91-13

Faculteit der Economische Wetenschappen en Econometrie

ET

05348

Serie Research Memoranda

Product Forms for Queueing Networks with Limited Clusters

Nico M. van Dijk

Research Memorandum 1991 - 13
January 1991

vrije Universiteit *amsterdam*

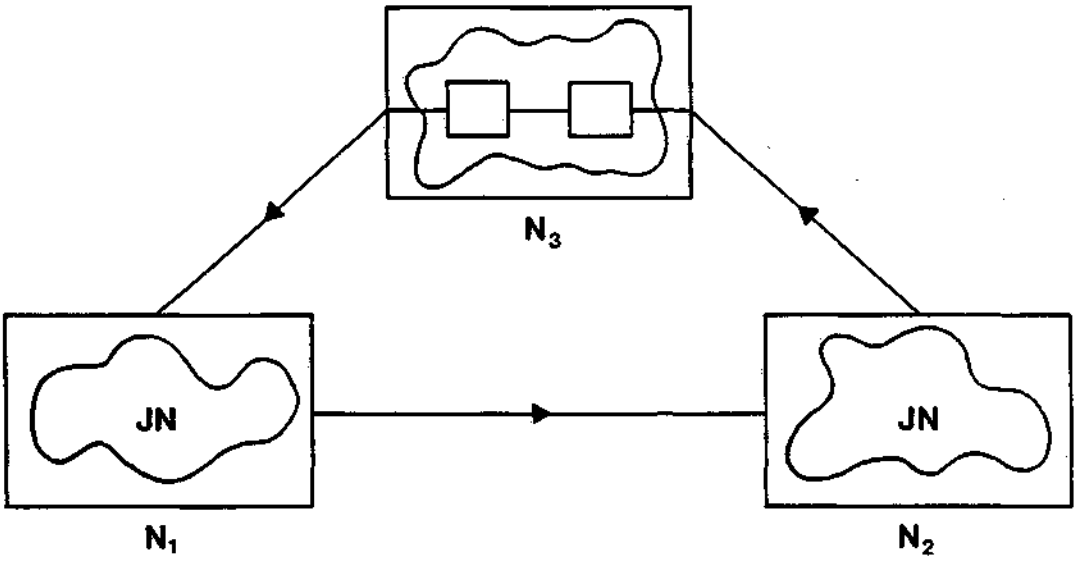
BIBL. VRIJE UNIVERSITEIT AMSTERDAM
E-T.
05348

PRODUCT FORMS FOR QUEUEING NETWORKS

WITH

LIMITED CLUSTERS

Nico M. van Dijk *
Bond University, Queensland, Australia



***Visiting Address:** School of Information and Computing Sciences, Bond University, Gold Coast, Queensland, 4229, Australia
Permanent Address: Department of Econometrics, Free University, The Netherlands

Date: Thursday 13 December 1990

Extended Abstract

Finite capacity constraints on groups or clusters of service stations are most common in queueing network applications, for example in

- *telecommunications* due to common trunk groups or switch devices
- *computer networking* due to store and forward buffers for parallel processors
- *flexible manufacturing* due to storage buffers between groups of workstations.

This paper aims to provide a unifying framework and simple basic insights from which one can conclude whether or not a queueing network with finite or limited station clusters can exhibit a product form for the steady state population distribution.

First, an extended overview and discussion of known results, most notably for queueing networks with finite individual stations will be given. This can be regarded as a comprehensive survey of product form results for queueing networks with blocking.

Next, the principle of station balance is shown to be a necessary and sufficient condition for product form results in queueing networks with state dependent routing. Roughly, this principle reads as:

the rate (or flow) out of any station =
the rate (or flow) into that same station

Particularly, this key-principle has the appealing practical properties that:

(i) It can be characterized in terms of local solutions of local state dependent routing or traffic equations. These solutions can usually be obtained explicitly.

(ii) It leads to simple physical insights from which at an intuitive or down-to-earth level one can quickly conclude whether or not a network can have a product form.

Though notions of partial balances are known to be related to product form results, neither of the statements (i) and (ii) has been reported, advocated or exploited as a practical tool to conclude product forms.

The station balance principle will then be combined with a similar principle at cluster level, which one could call cluster balance reading as:

the rate out of any cluster =
the rate into that cluster,

in order to extend product form results for networks with finite stations to networks with capacity constraints for clusters of stations. An instructive example (see example 4.1) illustrates the various implications and how the principles can be used at a non-mathematical practical level. In this example, both routing within clusters and between clusters is non-reversible.

To further illustrate the potential and the use of the local routing equations a fair number of novel product form examples with restricted clusters will then be given. These include novel features as

- weak reversible routing (section 5.1)
- reversible cluster routing with service delays (section 5.2)
- non-reversible cluster routing with conservative blocking (section 5.3.1)
- cyclic cluster routing with minimal capacity constraints (section 5.3.2)

Typical applications are

(i) two-phase cluster structures, such as of two communicating metropolitan area networks, of central processor connected to a set of users, or of material handling systems in manufacturing, and

(ii) cyclic connected clusters such as representing parallel processors, memory modules and read/write devices in computer networks, groups of switching devices in packet switching communications or groups of workstations in manufacturing systems.

Extensions to nested or hierarchical blocking structures and randomized routing are also possible (see section 5.5)

Finally, it will be shown that the same product form expressions can be concluded for both the "stop or communications protocol", under which services of total clusters are stopped or delayed, and the "recirculate protocol", under which blocked jobs are recirculated into clusters.

Keywords

Queueing network, Limited cluster, Product form, Station balance, Reversibility, Weak reversible routing, Local traffic equations, Reversible cluster routing, Non-reversible cluster routing, Stop protocol, Recirculate protocol, Conservative blocking, Nested blocking.

Contents

Extended Abstract 1

1 Introduction and Summary 5

2 Background and Literature 9

2.1 Finite Stations 9

2.2 Communications Blocking 13

2.3 Limited Clusters 13

3 Station Balance and Product Forms 14

3.1 Model 14

3.2 Station Balance 15

3.3 Discussion of Results 18

3.4 An Instructive Example: 19

3.5 Special case: Reversible routing 22

4 Limited Clusters 23

4.1 Model 23

4.2 Cluster Balance 24

4.3 An Illustrative Example. 25

4.4 Discussion and Summary 30

5 Applications 31

5.1 Weak Reversible Routing 33

5.2 Reversible Cluster Routing 38

5.3 Non-reversible Cluster Routing 43

5.3.1 Conservative blocking 43

5.3.2 Cyclic cluster routing; minimal workloads 45

5.3.3 One entry and departure station (cf [48]) 46

5.4 Nested Structure 48

6 Stop = Recirculate 51

6.1 Equivalence Results 51

6.2 Illustration of Results 55

7 Extensions 58

7.1 Open Systems. 58

7.2 Multi-job Classes. 58

7.3 Non-exponential Servicing. 58

8 Evaluation 59

9 References 61

1 Introduction and Summary

Background

Queueing network modeling has obtained a wide popularity for performance evaluation purposes in computer, communication and manufacturing engineering. Much of this success can most likely be attributed to

1. Jackson's celebrated product form expression and its various extensions which in principle enables a direct computation of relevant performance measures such as a throughput, system utilization or response time.

2. The efficient computational techniques developed such as MVA so as to efficiently compute these expressions also for very large networks.

3. The generic structure of queueing networks with jobs moving from one service stage to another in a possibly interfering manner, which applies to a large variety of applications in multiple disciplines such as most notably:

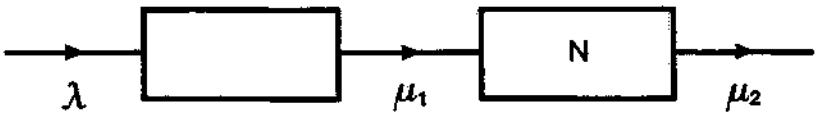
   - Telecommunications (Teletraffic): End-to-end connections for telephone calls or message transmissions are occupied for random transmission times (circuit switching).
   - Broad or multicasting: Messages are broadcasted to a set of neighbouring recipients (broadcasting) or multicasted to a specific set of destinations (multicasting) during some random amount of time.
   - Computer Networking: Programs or files are sent back and forth between processors, compilers, storage disks and users (terminals) according to fixed or load dependent probabilities with random processing times and queueing delays (packet switching).
   - Manufacturing: Parts or pallets are transported for being processed upon during random (or possibly fixed) amounts of time at various successive workstations.

This combination of explicit simple closed form expressions on the one hand and practical applications on the other hand, however, seems to be rather limited when blocking or state dependent routing are taken into account. These phenomena are most natural, for example, in

- Telecommunications: Transmissions from various origins usually share common finite trunk groups in (hierarchical) circuit switching.
- Broadcasting: Neighbouring transmitters may interfere by using the same carrier for broadcasting (CSMA)
- Computer Networking: Finite store and forward buffers are usually involved for parallel processors.

- Manufacturing: Finite capacity constraints or intermediate storage buffers are common in assembly lines.

Generally, product form results are know to be related to various notions of "balance" or "flow out = flow in" principles at some partial or detailed level. (cf [6], [16], [17], [27], [28], [33], [36], [39], [44], [45], [48], [55], [56], [59]). Unfortunately, realistic features such as blocking, resource sharing or dynamic routing will typically destroy these principles. For example, consider the assembly line example below and assume that the second station is saturated. Then



the flow out of station 1 := 0
the flow into station 1 :> 0.

Hence, a principle of flow balance per station necessarily fails at station 1. As a consequence, (as formally proven by theorem 3.1 in section 3.2), the system cannot exhibit a product form.

To the best of the author's knowledge, this simple physical insight of station balance so as to conclude, at a strictly non-mathematical basis, whether or not a concrete practical system can have a product form has not been reported or advocated in the literature.

In particular, as per this paper, this insight can be extended also to aggregations or groups of stations with finite capacity constraints. This will substantially enlarge the range of product form examples as well as provide further physical insight in the phenomenon of product forms.

Literature (Finite stations)

More precisely, product form results for queueing networks with blocking have been widely reported (e.g. [1], [2], [4], [5], [19], [27], [33], [42], [46], [59]) under the restrictive conditions of:

(i) A reversible routing

(ii) A finite constraint per station

Under condition (ii) some special examples with non-reversible routing have also been obtained (cf. [19], [27], [28], [35], [46], [48]). However, from a practical point of view, under the condition of finite individual stations a reversible routing can more or less be regarded as necessary for product forms.

A detailed review of these and other related references will be presented in section 3 and specific remarks will be included on the most closely related references as we proceed.

Motivation

In contrast with the condition of finite individual stations, though, in practice one often encounters capacity constraints imposed on groups or clusters of stations. For example, in:

- Telecommunications: Messages for multiple destinations, and thus trunk groups or stations, may have to be switched along a common restricted switching device.
- Computer Networks: Multiple processors or stations of parallel processors may share a common finite resource, such as a compiler or data bank. Store and forward buffers of several processors can be pooled together.
- Manufacturing: Finite storage buffers for groups of workstations are natural in push-and-pull manufacturing lines. A material handling system for an assembly line with multiple machines may have a finite central controller.

Literature (Limited clusters)

For special cases of limited station clusters product form results have been reported (cf. [31], [35], [36], [46], [48]). However, as will be explained in detail in section 2, section 5.1 for reference [46] and section 5.4 for reference [48], all these results still require some form of reversibility, such as: a two-cluster structure ([31], [36]), a reversible entry-parallel-departure structure per cluster ([35], [48]), or a "local reversibility" condition [46].

Objective

This paper, in the first place aims to further investigate the potential of product form results for queueing networks with limited clusters of stations rather than individual stations. To this end first the notion of station balance will be studied in detail and shown to be a key-property both formally and intuitively to conclude product form results for queueing networks with state dependent routing or blocking. Though somewhat related formal results are certainly available in the literature, the necessary and sufficient form as presented herein as well as the simple physical insights that it provides and from which product form results can be concluded at a down-to-earth non-mathematical basis have not been reported. Next, the station balance notion is particularized to networks with limited clusters which leads to an additional notion of balance per cluster. This combination of station and cluster balance can also be employed not only at a formal mathematical but also at an intuitive down-to-earth physical basis as will be illustrated by an instructive example (section 4.3). In particular, when limited

clusters rather than stationsare involved, the range of product form results will so appear to be substantially larger than generaly percieved.

Summary of Results

The results can be summarized by

- A necessary and sufficient condition for a product form result to hold. This condition is given in terms of local solutions of state dependent traffic equations which can usually be obtained explicitly.
- Simple physical insights to conclude whether or not a queueing network with finite capacity constraints can have a product form.
- Various novel classes of product form examples when limited clusters are involved. More precisely, these include the cases of a
  - *Weak-reversible cluster routing* as an extension of the standard restrictive reversible routing or the recently reported 'local reversible' routing ([46]), while allowing servicing within a cluster to continue. Examples include transmitters connected by a carrier or bus in communication systems, clusters of parallel processors with single entry and departure nodes in computer networks or decentralised controlled clusters of workstations within an assembly line.
  - *A reversible routing* from *one* cluster *to another*, with general non-reversible routing within a cluster and under special stopping (or delay) protocols. Here one may typically think of two interconnected metropolitan area communications networks or two linked groups of not necessarily parallel processors in computer networks, or two connected local assembly lines in manufacturing.
  - *Non-reversible routing* both *between* and within *clusters*, but, with the stopping (or delay) protocols working at both cluster and station level. The above applications now extend to multiple and arbitrarily connected clusters, however, to the price of more conservative or secure blocking protocols such as to avoid more than one cluster to become saturated at the same time.

Further, a global survey and discussion of reported results will be provided so as to provide somewhat more perspective and insight in this practically relevant topic. Since this survey only concentrates on queueing networks with blocking, excellent references on extended product form results such as [17], [22], [44], [45], and [59], are not discussed in detail as they do not concern blocking.

Outline

First, in section 2 this global survey and a basic physical insight will be provided of reported product form results for queueing networks with blocking. The

key-principle of station balance, its characterization by local solutions of traffic equations and its physical insight is outlined and illustrated in section 3. Section 4 further explores the consequences of this principle for queueing networks with limited clusters. In particular, an instructive example to illustrate the various consequences and conclusions by simple physical insights is studied in detail. Section 5 brings the conditions down to several classes of applications, covering the above mentioned different cluster routings.

In section 6 it is shown that the product form results remain the same under both the "stop" and the "recirculate" protocol. This equivalence seems to reveal a 'novel' implicit feature of product form results and directly leads to further product form extensions for queueing networks with limited clusters.

To highlight the essential features, throughout the presentation will be restricted to closed exponential systems with one job-class. Section 7 will briefly discuss the possible extensions to open, non-exponential and multi-class networks. An evaluation completes the paper.

2 Background and Literature

As of today product form results are widely used for performance evaluation of Jacksonian queueing networks without blocking but also generally perceived not to be applicable for more realistic systems that do involve blocking phenomena. To give somewhat more insight in the status of this perception or up to what extent it is true, this section aims to give a global survey of state-of-the art.

Though elegant surveying bibliographies on queueing networks and particularly queueing networks with blocking have recently been written (cf. [40], [41]), to which references the interested reader is referred, the overview below will be more descriptive and specific merely concentrating on product form results.

Clearly an overview on this topic cannot be exhaustive and more than certainly specific applications most notably with a two-stage reversible structure must be available in the literature, such as in more practically oriented journals. The listing below, however, aims to provide a global overview of the essential structures and the associated references having revealed these explicitly, which do permit a product form result while taking into account blocking phenomena.

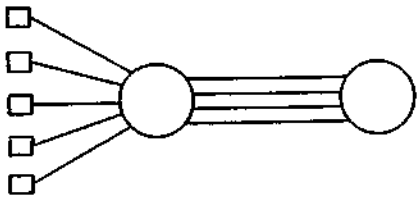
2.1 Finite Stations

In this subsection we consider the case in which only single stations have finite capacity constraints.

