilon(k_i) R(i, l_i)
+ \alpha \sum_i \mu_i \epsilon(k_i) \{ \\
\delta(l_i - L_i) \sum_j p_{ij} V_{k+1}^n(k + \epsilon_r, \epsilon_i + \epsilon_j, \perp - (L_i - 1)\epsilon_i) \\
+ (1 - \delta(l_i - L_i)) V_{k+1}^n(k, \perp + \epsilon_i) \}
+ (1 - \alpha \sum_i \mu_i \epsilon(k_i)) V_k^n(k, \perp)
\]

where \( \delta(k) = 1 \) if \( k = 0 \) and 0 elsewhere.

And

\[
V_{k+1}^{n+1}(k + \epsilon_r, \perp) = \alpha \sum_i \mu_i \epsilon(k_i) R(i, l_i) + \alpha \mu_r R(r, l_r)
+ \alpha \sum_i \mu_i \epsilon(k_i) \{ \\
\delta(l_i - L_i) \sum_j p_{ij} V_{k+1}^n(k + \epsilon_r, \epsilon_i + \epsilon_j, \perp - (L_i - 1)\epsilon_i) \\
+ (1 - \delta(l_i - L_i)) V_{k+1}^n(k + \epsilon_r, \perp + \epsilon_i) \}
+ \alpha \mu_r \{ \\
\delta(l_r - L_r) \sum_j p_{ij} V_k^n(k + \epsilon_r, \epsilon_j, \perp - (L_r - 1)\epsilon_r) \\
+ (1 - \delta(l_r - L_r)) V_{k+1}^n(k + \epsilon_r, \perp + \epsilon_r) \}
+ (1 - \alpha \sum_i \mu_i \epsilon(k_i) - \alpha \mu_r) V_{k+1}^n(k + \epsilon_r, \perp)
\]

We rewrite this relation as

\[
V_{k+1}^{n+1}(k + \epsilon_r, \perp) = \alpha \sum_i \mu_i \epsilon(k_i) R(i, l_i)
+ \alpha \sum_i \mu_i \epsilon(k_i) \{ \\
\delta(l_i - L_i) \sum_j p_{ij} V_{k+1}^n(k + \epsilon_r, \epsilon_i + \epsilon_j, \perp - (L_i - 1)\epsilon_i) \\
+ (1 - \delta(l_i - L_i)) V_{k+1}^n(k + \epsilon_r, \perp + \epsilon_i) \}
+ (1 - \alpha \sum_i \mu_i \epsilon(k_i)) V_k^n(k + \epsilon_r, \perp)
+ \alpha \mu_r (1 - \epsilon(k_i)) \}
\]
Using

\( R(r,i_r) \)

\[ + \delta(l_r - L_r) \sum_j p_{ij} V^p_{k+1}(k + e_{i_r} . \perp - (L_r - 1)e_{i_r}) \]

\[ + (1 - \delta(l_r - L_r)) V^p_{k+1}(k + e_{i_r} . \perp + e_{i_r}) \]

\[ - V^p_{k+1}(k + e_{i_r} . \perp) \}

Using

\[ V^p_{k+1}(k + e_{i_r} - e_{i} + e_{j} . \perp - (L_r - 1)e_{i}) \geq V^p_{k}(k - e_{i} + e_{j} . \perp - (L_i - 1)e_{i}) \]

(the induction hypothesis for \( m = n \))

\[ V^p_{k+1}(k + e_{i_r} . \perp + e_{i}) \geq V^p_{k}(k . \perp + e_{i}) \]

(the induction hypothesis for \( m = n \))

\[ (1 - \alpha \sum_i \mu_i \epsilon(k_i)) V^p_{k+1}(k + e_{i_r} . \perp) \geq (1 - \alpha \sum_i \mu_i \epsilon(k_i)) V^p_{k}(k . \perp) \]

(the induction hypothesis for \( m = n \))

we see that

\[ V^p_{k+1}(k + e_{i_r} . \perp) \geq V^p_{k}(k . \perp) \]

\[ + \alpha \mu_r (1 - \epsilon(k_r)) \}

\[ R(r,i_r) \]

\[ + \delta(l_r - L_r) \sum_j p_{ij} V^p_{k+1}(k + e_{i_r} . \perp - (L_r - 1)e_{i_r}) \]

\[ + (1 - \delta(l_r - L_r)) V^p_{k+1}(k + e_{i_r} . \perp + e_{i_r}) \]

\[ - V^p_{k+1}(k + e_{i_r} . \perp) \}

Clearly (2) holds for \( m = n+1 \) if we prove

**Theorem 2.**

If

\[ R(i,j) \geq 0 \text{ for all } i \text{ and } j \]

then for all \( m \) and \( r \)

\[ R(r,i_r) \]

\[ + \delta(l_r - L_r) \sum_s p_{rs} V^p_{k+1}(k + e_{s} . \perp - (L_r - 1)e_{i_r}) \]