Two-phase Models

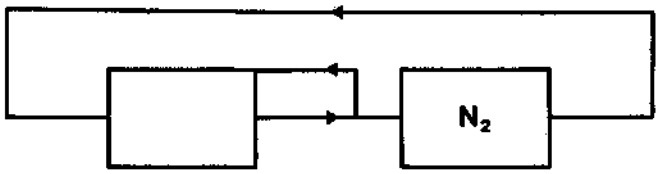
([7], [9], [10], [12], [13], [14], [20], [21], [25], [30], [32], [38], [49], [50], [51])

As most classical but also generic example, Engset's famous finite source loss system can already be seen as a first illustration of a two station queueing network with a finite capacity constraint at one station.



This example, for which a product form result has been reported under various extensions, most notably see [21], [30] and [55], clearly reveals the essence of partial or station balance for product forms as well as the importance of a reversible routing. As these notions play a key-role throughout this paper let us study this simple but characteristic example in some more detail.

An instructive example



Let both stations have infinite server capacities, that is each job present is in service, with exponential services with parameters μ_1 and μ_2 at station 1 and 2 respectively. Station 2 however has a finite storage constraint for no more than N_2 jobs where $N_2 < M$, the total number of jobs in the system. The global balance equations or "total rate out" = "total rate in" equations for the steady state distribution $\pi(n_1, n_2)$ of the number of jobs n_1 and n_2 at stations 1 and 2 are then expressed by

$$(2.1) \quad \begin{cases} \pi(n_1,n_2)\mu_2 1_{\{n_2>0\}} + \pi(n_1,n_2)\mu_1 1_{\{n_1>0\}} 1_{\{n_2<N_2\}} = \\ \\ \pi(n_1+1,n_2-1)\mu_1 1_{\{n_2>0\}} + \pi(n_1-1,n_2+1)\mu_2 1_{\{n_1>0\}} 1_{\{n_2<N_2\}} \end{cases}$$

where $1_{\{A\}} = 0$ if event A is satisfied and 0 otherwise. In particular, we note here that $1_{\{n_2<N_2\}}$ appears in the outrate due to blocking of the first station when $n_2 = N_2$, while also in the inrate to take into account that states are not admissible with $N_2 + 1$ jobs at station 2. When $n_2 = N_2$ the first terms of the rate out and rate in side thus have to be equal. That is,

> the rate out of state (n_1, n_2) due to a departure at station 1 =
> the rate into state (n_1, n_2) due to an arrival at station 1,

which thus has an interpretation of balance for station 1 separately. The second

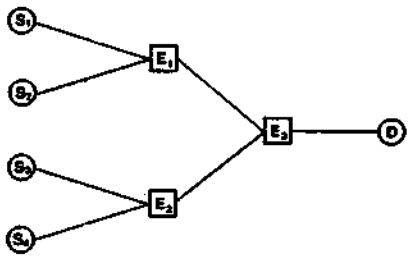
terms of the rate out and rate in have similar interpretations for station 2 when $n_2 < N_2$, while when $n_2 = N_2$ both are equal to 0. These detailed or partial balances per station 1 and 2 separately (that is, station balance equations) directly lead to the solution, with c a normalizing constant:

$$(2.2) \quad \pi(n_1, n_2) = c\frac{1}{n_1!}\left(\frac{1}{\mu_1}\right)^{n_1}\frac{1}{n_2!}\left(\frac{1}{\mu_2}\right)^{n_2} \qquad (n_2 \leq N_2, n_1 + n_2 = M)$$

Generalizations of this product form result with non-exponential and job (or source) dependent characteristics have been established in [22], [30] and and [55]. When multiple classes of jobs are involved while the blocking depends on the numbers of the various classes in a special manner, an extended product form can also be obtained. See [25], [32] under the condition of so-called "coordinate corvex blocking". Further, product form extensions for two-stage structures with multiple jobs changing at the same time are also possible (cf [10]).

A Special Application: Circuit switching communication networks

In particular, communications networks can frequently be seen as a two-stage queueing network. More precisely, a circuit switch structure for end-to-end communications can be seen as one service stage (Station 2) for the time of a transmission, while the finite (Engset-type) or infinite (Poissonian)source input, can be seen as a first stage (Station 1). Product form results are therefore obtainable as shown in [7], [14], [20].



Reversible Routing

(cf. [1], [2], [4], [5], [18], [19], [27], [33], [42], [47], [55], [57], [58], [59])

The two-station model can be extended to Jackson networks with finite stations provided the routing probabilities p_{ij} satisfy the reversibility conditions

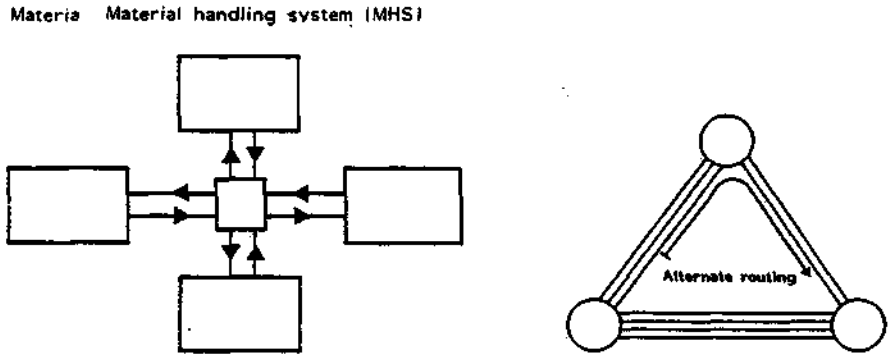
$$(2.3) \qquad \lambda_i p_{ij} = \lambda_j p_{ij}$$

for some set of $\{\lambda_j\}$ and all i, j. (Note that two-station models are included by $\lambda_1 = \lambda_2 = p_{12} = p_{21} = 1$). Under this routing condition Jackson's celebrated product form result was shown to remain valid also with finite capacity constraints N_i at stations i. More precisely, with $f_i(n_i)$ the service rate factor at station i when n_i jobs are present, the steady state distribution has the form

$$(2.4) \quad \pi(n_1, \ldots, n_N) = c\prod_i \left[\prod_{k=1}^{n_i} f_i(k)\right]^{-1} [\lambda_i/\mu_i]^{-n_i} \quad (n_i \leq N_i; i = 1, \ldots, N).$$

This result was first made explicit, along with a detailed investigation of reversible Markov chains and its consequences, in the famous book of Kelly [33]. The result has been reported also in various later references (e.g. [1], [2], [4], [5], [18], [19], [27]) Particularly, as per [33] and [42], any further state space limitation by truncation of the reversible model will remain reversible and retain a product form of the type (2.4).

(Special Applications) Two noteworthy practical applications within this category are the reversible structure of finite material handling systems in manufacturing (cf [15], [60], [61], [62]) and circuit switch multi-traffic models with finite trunk groups and alternate routing under the assumption of "call packing" (cf [7]).



Non-reversible Routing

(cf [27], [28], [47], [52])

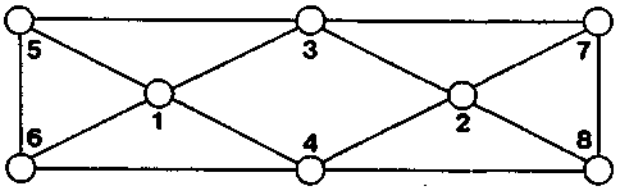
To the best of the author's knowledge a first investigation of product form results for queueing networks with blocking and non-reversible routing was performed in [27] by studying a cyclic three station model. As this model gives some essential insight also for extensions to limited clusters, it will be discussed in detail in example 3.9. A more general and abstract characterization of product form results for queueing networks with blocking and non-reversible routing was given in [28]. The results in section 3 are somewhat related to this reference, but, as will be discussed in more detail in remark 3.2, only concrete examples with limited stations rather than clusters were concluded and the restricted condition of job-local-balance was imposed which excludes for example stations with FCFS-servicing.

A special non-reversible example with blocking and allowing FCFS-disciplines can be found in [54]. In [46] an elegant framework is presented in which nodes (stations) are partitioned in groups which allow non-reversible structures under a partial balance condition, as will be dealt with under the name of "weak reversible" in section 5.1, so as to conclude product forms allowing blocking of various types. For the case of finite solutions or clusters, however, a more detailed condition of "local reversibility" is to be added, as will be outlined in more detail in remark 5.1.1.

2.2 Communications Blocking

(cf [9], [12], [13], [38], [49], [56])

Somewhat similar to the reversible circuit switch application already mentioned above, product form results could also be concluded for communication or interconnection networks such as involving CSMA, BTMA or Rude-CSMA random access protocols. Essentially, see [49], such structures come down to a two-station and thus reversible network but with each job representing a possible different source. The Kolmogorov-criterion or reversibility condition (cf. [33]) is thus to be verified for the blocking protocol under consideration in order to conclude a product form result.

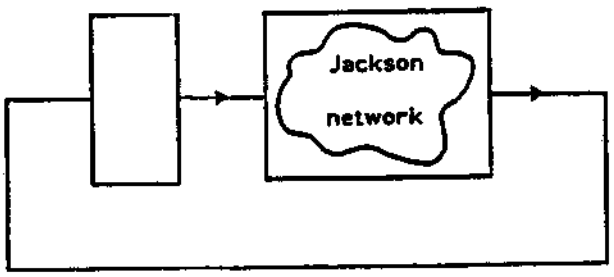


In [12] and [13] this has led to a necessary and sufficient nonrandomized blocking condition for product form multihop packet radio networks, such as satisfied by the CSMA and BTMA protocol (also see [9]), in [38] to a rather unconventional product form result for the so called Rude-CSMA protocol, and in [49], [56] to an invariance condition for general random access protocols to exhibit a product form.

2.3 Limited Clusters

(cf. [11], [31], [35], [36], [46], [48])

In fact, the first product form result for a queueing network with a finite capacity constraint on a group of stations, is already contained in Jackson's classical paper [31]. Herein, he shows his celebrated product form to remain valid when the total number of jobs must remain between certain levels and, to this end, when using a so-called lost-triggering protocol. This essential but not frequently referenced side result of Jackson's paper was extended by Lam [36] to multiclass queueing networks with class interdependent blocking for accessing or leaving the network.



To provide some insight reconsider the two-station example of section 2.1 in which the second station is replaced by a total Jackson network with capacity constraint N_2 on the total number of jobs n in this network. The indicator with n_2 replaced by n would then work in a similar manner as explained, that is, be either 1 in both sides or 0 in both sides due to blocking and non-admissible situations.

A different type result was derived in [48]. Herein clusters of parallel processors were studied with a single entry and a single departure node per cluster. Allowing capacity constraints on a group of parallel processors within a cluster, a product form was shown to hold. A multiclass extension of the results was given in [35]. In section 5.4 this type of product form blocking structures will be discussed in detail as a special application.

In [46] the product form result for the case of finite stations under the "local reversibility" condition was also shown to be extend ble to limited clusters of stations. This "local reversibility" result will be covered and relaxed in section 5.1 (See remark 5.1.1). Finally, as in the two-stage case, also here extensions to multiple jobs changing at the same time such as due to batch departures or arrivals are possible (cf. [11]).

3 Station Balance and Product Forms

In this section we provide a general basic framework so as to conclude product form results for queueing networks with state dependent routing. This framework is essentially related to [28] but the presentation is more concrete and relaxed to the notion of station rather than the more restricted notion of job-local-balance. (Also see remark 3.2). As such this framework has not yet been reported. It essentially reduces the question of product forms to state dependent local routing or traffic equations and an investigation of their solutions.

In particular, it will lead to a practical tool of simple insights at physical basis from which one can conclude whether or not a product form applies and, if so, of what basic form. In the next sections this framework will be used to extend known product form results for queueing networks with finite stations to a wider class of queueing networks with limited clusters of stations.

3.1 Model

Consider a closed queueing networks with N stations and M jobs. A job at station i requires an exponential service with parameter $\mu_i, i = 1, \ldots, N$.

Let the vector $\bar{n} = (n_1, n_2, \ldots, n_N)$ denote the number of jobs n_i at station $i, i = 1, \ldots, N$, thus with $n_1 + \ldots + n_N = M$ and let e_i be the i-th unit vector in $\mathcal{R}^N$. For convenience, we also use the notation $\bar{n} = \bar{m} + e_j$, thus with the

components of $\bar{m}$ summing to $M-1$.

Routing When the system is in state $\bar{n}$ and a job at station i completes its service it will route to station j, with the state dependent probability

$$p_{ij}(\bar{n})$$

In particular $j=i$ is hereby included most notably so as to model blocking.

Servicing When the system is in state $\bar{n}$ the service rate at station i is given by

$$\mu_i(\bar{n}) = \mu_i \frac{F(\bar{n}-e_i)}{G(\bar{n})}$$

where F and G are nonnegative functions which are zero only when their argument is the zero vector $\bar{0}=(0,0,\ldots,0)$ or a vector with a negative component.

Remark 2.1 (Service form) The above service form has been used also in [11] and [47], while in contrast in earlier references, most notably [33], [55] and [56], the much more restrictive form with $F=G$ was considered. An extensive elegant illustration of employing the flexibility of choosing different functions F and G is given in [47].

3.2 Station Balance

Assume that there exists a unique stationary distribution $\pi(\cdot)$ at some set S of admissible states. This distribution is then uniquely determined, up to normalization, by the

Global balance equations:

$$\pi(\bar{n})\sum_j\sum_i \mu_j(\bar{n})p_{ji}(\bar{n}) = \sum_i\sum_j \mu_i(\bar{n})p_{ij}(\bar{n}) \tag{3.1}$$

These in turn are satisfied by the more detailed

Station balance equations:

$$\boxed{\begin{array}{c}\pi(\bar{m}+e_j)\sum_i \mu_j(\bar{m}+e_j)p_{ji}(\bar{m}+e_j) \\ = \\ \sum_i \pi(\bar{m}+e_i)\mu_i(\bar{m}+e_i)p_{ij}(\bar{m}+e_i)\end{array}} \tag{3.2}$$

where for fixed $\bar{n}$ we substitute $\bar{n} = \bar{m} + e_j$ for all j. The following theorem shows that (3.2) has a general solution structure when reduced to purely the set of state dependent routing equations.

Theorem 2.1 Suppose that there exists a function H such that

$$\boxed{H(\bar{m}+e_j)\sum_i p_{ji}(\bar{m}+e_j) = \sum_i p_{ij}(\bar{m}+e_j)H(\bar{m}+e_i)} \tag{3.3}$$

for all $\bar{m}$ and j with $\bar{m} + e_j \epsilon S$, then with c a normalizing constant

$$\boxed{\pi(\bar{n}) = cH(\bar{n})G(\bar{n})\prod_{i=1}^{N}\left(\frac{1}{\mu_i}\right)^{n_i}} \tag{3.4}$$

Proof By substituting the service form and (3.4), equation (3.2) directly reduces to (3.3)

Remark 3.1 (Literature) The above decomposition result in a service and routing part has been detected and mentioned explicitly by several authors (cf. [26], [28], [46], [59]). In fact, one might already argue that also Jackson's classical product form reveals this phenomenon. In particular, it shows that the notion of station balance seems to be responsible for "product form type" results in the more traditional perception of factorizing to individual stations in terms $(1/\mu_i)^{n_i}$.

Essentially, however, theorem 3.1 has only replaced the difficulty of finding a solution for (3.2) in finding a solution for a similar set of equations (3.4) which excludes the service part. The main feature of state dependent routing and most notably blocking is herein still involved.

To investigate when and when not a solution of (3.4) and thus (3.2) can exist, what form it can have and how it can be computed, below we will first investigate (3.4) at a local level and define an auxiliary artificial Markov chain, called $\bar{p}$-model.

$\bar{p}$-model

Consider (3.4) for a fixed $\bar{m}$. Without restriction of generality (also see remark 3.7 below) assume that the local equation (3.3) for this fixed $\bar{m}$ has a unique probability solution $\{y_j(\bar{m})\}$. Hence, for all j

$$\boxed{y_j(\bar{m})\sum_i p_{ji}(\bar{m}+e_j) = \sum_i y_i(\bar{m})p_{ij}(\bar{m}+e_i)} \tag{3.5}$$

Now let $S(\bar{m}) = \{\bar{m}+e_j | \bar{m}+e_j \epsilon S\}$ and define a Markov chain at S with transition probabilities $\bar{p}(\bar{m}+e_j \rightarrow \bar{m}+e_j)$ for $\bar{m}+e_i, \bar{m}+e_j \epsilon S(\bar{m})$ given by

$$\frac{\bar{p}(\bar{m}+e_j \rightarrow \bar{m}+e_i)}{\bar{p}(\bar{m}+e_i \rightarrow \bar{m}+e_j)} = \frac{y_i(\bar{m})}{y_j(\bar{m})} \tag{3.6}$$

while transition probabilities $\bar{p}(\cdot \rightarrow \cdot)$ of any other form are assumed to be 0. Here it is noted that these transition probabilities are unique up to a constant

factor at $S(\bar{m})$ for any fixed $\bar{m}$. Further, note that by varying $\bar{m}$ the state space of this Markov chain coincides with that of the original queueing network: S.

Consider this (or in fact *a*) $\bar{p}$-model and denote its stationary distribution by $H(\cdot)$. The next theorem will then relate the station balance (3.2) or traffic equations (3.4) with a property of the $\bar{p}$-model. To this end, we recall (cf [33]) that a Markov chain with transition probabilities q_{ij} from a state i to j is called reversible if its stationary distribution π_i satisfies for all i, j.

$$\pi_i q_{ij} = \pi_j q_{ji}$$

Theorem 3.2 The traffic equations (3.4) have a solution $H(\cdot)$ if and only if the $\bar{p}$-model is reversible. That is, for all $\bar{m} + e_i, \bar{m} + e_j \epsilon S$:

$$\boxed{\frac{H(\bar{m} + e_i)}{H(\bar{m} + e_j)} = \frac{y_i(\bar{m})}{y_j(\bar{m})}} \tag{3.7}$$

or equivalently, for arbitrary reference state $\bar{n}_0$, any $\bar{n} \epsilon S$ and $c = H(\bar{n}_0)$:

$$\boxed{H(\bar{n}) = c \prod_{k=1}^{z} \frac{y_{j_k}(\bar{m}_k)}{y_{i_k}(\bar{m}_k)}} \tag{3.8}$$

for all possible trajectories:

$$\begin{aligned} \bar{n}_0 &= \bar{m}_1 + i_1 \rightarrow \bar{m}_1 + j_1 = \bar{m}_2 + i_2 \rightarrow \bar{m}_2 + j_2 = \ldots = \\ &= \bar{m}_k + i_k \rightarrow \bar{m}_k + j_k = \bar{m}_{k+1} + i_{k+1} \rightarrow \bar{m}_{k+1} + j_{k+1} = \ldots = \bar{m}_z + j_z = \bar{n} \end{aligned} \tag{3.9}$$

for which all denominators in (3.8) are positive and where z is arbitrary.

Proof Relation (3.7) directly follows from the definition of reversibility while the relation (3.8) is a direct consequence of the Kolmogorov-criterion for reversibility (e.g. [33] p21) and the definition (3.6).

Assume that the $\bar{p}$-model is reversible and thus that (3.7) holds for any pair $\bar{m} + e_i, \bar{m} + e_j \epsilon S(\bar{m})$. By dividing the left and right hand side from (3.3) by $H(\bar{m} + e_j)$, substituting (3.7) and applying (3.5), we then conclude that $H(\cdot)$ satisfies (3.4).

Conversely, let $H(\cdot)$ be a function satisfying (3.3). Then by choosing $\{y_j(\bar{m})\}$ as according to (3.7) we obtain a solution for (3.5) so that by (3.6):

$$\frac{\bar{p}(\bar{m} + e_j \rightarrow \bar{m} + e_i)}{\bar{p}(\bar{m} + e_i \rightarrow \bar{m} + e_j)} = \frac{y_i(\bar{m})}{y_j(\bar{m})} = \frac{H(\bar{m} + e_i)}{H(\bar{m} + e_j)} \tag{3.10}$$

This proves that the $\bar{p}$-model is reversible with steady state distribution $H(\cdot)$.

The practical advantage of the above characterization of station balance is that it shows when and of which form product form results can be concluded and

how these can be computed based upon merely the local solutions of the local traffic equations (3.5). These local solutions are usually much easier to obtain. Theorems 3.1 and 3.2 are thus summarized by:

Conclusion 3.3 To conclude the stationary distribution (3.4) we need to:

1. Determine local solutions $\{y_j(\bar{n})\}$ of equations (3.5).
2. Find a function $H(\cdot)$ that satisfies (3.7) or (3.8).

3.3 Discussion of Results

Remark 3.2 (Literature: [28]) The characterization of product form results by reversibility of an artificially defined associated Markov chain has been introduced in [28]. This reference investigated the more restrictive notion of job-local-balance, requiring balance per individual job, so as to obtain insensitivity results (cf. [29]). As a consequence, a much more abstract framework and more detailed notation was needed. More importantly, for the present application the job-local-balance notion is too restrictive. For example, a result such as (3.4) could not be concluded for queueing networks with FCFS-stations. The above theorem 3.2 therefore has brought the results from [28] down to a wider and simpler form. This form has not yet been reported or recognized in the literature. In particular, as per the next section, it is more appropriate to further investigate the possibilities for finite clusters rather than individual stations.

Remark 3.4 (Non-reversible routing). It is emphasized that theorem 3.2 allows the original queueing network model to be non-reversible. Particularly, as will be illustrated in the next section, the $\bar{p}$-reversibility condition and thus expression (3.4) can hold while both within and between clusters the routing is non-reversible.

Remark 3.6 (Checking (3.7) or (3.8)). As will be illustrated in section 5, the local solutions $y_j(\bar{n})$ can usually be obtained explicitly. These local solutions, in turn, often directly suggest a required form of $H(\cdot)$ in order to satisfy (3.7). As a consequence, rather than checking (3.8) for all possible trajectories, instead we can simply check whether expression (3.1) with this suggested form of $H(\cdot)$ indeed satisfies (3.3). All the examples in the next section have so been verified directly.

Particularly, when only strict blocking such as due to finite capacities is involved, the validity of (3.5) may be concludable directly on a purely down-to-earth physical basis with $H(\cdot)$ of standard form. This will be explained more detailed in remark 3.8 below.

Remark 3.7 (Irreducibility of local chains). The assumption of a unique solution, up to normalization, of the local traffic equations (3.5) for any fixed

$\bar{m}$ is equivalent to requiring that the local Markov chain at $S(\bar{m})$ is irreducible. However, from (3.6), (3.7) and the proof of theorem 3.2 it is readily concluded that only ratios of the local solutions $y_j(\bar{m})$ need to be unique for states $\bar{m} + e_i$ and $\bar{m} + e_j$ that communicate at $S(\bar{m})$. The results, therefore, remain valid if this local irreducibility assumption is relaxed to local chains which decompose in irreducible (or ergodic) sets. In other words that is, the local chain at $S(\bar{m})$ for the original model may not contain transient states. Such a relaxation can be useful for modeling blocked situations (cf [28]).

Remark 3.8 (Simple physical insight) The station balance principle, that is (3.2) or (3.3) verbally reads:

(3.11)

The rate out of state due to a departure at any particular station j
=
The rate into that state due to an arrival at that particular station j

When strict blocking is involved, such as due to finite capacities, it is often easily seen, as will be illustrated below, whether or not this station balance principle can apply or will be violated as it requires in any state, and thus particularly in states with saturations, and for any station j:

(3.12) $$\boxed{\text{Rate out of } j = 0 \Longleftrightarrow \text{Rate into } j = 0}$$

where in addition one has to take into account

(3.13) $$\boxed{S\text{: the set of admissible states}}$$

Roughly, as a general solution, suppose that $p_{ij}(\bar{n}) = p_{ij}$ as long as blocking does not occur, while upon blocking the job has to remain at station i, and that (3.12) is never violated, then

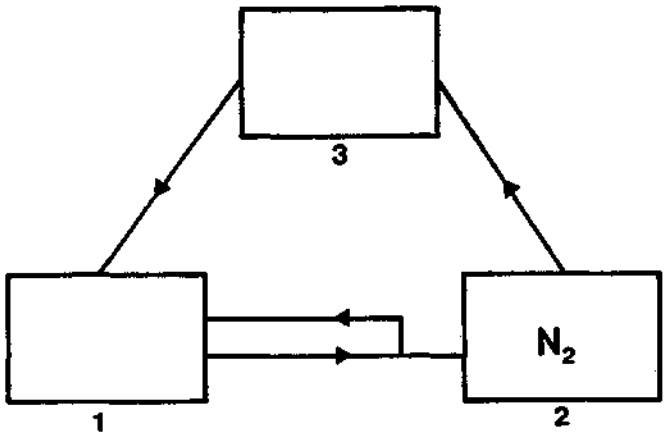
(3.14)
$$H(\bar{n}) = \prod_{i=1}^{N} (\lambda_i)^{n_i}$$
$$\lambda_j = \sum_i \lambda_i p_{ij}$$

3.4 An Instructive Example:

Example 3.9

Consider a network with 3 single server stations ($N = 3$). Upon completion at station i a job attempts to route to station $i + 1$, where $i + 1 = 1$ for $i = 3$. (i.e. we assume cyclic routing). When it is blocked, it has to repeat a service at station i.

1. Suppose that station 2 has a finite capacity constraint for no more than N_2 jobs and consider a state (n_1, n_2, n_3) with $n_1 > 0$ and $n_2 = N_2$. For example, with $N_2 = 10$ and $M = 15$ consider the state (3, 10, 2). Then

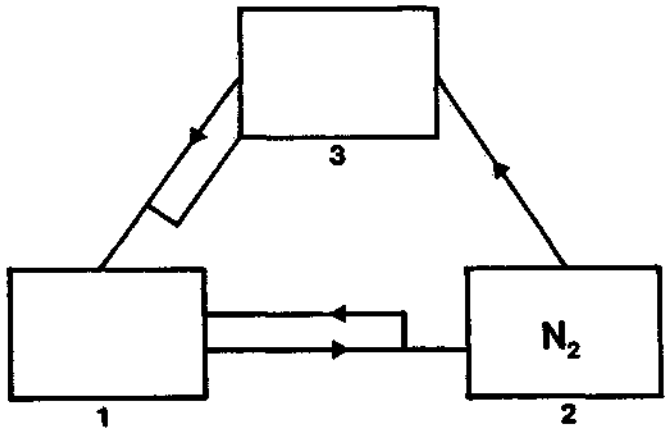


Rate out of state (3,10,2) due to a departure at 1 (i.e. into (2,11,2)) = 0

Rate into this state (3,10,2) due to a arrival at 1 (i.e. from (1,10,4)) > 0

As a consequence, (3.12) and thus also (3.3) and (3.2) necessarily fail for $j = 1$, so that an expression of the form (3.4), recalling the 'if and only if' in theorem 3.2, cannot hold, not for any form of H. (Indeed this is easily checked either by symbolic computation or by numerical computation and taking ratios in different states).

2. Reconsider the example of 1 but now assume that when station 2 is saturated, $(n_2 = N_2)$, not only the servicing at station 1 but also at station 3 is to be stopped. Then, again with $N_2 = 10$ and $M = 15$, we now have:



Rate out of state (3,10,2) due to a departure at 1 (i.e. into (1,11,3)) = 0

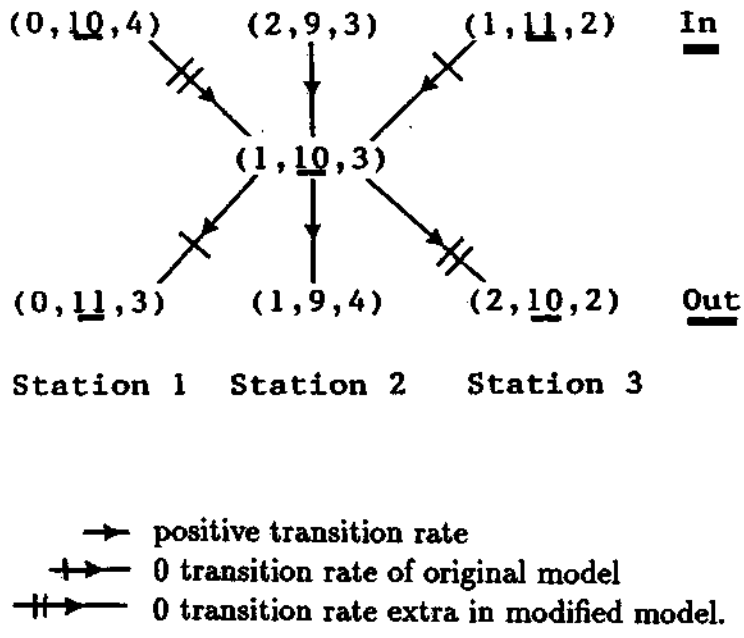
Rate into state (3,10,2) due to a arrival at 1 (i.e. from (1,10,4)) = 0

as the state (3,11,1) is not admissible. Indeed, station balance thus seems to be repaired and by substitution one directly verifies (3.3) with $H(\cdot) = 1$, or equivalently, (3.2) with

$$(3.15) \qquad \pi(\bar{n}) = c \prod_{i=1}^{3} (1/\mu_i)^{n_i} \qquad (n_2 \leq N_2, n_1 + n_2 + n_3 = M)$$

Remark 3.10 (Station balance repaired?)

At second glance one may think that the modification under 2 which intuitively seems to repair station balance at station 1, simply replaces the problem to station 3. Here however again one has to take into account the set of admissible states. As a state with $n_2 = N_2 + 1$ jobs is not admissible, not only the outrate but also the in ate at station 3 will then be 0. This is illustrated by the station balance relations below.



→ positive transition rate
+→ 0 transition rate of original model
+|→ 0 transition rate extra in modified model.

This appears to hold true more generally. That is by repairing 0 inconsistencies for station balance at one station, the station balance at other stations will not be affected due to the implicit structure of non-admissible states caused by blocking.

Remark 3.11 (Application)

The above example directly extends to cyclic multi-station assembly lines of any length where each station i has a capacity constraint N_i. This has possible applications to manufacturing either as a direct realistic blocking protocol so as to avoid congestions at more than one station at a time (cf [15]) or as a mean to obtain simple approximate expressions from which simple bounds such as for a throughput of an original assembly line of interest (like under) are concluded. (cf [51], [57])

3.5 Special case: Reversible routing

One general class of queueing networks with finite stations which is well-known to exhibit the product form (3.4) is the class of reversible networks. More precisely, assume that station i has a capacity constraint for no more than N_i jobs and that a job has to repeat a service at station i when upon service completion it wants to route to a saturated station j. The routing probabilities are p_{ij}. Assuming that there exists values $\{\lambda_i\}$ such that for all i, j:

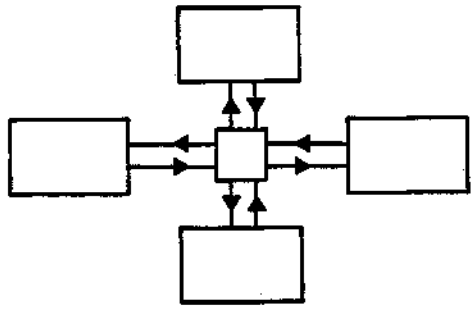
$$\lambda_i p_{ij} = \lambda_j p_{ji}, \tag{3.16}$$

equation (3.5) is directly verified with $y_j(\bar{m}) = y_j$ when $\bar{m} + e_j$ is admissible and 0 otherwise, and thus also (3.3) with $H(\bar{n})$ as according to (3.12) for all $\bar{n} \epsilon S = \{\bar{n} | n_i \leq N_i, i = 1, \ldots, N\}$ and 0 otherwise, in the detailed manner:

$$H(\bar{m} + e_j) p_{ji} = H(\bar{m} + e_i) p_{ij}. \tag{3.17}$$

The form (3.4) thus applies with H given by (3.12). For service rates of the form $\mu_i(n_i)$ this was found in [27], [33], [42]. For service rates with $F = G$ by [59] (Also see [58]), while for the general $F \neq G$ it has first been reported, to the best of my knowledge, by [46].