We can interpret this theorem as follows. Suppose that we have two possibilities if a phase has been completed. The first possibility is to make the jump to the next phase or queue and receive a reward. The second possibility is to do the phase all over again and receive no reward. Now theorem 2 states that we should prefer the first possibility. We shall prove theorem 2 in the next section.

So (2) holds for all \(m\), whence, by lemma 3.

\[ G(K+1) \geq G(K). \]

5. Proof of theorem 2.

For \(m = 0\) inequality (6) holds because

\[ R(r_l_l) \geq 0 \quad \text{(the main assumption)} \]

Assuming that inequality (6) holds for \(m = n\) we prove it for \(m = n+1\). We do this in two parts. In the first part we assume that \(l_r < l_r\) and in the second part we assume that \(l_r = l_r\).

The first part.

In case \(l_r < l_r\) we have to prove

\[ R(r_l_l) + \sum_{i \leq \sigma} \mu_i e(k_i) R(l_{i1}, l) + \alpha \mu_r R(r_l_l) \]

Using the recursion (1) we can express \(V_{K+1}^{0+1}(k + e_r, l + e_r)\) and \(V_{K+1}^{0+1}(k + e_r, l)\) in terms of \(V_{K+1}^{0+1}\).

First \(V_{K+1}^{0+1}(k + e_r, l)\).

\[
V_{K+1}^{0+1}(k + e_r, l) = \alpha \sum_{i \neq j} \mu_i e(k_i) R(l_{i1}) + \alpha \mu_r R(r_l_l) + \delta(l_r - l_r) \sum_{j} p_{ij} V_{K+1}^{0+1}(k + e_r, l_{j}, l_{j} + e_{j} - (l_j - 1)e_{j})
\]

And
\[ V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_r) = \alpha \sum_{i \in r} \mu_i e(k_i) R(i, l_i) + \alpha \mu_r R(r, l_r + 1) \]
\[ + \alpha \sum_{i \in r} \mu_i e(k_i) \}
\[ \delta(l_i - L_i) \sum_j p_{ij} V_{k+1}^{\text{opt}}(k + \varepsilon_r - \varepsilon_i + \varepsilon_j, \perp + \varepsilon_r - (L_i - 1)\varepsilon_i) \]
\[ + (1 - \delta(l_i - L_i)) V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_i) \} \]
\[ + \alpha \mu_r \}
\[ \delta(l_r + 1 - L_r) \sum_j p_{ij} V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_r + \varepsilon_i) \}
\[ + \alpha \mu_r \}
\[ \delta(l_r + 1 - L_r) \sum_j p_{ij} V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_r + \varepsilon_j, \perp + \varepsilon_r - (L_r - 1)\varepsilon_r) \}
\[ + (1 - \delta(l_r + 1 - L_r)) V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_r + 2\varepsilon_r) \} \]
\[ + (1 - \alpha \sum_{i \in r} \mu_i e(k_i) - \alpha \mu_r) V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_r) \]

If we use this relation and rewrite \( R(r, l_r) \) as
\[ \alpha \mu_r R(r, l_r) \]
\[ + \alpha \sum_{i \in r} \mu_i e(k_i) R(r, l_r) \]
\[ + (1 - \alpha \sum_{i \in r} \mu_i e(k_i) - \alpha \mu_r) R(r, l_r) \]

the left hand side of (7) results in
\[ R(r, l_r) + V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_r) = \alpha \sum_{i \in r} \mu_i e(k_i) R(i, l_i) + \alpha \mu_r R(r, l_r) \]
\[ + \alpha \sum_{i \in r} \mu_i e(k_i) \}
\[ \delta(l_i - L_i) \sum_j p_{ij} (R(r, l_r) + V_{k+1}^{\text{opt}}(k + \varepsilon_r - \varepsilon_i + \varepsilon_j, \perp + \varepsilon_r - (L_i - 1)\varepsilon_i)) \]
\[ + (1 - \delta(l_i - L_i)) (R(r, l_r) + V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_i)) \} \]
\[ + \alpha \mu_r \}
\[ R(r, l_r + 1) \]
\[ + \delta(l_r + 1 - L_r) \sum_j p_{ij} V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_r - (L_r - 1)\varepsilon_r) \]
\[ + (1 - \delta(l_r + 1 - L_r)) V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + 2\varepsilon_r) \} \]
\[ + (1 - \alpha \sum_{i \in r} \mu_i e(k_i) - \alpha \mu_r) (R(r, l_r) + V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_r)) \]

Using
\[ (8) \quad R(r, l_r) + V_{k+1}^{\text{opt}}(k + \varepsilon_r, \perp + \varepsilon_i + \varepsilon_j, \perp + \varepsilon_r - (L_i - 1)\varepsilon_i) \]
\[ \begin{align*}
\Rightarrow & \; V_{K+1}^\varphi(k + e_r - e_i + e_j - (L_i - 1)e_i) \\
& \text{(induction hypothesis for } m = n) \\
(9) & \; R(r,l_r) + V_{K+1}^\varphi(k + e_r, l + e_r + e_i) \\
& \Rightarrow V_{K+1}^\varphi(k + e_r, l + e_i) \\
& \text{(induction hypothesis for } m = n) \\
(10) & \; R(r,l_r + 1) \\
& + \delta(l_r + 1 - L_r) \sum_j p_j V_{K+1}^\varphi(k + e_j, l + e_r - (L_r - 1)e_r) \\
& + (1 - \delta(l_r + 1 - L_r)) V_{K+1}^\varphi(k + e_r, l + 2e_r) \\
& \Rightarrow V_{K+1}^\varphi(k + e_r, l + e_r) \\
& \text{(induction hypothesis for } m = n) \\
(11) & \; R(r,l_r) + V_{K+1}^\varphi(k + e_r, l + e_r) \\
& \Rightarrow V_{K+1}^\varphi(k + e_r, l) \\
& \text{(induction hypothesis for } m = n) \\
\text{we see that} \\
R(r,l_r) + V_{K+1}^\varphi(k + e_r, l + e_r) \geq V_{K+1}^\varphi(k + e_r, l) \\
\text{that is, (7) holds.} \\
The second part. \\
In case } l_r = L_r \text{ we have to prove} \\
(12) & \; R(r,l_r) + \sum_i p_i V_{K+1}^\varphi(k + e_r, l - (L_r - 1)e_r) \geq V_{K+1}^\varphi(k + e_r, l) \\
& \text{Using the recursion (1) we can express } V_{K+1}^\varphi(k + e_r, l - (L_r - 1)e_r) \text{ and } V_{K+1}^\varphi(k + e_r, l) \\
& \text{in terms of } V_{K+1}^\varphi. \\
& \text{First } V_{K+1}^\varphi(k + e_r, l). \\
& V_{K+1}^\varphi(k + e_r, l) = \alpha \sum_{i \neq r} \mu_i \varepsilon(k_i) R(i,l_i) + \alpha \mu_r R(r,l_r) \\
& + \alpha \sum_{i \neq r} \mu_i \varepsilon(k_i) \{ \\
& \delta(l_i - L_i) \sum_j p_j V_{K+1}^\varphi(k + e_r - e_i + e_j, l - (L_i - 1)e_i) \\
& + (1 - \delta(l_i - L_i)) V_{K+1}^\varphi(k + e_r, l + e_i) \} 
\end{align*} \]
\[ + \alpha \mu_r \sum_j p_{ij} V_{k+1}^{\delta}(k + e_j, \bot - (L_r - 1)e_i) \\
+ (1 - \alpha \sum_{i \neq r} \mu_i \epsilon(k_i) - \alpha \mu_r) V_{k+1}^{\delta}(k + e_r, \bot) \]

And with \( \delta_{ij} = 1 \) if \( i = j \) and 0 elsewhere.

\[ V_{k+1}^{\delta}(k + e_s, \bot - (L_r - 1)e_r) = \alpha \sum_{i \neq r} \mu_i \epsilon(k_i + \delta_{1r}) R(i,l) + \alpha \mu_r \epsilon(k_r + \delta_{2r}) R(r,1) \]