Note that the routing of example 3.9 is not reversible (i.e. fails (3.16)).



4 Limited Clusters

Unfortunately, despite example 3.9, remark 3.11 and the special reversible case of section 3.5, the class of queueing networks with finite individual stations that satisfy station balance and thus exhibit a product form is quite restricted for practical purposes.

However, by relaxing finite capacity constraints to clusters of stations, the range of possible structures that preserve station balance seems to become much wider. As such constraints are most realistic in practice, such as in manufacturing (e.g. one central buffer for a group of workstations), computer networks (e.g. store and forward buffers for a number of parallel processors) and communications systems (e.g. a common finite trunkgroup for a multi-source circuit switch fabric), this phenomenon will be investigated in more detail below.

4.1 Model

As we need to make the dependence on cluster configurations explicit, some further notation is given. Let the stations $1,\ldots,N$ be partitioned in clusters $C_1,\ldots,C_K$ and let $\bar{t}=(t_1,\ldots,t_K)$ denote the total number of jobs n_k in cluster $k, k=1,\ldots,K$, thus with $t_1+\ldots+t_K=M$. Further as confusion cannot arise, we also let e_i be the ith unit vector in $\mathcal{R}$, and for convenience we use the notation $\bar{t}=\bar{s}+e_j$, thus with the components of $\bar{s}$ summing to $M-1$. Throughout, let C_p be the cluster that contains station i and C_q that contains station j. Now assume that

$$(4.1)\qquad \begin{cases} p_{ij}(\bar{n}) &= p_{ij}B_{pq}(\bar{t}) \qquad (j\neq i) \\ p_{ii}(\bar{n}) &= 1-\sum_{j\neq i}p_{ij}(\bar{n}) \end{cases}$$

where $B_{pq}(\bar{t})\epsilon[0,1]$ for any p,q and t. In words that is, a job first requests to route from station i to station j with probability p_{ij}. However, depending on the cluster configuration $\bar{t}$ and the clusters involved that contain i and j, this request will be accepted with probability $B_{pq}(\bar{t})$ and blocked otherwise. When blocked, the job has to repeat a service at station i.

4.2 Cluster Balance

With $\bar{n} = \bar{m} + e_j$ and $\bar{t} = \bar{s} + e_j$, the traffic equation (3.3) now reduces to:

$$\boxed{H(\bar{m} + e_j) \sum_i p_{ji} B_{qp}(\bar{t}) = \sum_i H(\bar{m} + e_i) p_{ij} B_{pq}(\bar{s} + e_i)} \tag{4.2}$$

or more detailed

$$\boxed{\begin{array}{l} H(\bar{m} + e_j) \sum\limits_{p \neq q} \sum\limits_{i \epsilon C_p} p_{ji} B_{qp}(\bar{t}) + H(\bar{m} + e_j) \sum\limits_{i \epsilon C_q} p_{ji} B_{qq}(\bar{t}) \\ = \\ \sum\limits_{p \neq q} \sum\limits_{i \epsilon C_p} H(\bar{m} + e_i) p_{ij} B_{pq}(\bar{s} + e_i) + \sum\limits_{i \epsilon C_q} H(\bar{m} + e_i) p_{ij} B_{qq}(\bar{t}) \end{array}} \tag{4.3}$$

This explicit dependence on the cluster configuration and the decomposition of transitions within the same cluster and to another cluster as per (4.3) will play a crucial role so as to conclude solutions $y_j(\bar{m})$ of a simple form and thus to determine a function $H(\cdot)$ satisfying (3.3).

In particular, by requiring (4.3) for a fixed $\bar{n}$ while varying $j \epsilon C_q$ and assuming that $B_{qq}(\bar{t}) = 0$ when blocking arises, so that the last term in both the left and right hand side cancel, (4.3) would require in any state and for any cluster q:

Cluster balance

> The total rate out of a state due to a departure at that cluster q =
> The total rate into that state due to an arrival at that cluster q

and most notably:

$$\boxed{\text{Rate out of cluster } q = 0 \Longleftrightarrow \text{Rate into cluster } q = 0} \tag{4.4}$$

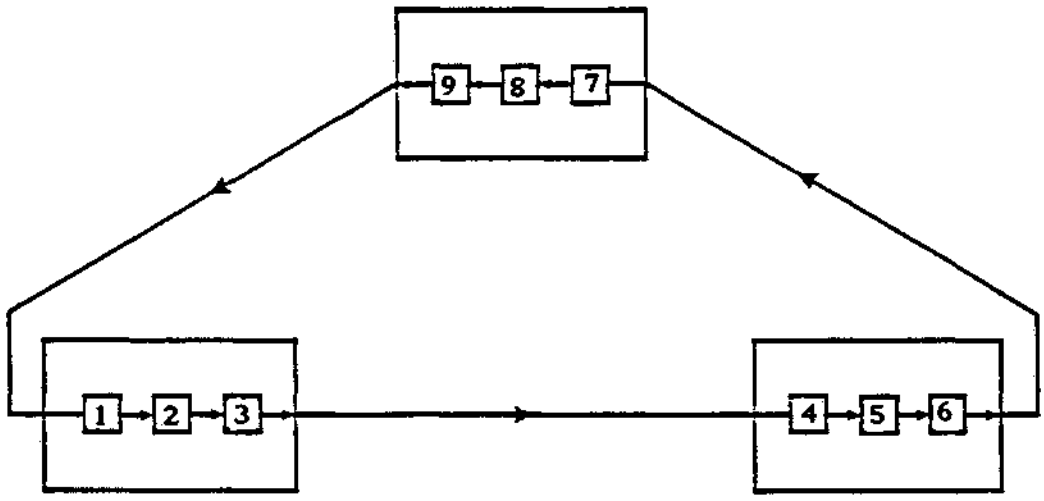
In fact, in order to have cluster balance satisfied we can simply adopt the characterization for station balance as per section 3 by regarding a cluster as one

$$\boxed{\text{"aggregated station"}}$$

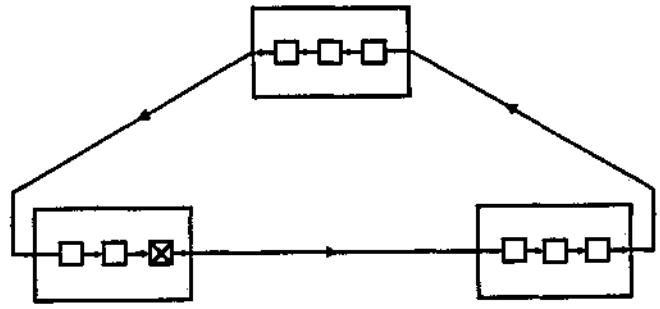
To illustrate this simple physical insight of cluster balance in combination with station balance when cluster blocking is involved, let us extend example 3.9 to clusters.

4.3 An Illustrative Example: Example 4.1

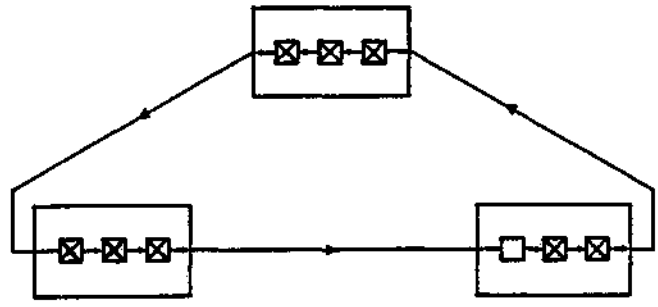
Consider a network with 9 single server stations and a cyclic routing from station i to station $i+1$, where $i+1=1$ for $i=9$. The stations are partitioned in 3 clusters of stations $C_1 = \{1,2,3\}, C_2 = \{4,5,6\}$ and $C_3 = \{7,8,9\}$. Only finite capacities will be mentioned in the various cases below. The + or − in the headings indicate that the product form (3.4) will apply or not.



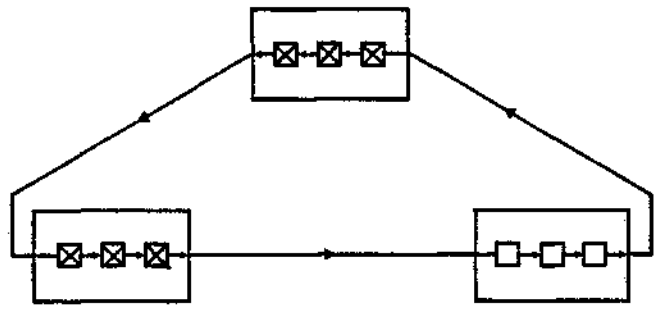
1. (Finite station; no delay; −) If a single station has a finite capacity constraint, say N_4 for station 4, while only the last preceding station, station 3, is stopped when this station is saturated, one directly argues similar to example 3.9.1 that station balance and thus (3.4) cannot hold.



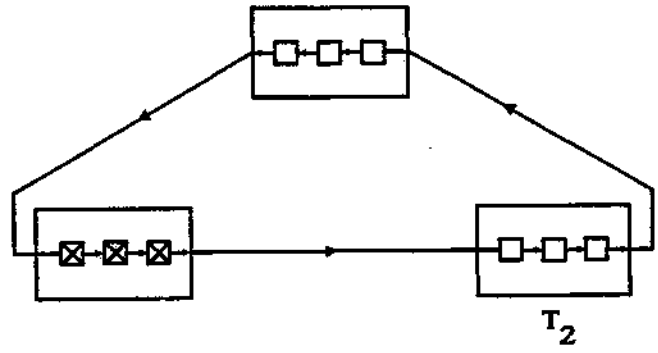
2. (Finite station; stop all other stations; +) As under 1 but when $n_4 = N_4$ servicing at all other stations, in particular also stations 5 and 6 is to be stopped. Then, as under example 3.9.2 one directly argues that station balance and thus (3.4) applies.



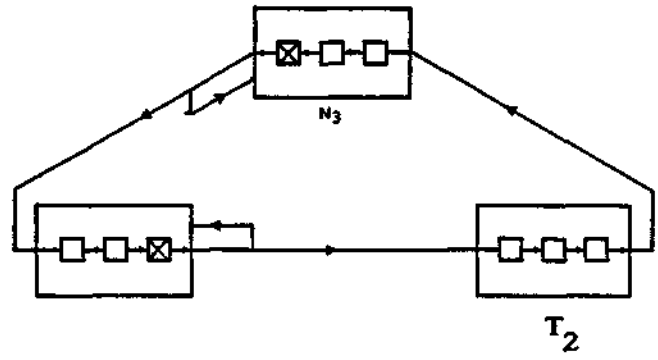
3. (Finite station; only stop all stations at other clusters; −) As under 2 except that stations 5 and 6, as they pertain to the same cluster as 4, continue to service also if $n_4 = N_4$. However, assume $N_4 = 10$, then in a state of the form $(n_1, n_2, n_3, 10, 2, \ldots)$ station balance fails at station 5 (positive outrate) as the state $(n_1, n_2, n_3, 11, 1, \ldots)$ is non admissible (0 inrate) (3.4) thus fails again if we wish to allow other stations within the same cluster to continue



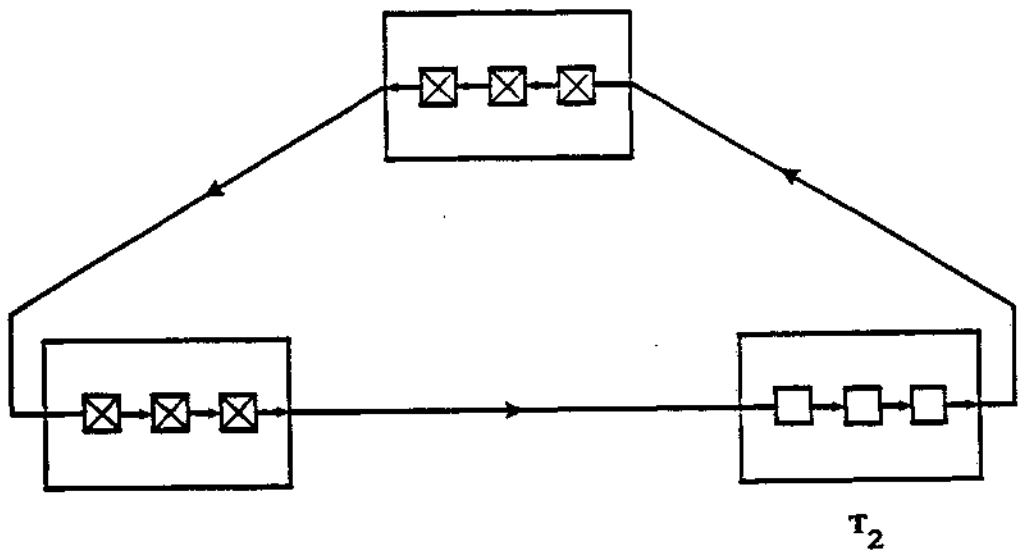
4. (Finite cluster; stop only one cluster; −) Now only assume a finite capacity constraint T_2 on the total number of jobs $t_2 = n_4 + n_5 + n_6$ at cluster 2. Then by stopping only station 3 at cluster 1 when $t_2 = T_2$, one would violate station balance at station 3 as jobs can still enter this station. To guarantee station balance we thus also have to stop station 2 as well as station 1, and thus cluster 1 throughout. But then cluster balance (4.4) would still be violated at cluster 1 similarly as in the station balance violation in example 3.9.1, as it would still allow jobs to enter cluster 1, while jobs cannot leave cluster 1.



5. (Finite cluster; stop other cluster departures only; −) As under 4 but alternatively, when $t_2 = T_2$, we block departures from clusters 1 and 3, so that similarly to example 3.9.2 the cluster balance (4.4) is satisfied. However, if we allow station 2 to continue also if $t_2 = T_2$, station balance would still be violated at station 3.



6. (Finite cluster; stop all other clusters; +) As under 4 and 5. That is, we combine both ideas of stopping all stations in a cluster if the departure station is blocked so as to meet (3.12) and of stopping cluster 3 departures if cluster 1 departures are blocked so as to meet (4.4). In other words, when $t_2 = T_2$ stop all stations at both clusters 1 and 2. Most notably, however, in contrast with example 3, all stations at cluster 2 can always continue to work. Intuitively, the station balance equations now seem to be satisfied.



More precisely, let

$$\text{(4.5)} \qquad \begin{cases} H(\bar{n}) = 1_{\{\bar{n} \epsilon S\}} \\ \\ S = \{\bar{n} | t_2 \leq T_2\} \end{cases}$$

and note that by virtue of the proposed protocol;

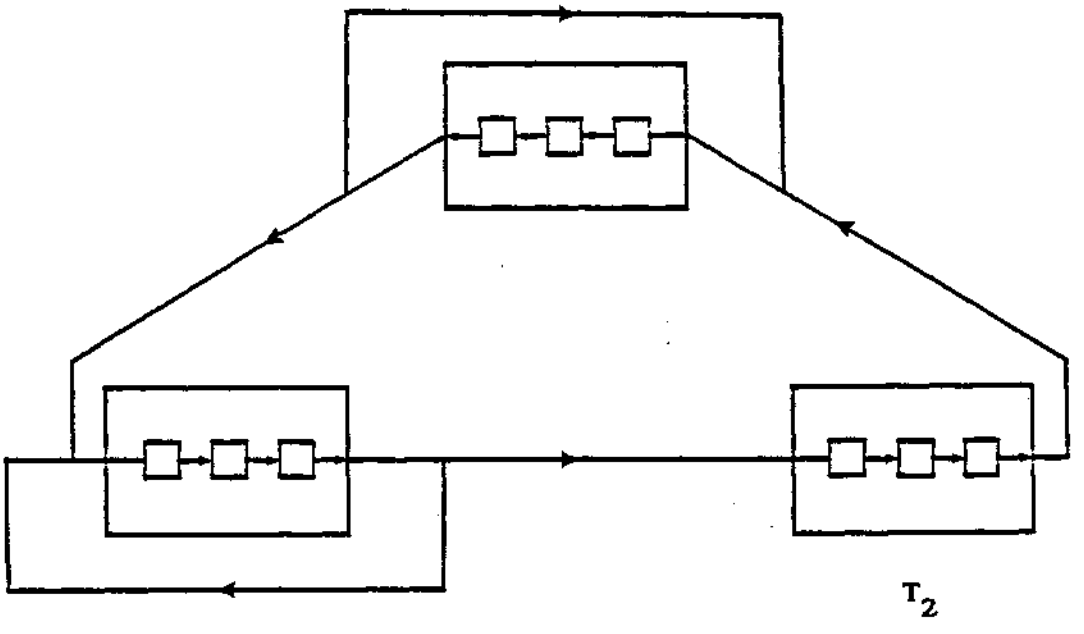
$$\text{(4.6)} \qquad \begin{cases} B_{qp}(\bar{t}) = B_{qq}(\bar{t}) = 1 \qquad (q = 2; p = 3) \\ B_{qp}(\bar{t}) = B_{qq}(\bar{t}) = 1_{\{t_2 < T_2\}} \qquad (q = 1,3)\ (p = 1,3) \end{cases}$$

Hence

For $j \epsilon C_2$ both the left and right hand side of (3.3) are equal to 1
For $j \notin C_2$ both the left and right hand side of (3.3) are equal to $1_{\{t_2 < T_2\}}$.

Expression (3.4) thus applies with $H(\cdot) = 1$ and $G(\cdot) = 1$ (single servers) while upon blocking of cluster 2 the stations within cluster 2 continue to work. Such a result has not been reported in the literature.

7. (Finite cluster; recirculate only departures at other clusters; +) As under 5, in order to satisfy (4.4) and as per the philosophy of example 3.9.2 if we regard a cluster as an aggregate station, we need to prohibit departures at both clusters 1 and 3 if cluster 2 is saturted. In contrast with 6 however, this can also be achieved by recirculating a departing job to the entry station of that cluster, that is from station 3 back to station 1 at cluster 1 and from station 9 back to station 7 at cluster 3, when $t_2 = T_2$. The servicing at all stations $1, \ldots, 9$ in this case always continues.



Intuitive arguments

As already argued, the cluster balance (4.4) now seems to be guaranteed. But in contrast the station balance (3.12) within a cluster, say cluster 1, at first glance has now become less obvious. However, by letting jobs that are blocked to leave a cluster recirculate as newly arriving jobs at that cluster, we more or less make that cluster independent of the rest of the total system as if it were an isolated cluster itself. Assuming that this cluster itself in isolation would satisfy station balance it thus seems preserved under recirculation . In section 6 we provide general formal support for this statement. Below we merely wish to illustrate it for the present example.

Exact verification

In fact, under this protocol, (4.1) is replaced by

$$(4.7) \qquad \begin{cases} p_{31}(\bar{n}) = p_{97}(\bar{m}) = 1_{\{t_2=T_2\}} \\ p_{i(i+1)}(\bar{n}) = 1 \qquad\qquad \text{otherwise} \end{cases}$$

so that (3.3) reduces to:

$$(4.8) \qquad \begin{aligned} H(\bar{m}+e_j) &= H(\bar{m}+e_{j+1}) && (j \neq 1,7) \\ H(\bar{m}+e_j) &= H(\bar{m}+e_{j+1}) && (t_2 < T_2) \quad (j = 1,7) \\ H(\bar{m}+e_j) &= H(\bar{m}+e_{j+2}) && (t_2 = T_2) \quad (j = 1,7) \end{aligned}$$

These equations are trivially satisfied by substituting $H(\bar{n}) = 1$ for all $\bar{n} \epsilon S = \{\bar{n} | t_2 \leq T_2\}$ and 0 otherwise. (One may note though that a positive outrate of station 7 is now balanced by a positive inrate from station 6 as long as $t_2 < T_2$ but by a positive inrate from station 9 if $t_2 = T_2$ as a state with $t_2 = T_2 + 1$ is non-admissible). As before, expression (3.4) thus applies with $H(\cdot) = 1$, in this case with stations also within clusters 1 and 3 always continuing to work. Again, such a result has not been reported in the literature.

4.4 Discussion and Summary

Remark 4.1 (Equivalence) The equivalence of the steady state distribution under the "stop protocol" as under 6 and the "recirculate protocol" as under 7 is of interest in itself and seems to be intrinsically connected to product form results. (Note that this equivalence of both protocols is certainly not true in general). In section 6 this will be investigated and generalized in more detail.

Remark 4.2 (Norton's theorem) For product form networks without blocking Norton's theorem is known to be applicable by which groups of stations can be replaced by one aggregate station. At first glance one might thus suspect this theorem to be applicable to extend product form results for networks with finite stations to networks with limited clusters. However, this is not generally possible as

(i) a microstate description specifying the population size for each station is to be kept.

(ii) also blocking or service delays within clusters need to be carefully taken into account, as illustrated for example by the various cases of example 4.1.

(iii) it is not clear at all a priori what simple service rate of one station is to be chosen to represent the aggregate rate of a whole cluster especially not when blocking, as the primary focus, is involved.

(iv) and even if these aggregate service rates were known, the question of a product form would then be replaced to that of a network with stations having special, sometimes 0, state dependent service rates, for which product form results are more or less limited to only a strict reversible routing, while most applications in the next section will go beyond reversibility.

In fact, the results herein implicitly show the reverse: When and when not Norton's theorem remains applicable when blocking is involved. For example, it does for case 6 but it does not for case 2 of example 4.1.

Summary

The above examples have illustrated the relevance of having to be careful about the particular protocol in order and how the station/cluster balance relation (4.3) can be violated on two levels:

At station balance level within a cluster
At cluster balance level between clusters

By rather simple physical insights as per (3.12) and (4.4), furthermore, a direct negative answer or first intuitive positive answer can frequently be obtained after which (3.2) is to be checked formally, such as by using theorem 3.2. Product form examples with finite clusters, non-reversible routing and continuing service within clusters appears possible.

5 Applications

Objective

This section aims to provide further insight in how, both at a down-to-earth physical level as based on the station and cluster balance principles as a first step and on a more formal level by studying the station balance equation (3.3) or its local version as a second step, further classes of product form networks with limited clusters can be recognized.

The examples that will be given are not exhaustive but will cover various essential structures such as:

- a weak reversible routing (section 5.1)
- a reversible cluster routing and delay (section 5.2)
- non-reversible cluster routing with extra stopping conditions (section 5.3)

with typical practical features like

- single cluster visiting
- parallel workstations
- one input/output station
- two-stage blocking structures
- deterministic (cyclic) routing

Practical Relevance

Though the examples themselves may not directly model a complete realistic application, as these are usually larger with several complicating factors involved at the same time, they do represent typical generic components of more complex realistic systems. By combining these components more complex examples could also have been given. By investigating them separately, however, better insight is provided. Moreover, it provides one a reference framework of basic elements from which one can work onward, such as to develop simple approximations or bounds. For example, the insight of station balance like in example 3.9 has already led to simple secure lower and upper throughput bounds for finite production lines or systems with breakdowns (cf [50], [51], [57]). The extension of example 3.9 to example 4.1 with limited clusters rather than finite individual stations, for example, will directly lead to similar bounds for production lines with finite buffers placed in between groups of workstations, as seems most common in so-called push and pull manufacturing systems.

Literature

With the exception of a subcase of the weakly reversible application 5.1, which has just recently and independently of the present paper been partially addressed in [46], as well as application 5.3.3 which is included to show that other seemingly different results from [35] and [48] are covered, all other examples are essentially new and cannot be concluded from existing literature as will be indicated in more detail as we proceed. Particularly, much attention is paid to different levels of non-reversible structures that can be involved, both within and between clusters.

Blocking Protocol

As the "stop" protocol, also referred to as the "communications" protocol, as in example 4.1.6 under which blocking leads to interruption of services or transmissions, is more realistic than the recirculate protocol, as in example 4.1.7, in this section we will first restrict our presentation to the "stop" protocol. Referring to section 6, though, it is mentioned in advance that for each of these examples the product from result is retained also under a recirculate protocol. Throughout we thus assume the state dependent routing probabilities to be of the form (4.1).

Without mentioning otherwise the function $H(\cdot)$ and values $\{\lambda_j\}$ will always be determined as per (3.14), that is by

(5.1)
$$\boxed{\begin{aligned} H(\bar{n}) &= \prod_{i=1}^{N} (\lambda_i)^{n_i} \\ \lambda_j &= \sum_i \lambda_i p_{ij} \end{aligned}}$$

Further, for convenience we recall here that similarly to (4.2) and (4.3) the local routing equations (3.5) under the assumption (4.1) reduce to:

(5.2)
$$\boxed{\begin{aligned} & y_j(\bar{m}) \sum_{p \neq q} \sum_{i \epsilon C_p} p_{ji} B_{qp}(\bar{t}) + y_j(\bar{m}) \sum_{i \epsilon C_q} p_{ji} B_{qq}(\bar{t}) \\ &= \\ & \sum_{p \neq q} \sum_{i \epsilon C_p} y_i(\bar{m}) p_{ij} B_{pq}(\bar{s} + e_i) + \sum_{i \epsilon C_q} y_i(\bar{m}) p_{ij} B_{qq}(\bar{t}) \end{aligned}}$$

Finally, in these examples only the routing feature will be addressed so as to satisfy (4.3) and thus (3.3), as the major point of interest of this study, that is blocking, is hereby taken into account. The service can be thought of as of any form. For an elegant exposé of the possibilities of this general service form we refer to [46].

5.1 Weak Reversible Routing

The class of examples in this section typically applies to:

- Communication networks in which packets or tokens are sent through finite subnetworks by some returning carrier or bus as in token-ring mechanisms or by "dual bus" systems like DQDB.
- Manufacturing systems with finite clusters of workstations that are interconnected by single entry/departure nodes such as to control each cluster decentralized.
- Computer networks with parallel processors sharing a common restricted resource, such as a memory module or store and forward buffer.

Model As an extension of reversibility now suppose for any cluster with a finite constraint $U_p < \infty$ for no more than U_p jobs, any cluster $q \neq p$, and station $j \epsilon C_q$ that the total routing probability from station j to cluster p is equal to the total reversed routing probability from cluster p to station j. In formula:

$$\lambda_j \sum_{i \epsilon C_p} p_{ji} = \sum_{i \epsilon C_p} \lambda_i p_{ij} \qquad \text{(for all } j \text{ and } p\text{: } U_p < \infty) \tag{5.3}$$

Further, for each cluster p assume that the total number of jobs cannot exceed this upper limit U_p or be less than some lower limit $L_p \leq U_p$. More precisely, a job which attempts to enter cluster p while t_p jobs are already present is blocked with probability $a_p(t_p)$, in which case it is returned to its node where it came from to receive a new service. Similarly, a job which attempts to leave cluster p while t_p jobs, that job included, are present at cluster p is blocked with a probability $d_p(t_p)$, in which case it also has to undergo a new service at the node it came from. Here, it is assumed that

$$\begin{cases} a_p(t_p) > 0 & \text{for } t_p < U_p \\ d_p(t_p) > 0 & \text{for } t_p > L_p \\ a_p(U_p) = 0 \\ d_p(L_p) = 0 \end{cases} \tag{5.4}$$

Note here that servicing at nodes always continues.

Parametrization and Result Under (4.1) this model is parametrized by

$$(5.5) \qquad \begin{cases} B_{pq}(\bar{t}) & = a_q(t_q)d_p(t_p) \quad \text{(for all } q \neq p\text{)} \\ B_{pp}(\bar{t}) & = 1 \qquad\qquad\quad \text{(for all } p\text{)} \end{cases}$$

Using the "weak reversibility" condition (5.3), the regularity conditions (5.4) and

$$(5.6) \qquad S = \{\bar{n} | L_p \leq t_p \leq U_p \quad , p = 1, \ldots, P\},$$

the local routing equations (5.1) are now verified substituting

$$(5.7) \qquad y_i(\bar{m}) = \frac{a_p(t_p)}{d_p(t_p + 1)} \lambda_i$$

Relation (4.3) and thus the product form expression (3.4) is then easily checked with

$$(5.8) \qquad H(\bar{n}) = \left\{ \prod_{i=1}^{N} (\lambda_i)^{n_i} \right\} \left\{ \prod_{p=1}^{P} \prod_{k=L_p}^{t_p - 1} \frac{a_p(k)}{d_p(k+1)} \right\} \quad (\bar{n} \epsilon S)$$

Remark 5.1.1 (Literature [46]) At the time of submission of this paper, independently, a rather closely related characterization was obtained in [46]. More precisely, in this reference the network of nodes is partitioned in nodes (see (14)–(16) of [46]) that are strict reversible in the classical sense of

$$(5.9) \qquad \lambda_j p_{ji} = \lambda_i p_{ij},$$

and for end of these nodes j a so-called "routing balance partition" of disjoint clusters $B_1, \ldots, B_m$ where for each cluster the "weak reversibility", in this reference called "partial balance", condition (4.2) is satisfied, i.e. for each $B = B_1, \ldots, B_m$.

$$(5.10) \qquad \left(\sum_{k \epsilon B} \lambda_j p_{jk} - \lambda_k p_{kj} \right) = 0$$

In this setting, which significantly extends the standard reversible case where (5.9) is required allover, a product form result is concluded provided the state dependent routing probabilities, involving blocking, are so called symmetric. (see theorem 5.1 and definition 4.3 of [46]). New examples with state-dependent acceptance sets or finite stations were hereby concluded.

However, in order to conclude a product form result when a limited capacity constraint is imposed on a cluster of nodes, the more restrictive assumption is imposed (see example 5.5 of [46]: maximum capacity on sets of nodes) of the network being partitioned (see example 3.7) in disjoint clusters $C_1, \ldots, C_K$ where for any $q = 1, \ldots, K$ it is required (called "local reversible") that

$$(5.11) \qquad \lambda_j p_{jk} = \lambda_k p_{kj} \quad \text{(for all } j \epsilon C_q, k \notin C_q\text{)}$$

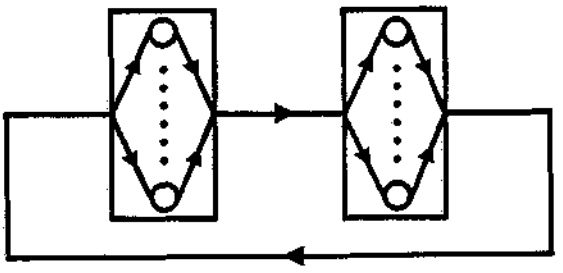
This same condition is also required in [18] and [19].

Remark 5.1.2 (Other literature) Beyond the comparison with reference [46] as per remark 5.1.1, application 5.1.1 does not seem to be available in the literature. More precisely, it extends the more standard product form results for networks with blocking as mentioned in the introduction, in that the routing is allowed to be non-reversible within clusters and "weakly reversible" between clusters. It essentially differs from [31] and [36] in that a blocked departure of a cluster leads to a repetition only at the last visited service station rather than a recirculation throughout the entire cluster. Further, both references basically restrict to one cluster with a Poissonian input. The results from [35] and [48], which will be discussed in detail in section 5.3.3, exclude blocking between clusters.

Remark 5.1.3 (Service continuation). In contrast with section 5.2.1 below, note that here servicing and routing within clusters is continued normally regardless of the probability with which jobs can actually leave the cluster. Only cluster departures are blocked or delayed.