If we split the summation over all \( i \neq r \) in summation over all \( i \neq r \) with \( k_i > 0 \) and over all \( i \neq r \) with \( k_i = 0 \) and order terms, we see that

\[ V_{k+1}^{\delta}(k + e_s, \bot - (L_r - 1)e_r) = \alpha \sum_{i \neq r} \mu_i R(i,l) \]

\[ + \alpha \sum_{\begin{array}{c} i \neq r \\ 0 < k_i \end{array}} \mu_i \}

\[ \delta(l_i - L_i) \sum_j p_{ij} V_{k+1}^{\delta}(k + e_s - e_i + e_j, \bot - (L_r - 1)e_r - (L_i - 1)e_i) \\
+ (1 - \delta(l_i - L_i)) V_{k+1}^{\delta}(k + e_s, \bot - (L_r - 1)e_r + e_i) \]
Using

\[ \sum_{i \in \mathcal{R}} f(i) = \sum_{i \in \mathcal{R}} \varepsilon(k_i) f(i), \text{ where } f \text{ is a function of } i. \]

(13)

\[ \sum_{i \in \mathcal{R}} f(i) = \sum_{i \in \mathcal{R}} \varepsilon(k_i) f(i), \text{ where } f \text{ is a function of } i. \]

(14) \quad R(i, l_i)

+ \delta (l_i - L_i) \sum_j p_{j} V_{k+1}^a( \mathbf{k} + \mathbf{e}_s - \mathbf{e}_r + \mathbf{e}_j, \bot - (L_r - 1)\mathbf{e}_r - (L_r - 1)\mathbf{e}_i )

+ (1 - \delta (l_i - L_i)) V_{k+1}^a( \mathbf{k} + \mathbf{e}_s, \bot - (L_r - 1)\mathbf{e}_r + \mathbf{e}_r )

\equiv 0

\text{(the induction hypothesis for } m = n)\]

(15) \quad R(r, 1)

+ \delta (1 - L_r) \sum_j p_{j} V_{k+1}^a( \mathbf{k} + \mathbf{e}_s - \mathbf{e}_r + \mathbf{e}_j, \bot - (L_r - 1)\mathbf{e}_r - (L_r - 1)\mathbf{e}_i )

+ (1 - \delta (1 - L_r)) V_{k+1}^a( \mathbf{k} + \mathbf{e}_s, \bot - (L_r - 1)\mathbf{e}_r + \mathbf{e}_r )

\equiv 0

\text{(the induction hypothesis for } m = n)\]

we see that

(16) \quad V_{k+1}^a( \mathbf{k} + \mathbf{e}_s, \bot - (L_r - 1)\mathbf{e}_r ) \equiv \alpha \sum_{i \in \mathcal{R}} \mu_i \varepsilon(k_i) R(i, i)
\[ + \alpha \sum_{i \neq r} \mu_i \epsilon(k_i) \{ \]

\[ \delta(l_i - L_i) \sum_j p_{ij} V_{k+1}^i (k + e_s - e_i + e_j, \bot - (L_r - 1)e_r - (L_i - 1)e_i) \]

\[ + (1 - \delta(l_i - L_i)) V_{k+1}^i (k + e_s, \bot - (L_r - 1)e_r + e_i) \} \]

\[ + (1 - \alpha \sum_{i \neq r} \mu_i \epsilon(k_i)) V_{k+1}^i (k + e_s, \bot - (L_r - 1)e_r) \]

Applying (16) to the left hand side of (12) results in

\[ R(r,i) + \sum_{s} p_{rs} V_{k+1}^s (k + e_s, \bot - (L_r - 1)e_r) \geq R(r,i), \]

\[ + \sum_{s} p_{rs} \alpha \sum_{i \neq r} \mu_i \epsilon(k_i) R(i,i), \]

\[ + \sum_{s} p_{rs} \alpha \sum_{i \neq r} \mu_i \epsilon(k_i) \{
\]

\[ \delta(l_i - L_i) \sum_j p_{ij} V_{k+1}^i (k + e_s - e_i + e_j, \bot - (L_r - 1)e_r - (L_i - 1)e_i) \]

\[ + (1 - \delta(l_i - L_i)) V_{k+1}^i (k + e_s, \bot - (L_r - 1)e_r + e_i) \} \]

\[ + \sum_{s} p_{rs} (1 - \alpha \sum_{i \neq r} \mu_i \epsilon(k_i)) V_{k+1}^s (k + e_s, \bot - (L_r - 1)e_r) \]