Example 5.1.1 (One station per cluster visit)

A most simple example is the pure reversible case in which (5.1.6) holds for all i, j. For example, consider a structure of two interconnected clusters of parallel processors. When entering cluster p, a job is assigned processor $j \epsilon C_p$ with probability b_j. When completing service at a processor a job will route to the other cluster provided this is possible as according to (5.5).



Essentially, per visit at a cluster a job only uses one processor. The reve ility condition (5.9), in fact (5.11), and thus expression (5.8) now applies wit

$$\lambda_j = b_j \tag{5.12}$$

With $a_p(t) = 1$ for $t < U_p$ and $d_p(t) = 1$ for $t > L_p (p = 1, 2)$, this example can also be seen as a state space truncation of a reversible queueing network and as such be concluded from [33] or [42]. As per remark above, for arbitrary scalings $a_p(\cdot)$ and $d_p(\cdot)$ of the form (5.4), it also follows from [46].

Example 5.1.2 (One input/output station per cluster)

Suppose that each cluster has one particular station at which jobs enter and leave the cluster and assume that the clusters are interconnected in a reversible manner. That is, with R_{pq} the routing probability from cluster p to q assume that for some $\{\gamma_1, \ldots, \gamma_p\}$:

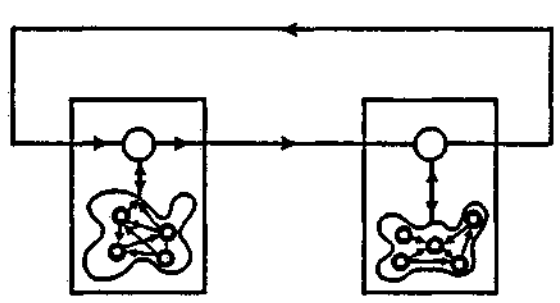
$$\gamma_p R_{pq} = \gamma_q R_{qp} \tag{5.13}$$

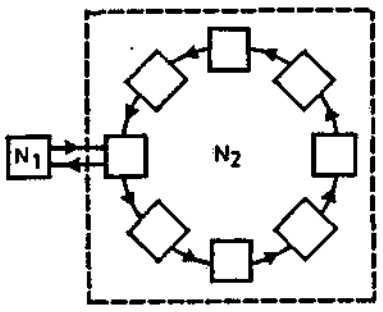
Then with j^* and i^* denoting these special input/output stations of clusters q and p we necessarily have

$$\frac{\lambda_{j^*}}{\lambda_{i^*}} = \frac{\gamma_q}{\gamma_p} = \frac{R_{pq}}{R_{qp}} = \frac{p_{j^*i^*}}{p_{i^*j^*}}, \tag{5.14}$$

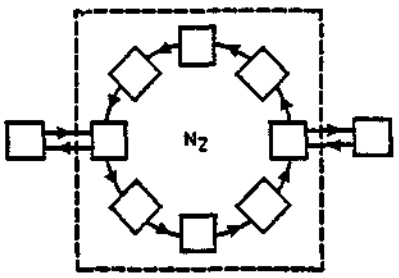
by which (5.10) is readily verified.

This special station for instance can represent a memory module that is to be updated upon entrance and departure. Or, this station can represent the communication channel of a token ring system.





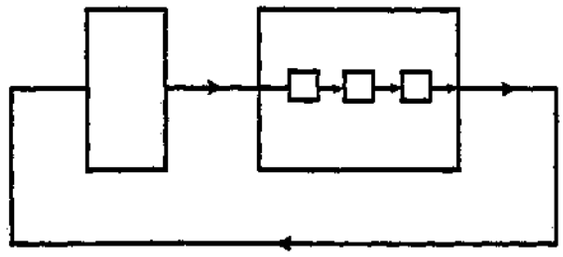
Clearly, the above example directly extends to more entry/departure nodes. For example, it also applies to the following structure somewhat related to the DQDB (dual queue distributed bus) protocol:



Note that in both examples 5.1.1 and 5.1.2 the local reversibility condition (5.11) applies so that these examples could also have been concluded from [46].

Example 5.1.3

An example which satisfies (5.10) but not (5.11) is the following. The total flow out of cluster 2 equals the total flow into cluster 2 but not per station.



Clearly, λ_i as according to (5.1) is given by

$$\lambda_i = 1 \quad (i = 1, \ldots, 4) \tag{5.15}$$

so that the weak reversibility condition (5.10) applies for station $j = 1$ and cluster $q = 2$. Assuming that only cluster 2 has a finite capacity constraint $U_2 < \infty$, while $U_1 = \infty, L_1 = L_2 = 0$, all necessary weak reversibility conditions (5.10) are satisfied so that (5.8) applies.

This simple example can strictly not be concluded from literature directly.

5.2 Reversible Cluster Routing

This section will relax the weak reversibility condition (5.10) per individual station j to merely the reversibility condition (5.13) for the routing from one cluster to another. A most simple but rather general application is that of a two-cluster interconnected network with typical examples like (see section 5.2.1):

- Interconnected metropolitan area's (MAN's) (cf. [43])
- Two interconnected assembly lines (cf [15], [62])

But is also applies for example to (see section 5.2.2):

- Computer network architectures with a group of processors connected, possibly with randomization, to groups of memory modules, disk drives and drums.

In contrast with examples 5.1.1 and 5.1.2, a single input/output station per cluster or a single station visit per cluster visit, is not required. Instead, the price for this relaxation is some form of global or randomized delay of an entire cluster when departures from that cluster will be blocked. For illustration we will consider two separate examples.

Generally, we assume that upon entering a cluster p a job is assigned station $j \epsilon C_p$ with fixed probability b_j and that the routing of a job to another cluster q only depends on the cluster p and not the station $i \epsilon C_p$ that it came from. Hence

$$p_{ij} = \begin{cases} g_i R_{pq} b_j & (i \epsilon C_p, j \epsilon C_q, q \neq p) \\ p_{ij} & (i, j \epsilon C_p) \end{cases} \tag{5.16}$$

where g_i is the probability that a job having completed service at station $i \epsilon C_p$ requests to leave that cluster p. Essentially, routing from one cluster to another is thus independent of the source and destination node within these clusters. Further, the cluster routing itself is assumed to be reversible, i.e. for some $\{\gamma_1, \ldots, \gamma_p\}$:

$$\gamma_p R_{pq} = \gamma_q R_{qp} \tag{5.17}$$

In fact, in this case the solution $\{\lambda_1, \ldots, \lambda_N\}$ is determined by

$$\lambda_j = \gamma_p \tau_j \qquad (j \epsilon C_p) \tag{5.18}$$

5.3 Non-reversible Cluster Routing

In this section, reversibility conditions not only within but also between clusters will be totally left. In turn, the blocking protocol will become more conservative (sections 5.3.1 and 5.3.2) or require special entry buffering (section 5.3.3).

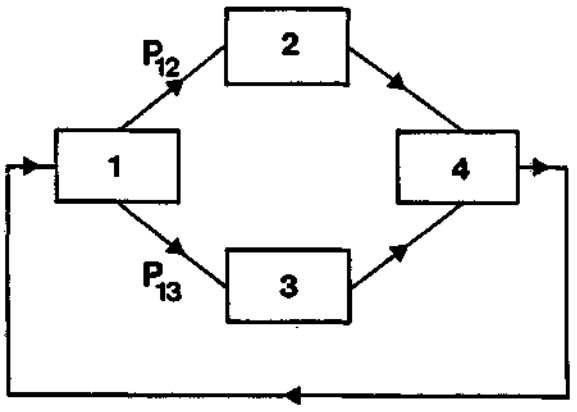
Though somewhat special, such protocols may be practical in certain applications, most notably:

- In computer or manufacturing situations where one saturated part functions as a bottle neck for the whole system (see section 5.3.1).
- Similarly, to avoid that more than one cluster can become saturated at the same time (also section 5.3.1).
- In manufacturing systems in which workload is aimed to be balanced over specific clusters (just-in-time principle) (see section 5.3.2)
- In computer, communication or manufacturing with parallel processors, channels or workstations to which jobs are assigned via a central entry station (see section 5.3.3).

In particular, these protocols become more practical when combination or nesting of the given examples are allowed as in section 5.4.

Examples 5.3.1 and 5.3.2 are new as cluster extensions of examples in [28]. Example 5.3.3 is adopted from [48] but included to illustrate that the different setting of [35] and [48] is covered.

5.3.1 Conservative blocking



As an extension of example 4.9.6, consider an arbitrary network with P clusters and arbitrary routing probabilities p_{ij}.

Cluster C_P pas a capacity limitation of N_p jobs, $p = 1, \ldots, P$. As soon and long as one of the clusters is saturated, i.e. cluster p when $s_p = N_p$ servicing at all stations at any other cluster is to be stopped. For example, in the figure below not only cluster 4 but also clusters 2 and 3 are to be stopped when $t_1 = N_1$.

Remark 5.3.1

Note that this protocol prevents more than one cluster to become saturated at the same time. This can be of practical interest from a reliability point of view so as to avoid deadlocks or catastrophes or to guarantee a certain system efficiency. One could thus call it a conservative protocol.

The set of admissible states is given by

$$S = \{\bar{n} | s_p \leq N_p, p = 1, \ldots, P \quad \text{and} \quad s_p + s_q < N_p + N_q \quad \text{for all} \quad p \neq q\}$$

and the corresponding parametrization becomes

$$\text{(5.35)} \qquad \begin{cases} B_{pq}(\bar{t}) = 1_{\{t_c < N_c \text{ for all } c \neq p\}} \\ B_{pp}(\bar{t}) = 1_{\{t_c < N_c \text{ for all } c \neq p\}} \end{cases}$$

With this parametrization, the local routing equations (5.2) become: for $j \epsilon C_p$ and $\bar{t}$ admissible so that $t_p \leq N_p$:

$$\text{(5.36)} \qquad y_j(\bar{m}) \prod_{q \neq p} 1_{\{t_q < N_q\}} =$$

$$\sum_{q \neq p} 1_{\{t_q < N_q\}} \prod_{s \neq q,p} 1_{\{t_s < N_s\}} \sum_{i \epsilon C_q} y_i(\bar{m}) p_{ij} + \prod_{q \neq p} 1_{\{t_q < N_q\}} \sum_{i \epsilon C_p} y_i(\bar{m}) p_{ij}$$

As a consequence, when $t_q < N_q$ for all $q \neq p$ these equations reduce to the standard traffic equations:

$$\text{(5.37)} \qquad y_j(\bar{m}) = \sum_i y_i(\bar{m}) p_{ij}$$

with solution $y_i(\bar{m}) = \lambda_i$ for all i.

When one cluster is saturated, say cluster v:

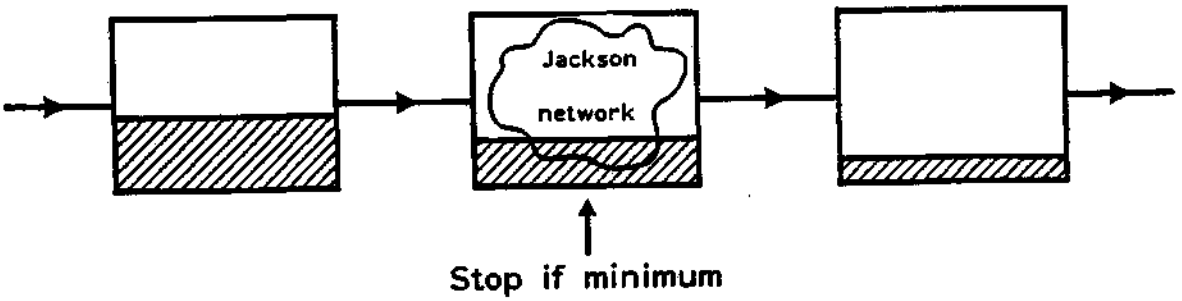
for $j \epsilon C_v$: (5.36) also reduces to (5.37)
for $j \epsilon C_p$ with $p \neq v$: both the left and right hand side of (5.36) are 0.

The standard solution $y_j(\bar{m}) = \lambda_j$ for all j and $\bar{m}$ with $\bar{m} + e_j \epsilon S$ thus applies which yields the product form expression (5.1).

5.3.2 Cyclic cluster routing; minimal workloads

Now assume the routing from one cluster to another to be cyclic, as illustrated below for $p = 3$. Each cluster, however, always requires some minimal number of jobs (workload) to be present in total, say at least M_p jobs at cluster p. (This is typical in western production control such as due to safety stocks or for MRP purposes).

In order to achieve this, the servicing of all stations at a cluster with minimal workload $t_p = M_p$ is stopped, while servicing of other clusters is not affected.



With the corresponding parametrization

$$(5.38) \qquad \begin{cases} S = \{\bar{n} | t_p \geq M_p, p = 1, \ldots, P\} \\ B_{p,p+1}(\bar{t}) = 1_{\{t_p > M_p\}} \quad (p < P) \\ B_{P,1}(\bar{t}) = 1_{\{t_P < M_P\}} \end{cases}$$

the local traffic equations restricted to S become for $j \epsilon C_p$:

$$(5.39) \qquad y_j(\bar{m}) 1_{\{t_p > M_p\}} = \sum_{i \epsilon C_{(p-1)}} y_i(\bar{m}) 1_{\{\bar{m} + e_i \epsilon S\}} p_{ij} + \sum_{i \epsilon C_p} y_i(\bar{m}) 1_{\{t_p > M_p\}} p_{ij}$$

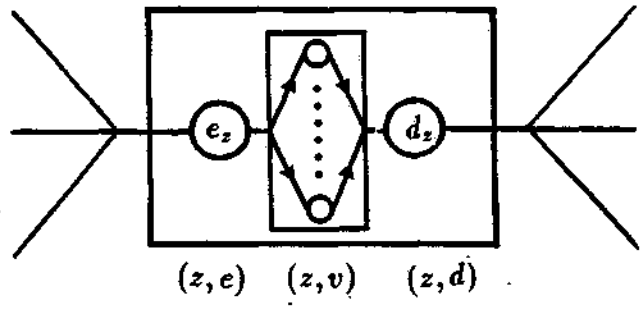
where $(p-1)$ can be read as P for $p = 1$. However, as $\bar{m} + e_j \epsilon S$ we have $\bar{m} + e_i \epsilon S$ for any $i \epsilon C_{(p-1)}$ if and only if: $t_p > M_p$. Consequently, for $t_p > M_p$, (5.39) reduces to (5.37) again, while for $t_p = M_p$ all terms involved are equal to 0. As above, the product form expression (5.1) is hereby concluded.

Remark 5.3.2 (Workload balancing) By also imposing capacity upper limits N_p for all clusters p and employing the conservative stopping protocol of example 5.3.1, we can thus control the workload at cluster p between limits $M_p \leq t_p \leq N_p$ and retain the product form (5.1). In particular, the workload at successive clusters can so be balanced within certain margins (loadbalancing).

5.3.3 One entry and departure station (cf [48])

Though the framework in [48] and its multi-class extension in [35] is somewhat different from the set-up herein, below it will be illustrated that it also fits in the framework of this paper. Essentially, the blocking comes down to that of section 5.1.2 within local subnetworks, while blocking is not involved between subnetworks. To highlight the transformation, the presentation is slightly different from that in section 4 of [48].

Consider a network of Z disjoint so-called subnetworks where each subnetwork, say the z-th, has a number of m_z parallel processors, called branches, labeled $(z,1),\ldots,(z,m_z)$, one entry node e_z labeled (z,e) and one departure node d_z, labeled (z,d),



Jobs route from one subnetwork q to another subnetwork z according to routing probabilities α_{qz}. Consider a fixed subnetwork z. Let n_{zv} denote the number of jobs at the v-th branch (processor) and $n_z = n_z + \ldots + n_{zm_z}$. Then upon service completion at the entry node a job is routed to the v-th branch with probability:

$$\beta_{zv} h_{zv}(n_{zv}) h_z(n_z) \tag{5.40}$$

where β_{zv} are arbitrary probabilities and $h_{zv}(\cdot)$ and $h(\cdot)$ arbitrary nonnegative functions. In contrast with [48], we do not require these probabilities to sum up to one which imposes linearity conditions upon the functions h (see theorem 3 of [48]). Strict blocking is thus allowed in which case the job has to undergo a new service at the entry node.