If we change the summation over all \( s \) and over all \( i \neq r \) and if we add

\[ 0 = \alpha \mu, s \sum_{j \neq i} p_{rs} V_{k+1}^j (k + e_s, \bot - (L_r - 1)e_r) \]

\[ - \alpha \mu_i s \sum_{j \neq i} p_{rs} V_{k+1}^j (k + e_s, \bot - (L_r - 1)e_r) \]

we see that

\[ R(r,i) + \sum_{s} p_{rs} V_{k+1}^s (k + e_s, \bot - (L_r - 1)e_r) \geq R(r,i), \]

\[ + \alpha \sum_{i \neq r} \mu_i \epsilon(k_i) R(i,i), \]

\[ + \alpha \sum_{i \neq r} \mu_i \epsilon(k_i) \{
\]

\[ \delta(l_i - L_i) \sum_j p_{ij} \sum_s p_{rs} V_{k+1}^i (k + e_s - e_i + e_j, \bot - (L_r - 1)e_r - (L_i - 1)e_i) \]

\[ + (1 - \delta(l_i - L_i)) \sum_s p_{rs} V_{k+1}^i (k + e_s, \bot - (L_r - 1)e_r + e_i) \} \]

\[ + \alpha \mu, s \sum_{j \neq i} p_{rs} V_{k+1}^j (k + e_s, \bot - (L_r - 1)e_r) \]

\[ + (1 - \alpha \sum_{i \neq r} \mu_i \epsilon(k_i) - \alpha \mu_i) \sum_{s} p_{rs} V_{k+1}^s (k + e_s, \bot - (L_r - 1)e_r) \]

Rewriting \( R(r,i) \) as
\[ \alpha \mu_r R(r, l_r) \]
\[ + \alpha \sum_{i \in r} \mu_i \epsilon(k_i) R(r, l_r) \]
\[ + (1 - \alpha \sum_{i \in r} \mu_i \epsilon(k_i) - \alpha \mu_r) R(r, l_r) \]

results in

\[
R(r, l_r) + \sum_s p_{rs} V_{k+1}^s (k + e_s, \perp - (L_r - 1)e_r) \geq \alpha \sum_{i \in r} \mu_i \epsilon(k_i) R(i, l_i) + \alpha \mu_r R(r, l_r) \\
+ \alpha \sum_{i \in r} \mu_i \epsilon(k_i) \{ \\
\delta(l_i - L_i) \sum_j p_{ij} \}
\]

\[
R(r, l_r) + \sum_s p_{rs} V_{k+1}^s (k + e_s, \perp - (L_r - 1)e_r) \\
+ (1 - \delta(l_i - L_i)) ( \\
R(r, l_r) + \sum_s p_{rs} V_{k+1}^s (k + e_s, \perp - (L_r - 1)e_r) ) \}
\]

\[
+ \alpha \mu_r \sum_s p_{rs} V_{k+1}^s (k + e_s, \perp -(L_r - 1)e_r) \\
+ (1 - \alpha \sum_{i \in r} \mu_i \epsilon(k_i) - \alpha \mu_r) (R(r, l_r) + \sum_s p_{rs} V_{k+1}^s (k + e_s, \perp -(L_r - 1)e_r))
\]

Using

(17) \[ R(r, l_r) + \sum_s p_{rs} V_{k+1}^s (k + e_s, \perp - (L_r - 1)e_r - (L_i - 1)e_i) \]
\[ \geq V_{k+1}^s (k + e_r, \perp - (L_r - 1)e_r - (L_i - 1)e_i) \]

(the induction hypothesis for \( m = n \))

(18) \[ R(r, l_r) + \sum_s p_{rs} V_{k+1}^s (k + e_s, \perp - (L_r - 1)e_r + e_i) \]
\[ \geq V_{k+1}^s (k + e_r, \perp + e_i) \]

(the induction hypothesis for \( m = n \))

(19) \[ R(r, l_r) + \sum_s p_{rs} V_{k+1}^s (k + e_s, \perp - (L_r - 1)e_r) \]
\[ \geq V_{k+1}^s (k + e_r, \perp) \]

(the induction hypothesis for \( m = n \))

we see that
that is, (12) holds.

This completes the proof of theorem 2, and thus the proof of theorem 1.
References.


Monotonicity of the throughput of a closed exponential queueing network in the number of jobs. Eindhoven University of Technology, Department of Mathematics and Computing Science, Memorandum COSOR 85-21.