Let $(\gamma_1,\ldots,\gamma_z)$ be a solution, unique up to normalization of

$$\gamma_z = \sum_q \gamma_q \alpha_{qz} \qquad (z = 1,\ldots,Z) \tag{5.41}$$

We now reformulate the above network in the terminology of section 4 as follows. For each z-th subnetwork consider each processor $v = 1,\ldots,m_z$, the entry node and departure node as a separate cluster where for clarity we identify the corresponding two-dimensional labelling $(z,1),\ldots,(z,m_z),(z,e),(z,d)$ with the one dimensional numbering j for separate stations. Then

$$B_{ze,zv}(\bar{t}) = h_{zv}(t_{zv}) h_z(t_{z1} + \ldots + t_{zm_z})$$

and $B_{\cdot\cdot}(\cdot) = 1$ otherwise, parametrizes the branching and blocking described above. The local routing equations (3.5) now become for $z = 1, \ldots, Z$ and $v = 1, \ldots, m_z$:

$$(5.42) \qquad \begin{cases} y(z,v|\bar{m}) = y(z,e|\bar{m})B_{ze,zv}(\bar{t}) \\ y(z,e|\bar{m}) = \sum_q y(q,d|\bar{m})\alpha_{qz} \\ y(z,d|\bar{m}) = \sum_v y(z,v|\bar{m}). \end{cases}$$

With $n_{zv} = \min\{k|h_{zv}(k) = 0\}$ and $N_z = \min\{k|h_z(k) = 0\}$, and

$$S = \{\bar{n}|n_{pv} \leq N_{pv}, v = 1, \ldots, m_p; n_{p1} + \ldots + n_{pm_p} \leq N_p, n = M\}$$

these local equations restricted to S, the reversibility condition (3.7) and thus also the product form result(3.4) are easily verified by choosing $H(\bar{n}) = 0$ for $\bar{n} \not\in S$ and

$$(5.43) \qquad \begin{aligned} H(\bar{n}) \; = \; & \\ & \prod_{z=1}^{Z} [\gamma_z]^{[n_{pc}+n_{pd}+n_z]} \prod_{k=1}^{n_z} h_z(k-1) \\ & \prod_{v=1}^{m_z} [\beta_{zv}]^{n_{zv}} \prod_{k=1}^{n_{zv}} h_{zv}(k-1) \qquad (\bar{n} \in S). \end{aligned}$$

Remark 4.3 Extensions such as to a common entry node for various subnetwork as in [48] and to multiclass and hierarchical subnetworks structuring as in [35] can also be transformed in a similar manner.

Remark 4.4 At first glance one may be amazed about the fact that no routing condition at all, such as reversibility, is required to route from one subnetwork to another. However, the implicit condition to this end is that blocking is allowed only within subnetworks. There, the special entry parallel processor-departure structure is essentially a reversible one as in section 5.1.2.

5.4 Nested Structure

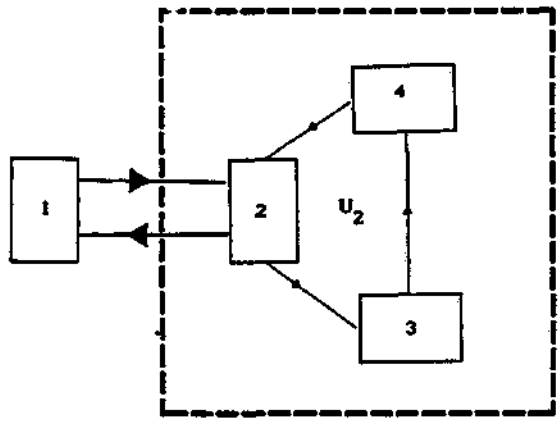
Combinations of the various preceding examples are also possible. In particular, hierarchical or nested blocking structures are obtained. This section merely aims to illustrate this by a few of such examples to give some insight of the nonexhaustive extensions in this direction.

Just as capacity constraints on groups of stations are natural also additional capacity constraints on groups of clusters appear natural in various computer networks, communication structures or manufacturing systems. Nested or hierarchical blocking structures then arise.

All examples below exhibit the product form (3.3) with $H(\cdot)$ as per (5.1) and appear to be new in the literature.

The technical details of verification are just combinations of earlier verifications and therefore left to the reader. Further, as illustrated also by the figures in all examples the routing from one cluster to another is assumed to be independent of the source and destination node within these clusters, i.e. to be of the form (5.16).

(i) As a combination of 5.1.3 and 5.3.1, consider the structure



with finite capacity constraints

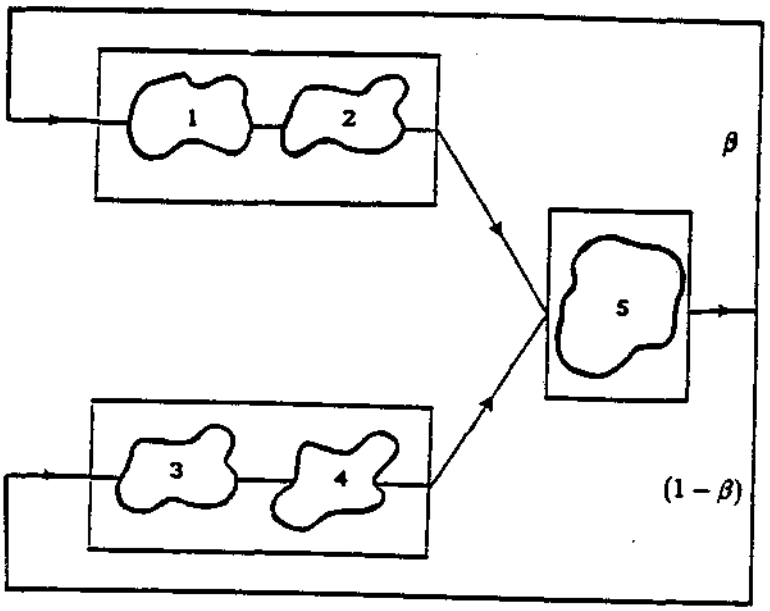
$$\begin{cases} t_p \leq N_p \quad (p = 1,2,3,4) \\ t_2 + t_3 + t_4 \leq U_2 \end{cases}$$

- "As long as one cluster is saturated servicing at all other clusters is to be stopped"

  while as in 5.1.3:

- "When $t_2 + t_3 + t_4 = U_2$ only departures from cluster 1 are blocked, while serving continues within all clusters, in particular also cluster 1. Jobs blocked to depart cluster 1 remain at the station at which they completed service".

(ii) As a combination of 5.2.1 and 5.3.1 consider the structure



with the capacity constraints:

$$\begin{cases} t_1 + t_2 \leq U_1 \\ t_3 + t_4 \leq U_2 \\ t_p \leq N_p \qquad (p = 1, \ldots, 5) \end{cases}$$

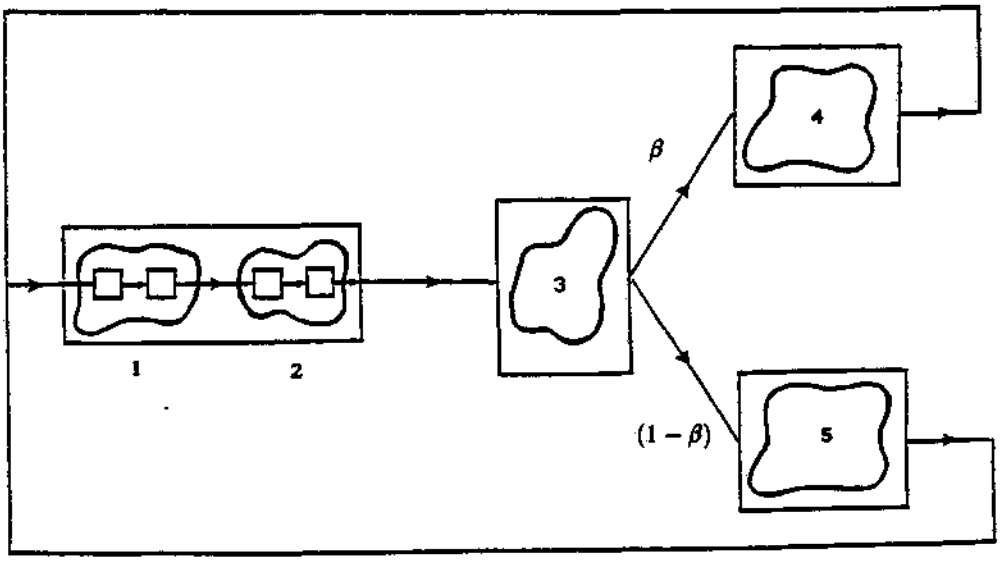
and under the following blocking protocol

- When cluster 5 is saturated serving at any station of all other clusters $1, \ldots, 4$ is stopped.
- When $t_1 = N_1$ or $t_2 = N_2$ or $t_1 + t_2 = U_2$ servicing at any station of cluster 5 is delayed by a factor $(1 - \beta)$. (A fraction β of capacity vanishes). Further, servicing is stopped also at

  cluster 2 when $t_1 = N_1$
  cluster 1 when $t_2 = N_2$

  Servicing at the other clusters in particular 3 and 4 is thus not interrupted.
- When $t_3 = N_3$ or $t_4 = N_4$ or $t_3 + t_4 = U_4$ similar protocol conditions are needed with delay factor $(1 - \beta)$ replaced by β.

(iii) As a combination of 5.2.1 and 5.3.1 again now consider the structure



with capacity constraints:

$$\begin{cases} t_p \leq N_p \qquad (p = 1, \ldots, 5) \\ t_1 + t_2 \leq U_1 \\ t_3 + t_4 + t_5 \leq U_2 \end{cases}$$

and under the following blocking protocol.

- When any of the clusters 1,2 or 3 is saturated servicing at all other clusters is stopped.
- When $t_1 + t_2 = U_1$ servicing at all clusters 3, 4 and 5 is stopped.
- When $t_3 + t_4 + t_5 = U_2$ servicing at cluster 1 and 2 is stopped.
- When $t_4 = N_4$ or $t_5 = N_5$ servicing at any station at clusters 1, 2 and 3 is delayed by a factor $(1 - \beta)$ or β respectively. Only when both $t_4 = N_4$ and $t_5 = N_5$ servicing at these clusters is stopped completely.

6 Stop = Recirculate

So far, the presentation has been restricted to the "stop" protocol as

(i) it corresponds to the natural communication protocol

(ii) it provides a simple tool to conclude product forms by using the station flow out = in principle (e.g. see the examples 3.9).

However, as mentioned earlier (in section 2.3), product form results for Jackson networks with departure blockings have been obtained in the literature (cf [31], [36]) under the so-called recirculate or lost and triggering protocol. Under this protocol, a job which is blocked to depart a Jackson network is instantaneously recirculated into that network as a newly arriving job.

Recently, in [52] it was shown that for single Jackson networks with departure blocking the "stop" and "recirculate" protocol have the same steady state distribution provided it is of product form for either of them. In example 4.9.7 this was also shown for the specific cyclic three cluster example.

This section will generalize these results to interconnected Jackson clusters in the general framework of section 3. As a particular consequence for all preceding examples the product form is retained if in the occasion of blocking rather than stopping service of a cluster we let service within that cluster continue but jobs leaving that cluster recirculate into that cluster. This somewhat amazing and counterintuitive result seems to be new.

6.1 Equivalence Result

Assume that the routing probabilities are of the form (5.2.1) under the recirculate protocol. More precisely, let p_{ij}, representing routing request probabilities be of the form:

(6.1) $$p_{ij} = g_i R_{pq} b_j \quad (\text{for } i \epsilon C_p, j \epsilon C_q, q \neq p)$$

reflecting that routing from one cluster to another is independent of the source and destination nodes within these clusters where g_i is the request probability of leaving its cluster after completing service at station i and where b_j is the probability of being assigned station j upon entering a cluster, while the direct request probability from i to j where i and j are within the same cluster is given by arbitrary probabilities $p_{ij}, (i, j \epsilon C_p)$. However, as in section 4, a routing from cluster p to q is accepted only with state dependent probability $B_{pq}(\bar{t})$, in cluster configuration $\bar{t}$.

Stop protocol

Under the stop protocol, we assume the form (4.1) with

$$B_{pp}(\bar{t}) = \sum_q R_{pq} B_{pq}(\bar{t}) \tag{6.2}$$

Effectively, any routing request within cluster p is thus blocked and rejected by the total blocking probability that upon departure of that cluster would be met.

Remark Note that (6.2) has been used in the preceding sections. See for example: (5.20), (5.27), (5.35) or (5.38).

Recirculate Protocol

Under the recirculate protocol, the actual routing probabilities $p_{ij}(\bar{n})$ are no longer given by (4.1) but now become:

$$p_{ij}(\bar{n}) = \begin{cases} p_{ij} B_{pq}(\bar{t}) & (i \epsilon C_p, j \epsilon C_q, q \neq p) \\ p_{ij} + \left\{ \sum_{q \neq p} [1 - B_{pq}(\bar{t})] \sum_{\ell \epsilon C_q} p_{i\ell} \right\} b_j & (i, j \epsilon C_p) \end{cases} \tag{6.3}$$

where the second possibility reflects the recirculation within the same cluster due to blocking. Further as in section 5.2 by substituting (6.1) in the traffic equations:

$$\begin{cases} \lambda_j = \sum_i \lambda_i p_{ij}, \\ \gamma_p = \sum_q \gamma_q R_{pq} \end{cases} \tag{6.4}$$

which are assumed to have unique normalized solutions $\{\lambda_1, \ldots, \lambda_N\}, \{\gamma_1, \ldots, \gamma_P\}$ one easily checks:

$$\begin{cases} \lambda_j = \gamma_p \tau_j & (j \epsilon C_p) \\ \tau_j = b_j + \sum_{i \epsilon C_p} \tau_i p_{ij} & (j \epsilon C_p) \end{cases} \tag{6.5}$$

The following general result now generalizes [52] to interconnected clusters

Theorem 6.1 (Equivalence result) Suppose that there exists a function $H(\cdot)$ such that:

$$\frac{H(\bar{m}+e_j)}{H(\bar{m}+e_i)} = \frac{\lambda_j}{\lambda_i} \quad (i,j \epsilon C_p) \tag{6.6}$$

and which satisfies the station balance equations (3.3) with $p_{ij}(\bar{n})$ of the form (4.1), that is under the *stop protocol*, then the same function $H(\cdot)$ also satisfies the station balance equations (3.3) with $p_{ij}(\bar{n})$ of the form (6.2), that is under the *recirculate protocol* and vice versa. The same product form expression (3.4) thus applies to both protocols, provided it applies to either of them.

Proof Consider the stop protocol and by virtue of (4.1) and (6.2) first note that the effective probability of routing out of station j, where for simplicity we assume $p_{ii} = 0$, is given by

$$\begin{aligned} B_{pp}(\bar{t}) \sum_{j \epsilon C_p} p_{ji} \quad & + g_i \sum_q R_{pq} B_{pq}(\bar{t}) \sum_{i \epsilon C_q} b_i \\ & = \left[\sum_q R_{pq} B_{pq}(\bar{t})\right] \left[\sum_{j \epsilon C_p} p_{ij} + g_i\right] \\ & = \left[\sum_q R_{pq} B_{pq}(\bar{t})\right] \end{aligned} \tag{6.7}$$

As a consequence, the station balance routing equations (4.2) now become

$$\begin{aligned} H(\bar{m}+e_j)\left[\sum_q R_{pq} B_{pq}(\bar{t})\right] \quad & = \sum_{q \neq p} \sum_{i \epsilon C_q} H(\bar{m}+e_i) p_{ij} B_{qp}(\bar{s}+e_i) \\ & + \sum_{i \epsilon C_p} H(\bar{m}+e_i) p_{ij} \left[\sum_q R_{pq} B_{pq}(\bar{t})\right] \end{aligned} \tag{6.8}$$

Under the recirculate protocol, that is (6.3) with (6.1) substituted, in contrast, the station balance routing equations (4.2) are given by:

$$\begin{aligned} H(\bar{m}+e_j) \quad & = \sum_{q \neq p} \left(\sum_{i \epsilon C_q} H(\bar{m}+e_i) p_{ij} B_{qp}(\bar{s}+e_i) \right. \\ & \left. + \sum_{i \epsilon C_p} H(\bar{m}+e_i) \left\{ p_{ij} + g_i \left[1 - \sum_q R_{pq} B_{pq}(\bar{t})\right] b_j \right\} \right. \end{aligned} \tag{6.9}$$

so that it suffices to show that substituting (6.8) in (6.9) verifies (6.9). By doing so we are left to verify:

$$H(\bar{m}+e_j)\left[1-\sum_q R_{pq}B_{pq}(\bar{t})\right]$$

(6.10)

$$=\sum_{i\epsilon C_p} H(\bar{m}+e_i)\left[1-\sum_q R_{pq}B_{pq}(\bar{t})\right][p_{ij}+g_i b_j]$$

Substituting $H(\bar{m}+e_i)=H(\bar{m}+e_j)\tau_i/\tau_j$ and recalling, as per (6.4) and (5.34) that,

(6.11)
$$\sum_{i\epsilon C_p}\tau_i g_i = 1$$

relation (6.10) is verified, recalling (6.5) again by

(6.12)
$$\sum_{i\epsilon C_p}\tau_i p_{ij}+\left(\sum_{i\epsilon C_p} g_i\tau_i\right)b_j=\tau_j$$

Remark 6.2 Note that (6.6) seems natural as the state dependent blocking only arises between clusters. For all examples of the preceding sections (6.6) is satisfied.

Corollary 6.1

For any product form example under the stop protocol as according to (6.2) and with source-destination independent cluster routing (6.1), the product form is retained if stopping service at a cluster is replaced by continuing service but recirculating departing jobs.

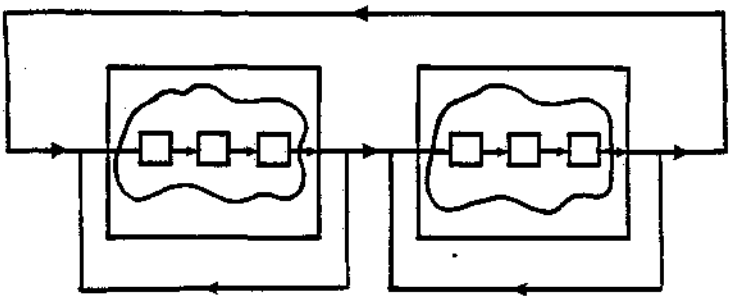
Remark 6.3 Note that most preceding examples satisfy (6.1) and (6.2), so that corollary 6.1 applies. Below we give some illustration.

6.2 Illustration of Results

Examples 6.1

(i) A first illustration has already been given by examples 4.9.6 and 4.9.7

(ii) Reconsider example 5.2.1 but now assume that service is always continued while a job leaving cluster 1 is recirculated



with probability $a_2(t_2)$ and for cluster 2 with probability $a_1(t_1)$. By virtue of theorem 6.1, the product form expression (5.25) remains valid

(iii) Reconsider example 5.3.1 under assumption (6.1), but now assume that service at clusters is never delayed or stopped, but in contrast, a job leaving cluster 3 is recirculated into cluster 3 with probability

$$\beta 1_{\{t_1=N_1\}} + (1-\beta)1_{\{t_2=N_2\}}.$$

and a job leaving cluster $p = 1, 2$ is recirculated into that cluster p with probability

$$1_{\{t_3=N_3\}}$$

The product form of section 5.2.2, that is with $H(\cdot)$ as per (5.1) still applies.

(iv) Reconsider example 5.3.1 but with servicing within all clusters always continued. However, a job having a cluster p is recirculated into that cluster as soon as one of the other clusters is currently saturated, i.e. with probability.

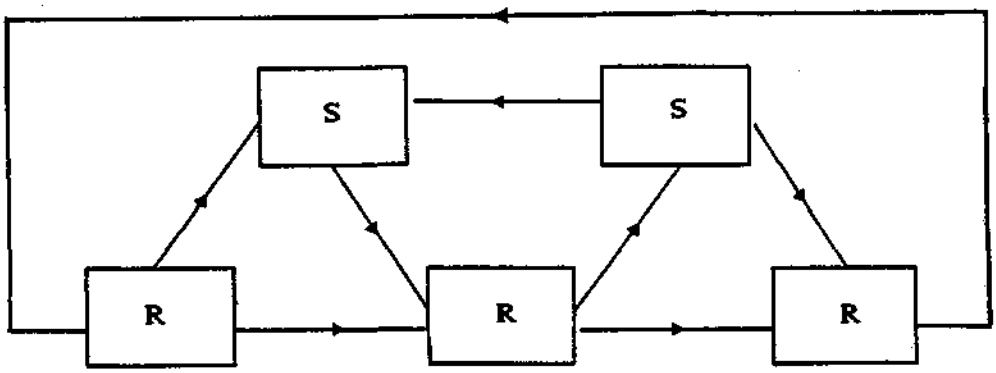
$$1_{\{t_q=N_q \text{ for at least one } q\neq p\}}$$

The product form (5.1) is still valid.

Remark 6.4 (Mixed extensions) The proof of theorem 6.1 can also be extended to allow mixed situations in which some clusters follow the "stop" and other the "recirculate" protocol. One only needs to distinguish these clusters separately in the equations (6.8) and (6.9). The details are left to the reader. We give one example below.

Example 6.2

Reconsider example 5.3.1 of an arbitrary network of interconnected clusters. As soon as one of the clusters is saturated, at any



other cluster, of fixed type S or R:

servicing is stopped if of type S
departing jobs are recirculated if of type R

As above, the product form (5.1) is still valid.

Remark 6.5 (Nested extensions) Similarly, in nested or hierarchical blocking structures such as in section 5.5, some levels of blocking may lead to the "stop" protocol while others to the "recirculate", without effecting the product form under the "stop" protocol allover. The technical details of this claim depend on the specific example in order but follow the details of theorem 6.1 at different blocking levels. Let us give just one illustration.

Example 6.3 Reconsider example 5.5 (iii), but under the following blocking conditions:

- When cluster 3 is saturated all other clusters stop servicing.
- When cluster 1 or 2 is saturated, cluster p departures at any cluster except the saturated one are recirculated into that cluster p.
- When $t_3 + t_4 + t_5 = U_2$ servicing at both clusters 1 and 2 is continued. However, a departure from cluster 2 is recirculated into cluster 1.
- When $t_1 + t_2 = U_1$ servicing at clusters 3, 4 and 5 is continued. However, a departure from cluster 4 or 5 is recirculated into cluster 3.
- When $t_4 = N_4$ or $t_5 = N_5$ service at all clusters is continued but cluster departures at clusters 1, 2 and 3 are recirculated into that cluster with probability β or $(1 - \beta)$ respectively.

Leaving the proof by substitution in the station balance equations for each of these situations to the reader, it is claimed that the product form (5.1), as in section 5.3, remains valid.

7 Extensions

7.1 Open Systems.

Similar results can be derived for open networks. To this end, one can either use standard limiting arguments or one can include an extra station to represent the exterior of the network. The main difference will arise in relations (3.1) - (3.3) in which an extra term at either side is to be included. By introducing corresponding local solutions $y_0(\bar{m})$ in (3.5) the results in the remainder can then be used directly. For example, the open analog of example 4.1 is an open tandem line of finite Jackson networks.

7.2 Multi-job Classes.

Clearly, when multiple jobclasses are allowed while jobs of one class route independently from other jobclasses, the above results directly transfer per jobclass. However, in practical situations with multiple jobclasses, interferences of different jobclasses is most common. In order to extend the present results to such situations, (e.g. similarly to [4, 5, 6, 17, 25, 28, 33, 35, 36]), a more refined notion of station-class balance is to be applied. Similar results can then be derived in analogy with the job-local-balance analysis in [28].

7.3 Non-exponential Servicing.

It is well known that product form expressions, as based upon station or local balance notions in the exponential case, remain valid without exponentiality assumptions provided "locally balanced", such as so-called "symmetric" service disciplines, are in order (e.g. [4, 5, 6, 16, 17, 28, 32, 35, 39]). Based upon the station balance property (3.3), this latter statement applies just as well in the present setting. For example, with each station behaving as an infinite server queue and the capacity limitations taken into account, the product form expression (3.4) can be shown to be insensitive for service distributional forms. More precisely, in the pure infinite server case it will apply with $G(\bar{n}) = [n_1!n_2!\ldots n_N!]^{-1}$.

8 Evaluation

Product form results for queueing networks with finite stations are known to be rather limited. Roughly speaking, only the case of a reversible routing from one station to another, such as of closed two-station networks, will be included.

In computer networking, telecommunications or manufacturing applications, however, capacity constraints on groups of stations (clusters) rather than individual stations may appear more realistic.

This paper has shown that for networks, with limited clusters, product form results are much wider applicable than for the restricted reversible case. Most notably, examples are concluded for networks with arbitrary non-reversible clusters and a reversible, weak-reversible or non-reversible routing between clusters, provided the blocking protocol in order is of special form.

Very roughly speaking the protocol needs to have conservative characteristics so as to avoid accumulation of saturations. Though such protocols are not always common, they may appear practical for designing a reliable system, for avoiding deadlocks or for balancing workloads. Moreover, the product form results provide one a reference framework which might be applicable to give quick numerical approximations or simple bounds for realistic non-product form situations.

The product form examples can be concluded by using local solutions of state dependent local traffic equations and a basic principle of stations balance. In particular, this principle can often be used at a non-mathematical level based on simple physical insights so as to conclude whether a system can have a product form or not and of what form.

To highlight the essential insights the presentation is restricted to closed networks with a single jobclass. However, extensions to open networks and multiple jobclasses are possible along the same lines. Further examples, of which must be many, are left to the reader and most notably solicited to practitioners to be found by using the simple intuitive balance principles.

Acknowledgements

I wish to express my gratitude most of all to the associate editor for handling and judging this rather unconventional paper, and to the referees for their time and positive comments by which the initial version was significantly improved. Further would I like to thank Les Berry and the Centre for Teletraffic Research at Bond University for providing me with the stimulating environment in which a major part was written. Last but not least I am indebted to Jeanette Niehus for her excellent typing job.

References

[1] Akyildiz, I.F. (1988), "On the exact and approximate throughput analysis of closed queueing networks with blocking", IEEE Trans. Softw. Eng. SE-14, 62-71.

[2] Akyildiz, I.F. (1989), "Exact analysis of queueing networks with rejection blocking" in: Perros and Altiok, Eds., Proc First International Workshop on Queueing Networks with Blocking, North-Holland, Amsterdam, 19-29.

[3] Akyildiz, I.F., Liebeherr, J. (1989), "Application of Norton's theorem on queueing networks with finite capacities", INFOCOM 89, 914-923.

[4] Akyildiz, I.F., von Brand, H. (1989), "Exact solutions for open, closed and mixed queueing networks with rejection blocking", Theoret. Comput. Sci. 64, 203-219.

[5] Akyildiz, I.F., von Brand, H. (1989), "Central server models with multiple job classes, state dependent routing and rejection blocking", IEEE Trans. Softw. Eng., to appear.

[6] Baskett, F., Chandy, M., Muntz, R. and Palacios, J. (1975), "Open, Closed and Mixed networks of queues with different classes of customers", J.A.C.M., 22.

[7] Berry L.T.M. and Henderson, W. (1989), "Some exact results in performance analysis of alternative routing communication networks",

[8] Bertsekas, D. and Gallager, R. (1987), "Data Networks", Prentice-Hall.

[9] Boorstyn, R.R., Kershenbaum, A., Maglaris, B. and Sahin, V. (1987), "Throughput analysis in multihop CDMA packet radio networks". IEEE Trans. Commun. 35, 267-274.

[10] Boucherie. R.J., and Van Dijk, N.M. (1990), "Spatial birth-death processes with multiple changes and applications to batch service networks and clustering processes", Adv. Appl. Prob. 22, 433-455.

[11] Boucherie. R.J., and Van Dijk, N.M. (1989), "Product forms for queueing networks with state dependent multiple job transitions", Research Report, Free University, Amsterdam. To appear: Adv. Appl. Prob.

[12] Brazio, J. and Tobagi, R. (1985), "Conditions for product form solutions in multihop packet radio network models", SEL Technical Report 85-274, Computer Systems Laboratory, Stanford University, Stanford CA, April 1985.

[13] Brazio, J. (1987), "Capacity analysis of multihop packet radio networks under a general class of channel access protocols and capture models", Technical Report CSL-TR-87-318, Computer Systems Laboratory, Stanford University, Stanford, CA.

[14] Burman, D.Y., Lehoczky, J.P. and Lin, Y. (1982), "Insensitivity of Blocking Probabilities in a circuit switching networks", J. Appl Prob. 21, 850-859.

[15] Buzacott J.A. and Yao D.D. (1986), "On queueing network models of flexible manufacturing systems", Queueing Systems 1, 29-66.

[16] Chandy, K.M., Howard, J.H. and Towsley, D.F. (1977), "Product form and local balance in queueing networks", J.A.C.M., 24, 250-263.

[17] Chandy, K.M. and Martin, A.J. (1983), "A characterization of product form queueing networks", J.A.C.M., 30, 286-299.

[18] Chin, K.E. (1986), "Networks of queues with state-dependent flows", PhD Dissertation, Georgia Institute of Technology.

[19] Chin, K.E. and Serfozo R.F. (1989), "Networks of queues with blocking and load balancing", in Proc Material Handling Research Forum, to appear.

[20] Conway, A.E. (1989), "Product form and insensitivity in circuit-switched networks with failing links", Performance Evaluation 9, 209-215.

[21] Cohen, J.W. (1957), "The generalized Engset formulae". Philips Telecommunications Review, 18, No 4, 158-170.

[22] Cohen, J.W. (1979), "The multiple phase service network with generalized processor sharing". Acta. Inf. 12, 245-284.

[23] Dallery, Y. (1986), "A queueing network model of flexible manufacturing systems consisting of cells", Proc IEEE Int. Conf on Robotics and automation.

[24] Dallery, Y. (1986), "Modelling of flexible manufacturing cells", in Kusiak, Ed, Modelling and Design of Flexible Manufacturing Systems, North-Holland Amsterdam, 289-300.

[25] Foshini, G.J. and Gopinath, B. (1983), "Sharing memory optimally", IEEE Trans. Commun. 31, 352-359.

[26] Henderson, W. and Taylor, P. (1988), "Alternative routing networks and interruptions", ITC-12, Torino, 5.1B.2.1.

[27] Hordijk, A. and Van Dijk, N.M. (1981), "Networks of queues with blocking", Performance '81, ed. K.J. Kylstra, North Holland, 51-65.

[44] Schassberger, R. (1978), "The insensitivity of stationary probabilities in networks of queues", Adv. Appl. Prob. 10, 906-912.

[45] Schassberger, R. (1979), "A definition of discrete product form distributions in networks of queues", Zeitschrift für Operations Research 23, 189-195.

[46] Serfozo, R. (1989), "Markovian network processes: Congestion dependent routing and processing", Queueing Systems 5, 5-36.

[47] Tobagi, F.A. and Shur, D.H. (1986), "Performance evaluation of channel access schemes in multihop packet radio networks with regular structure by simulation", Computer Networks and ISDN Systems 12, 39-60.

[48] Towsley, D.F. (1986), "Queueing network models with state-dependent routing", J.A.C.M 27, 323-337.

[49] Van Dijk, N.M. (1988), "Product form for random access schemes", To appear: Computer Networks and ISDN Systems.

[50] Van Dijk, N.M. (1988), "Simple bounds for queueing systems with breakdowns", Performance Evaluation 8, 117-128.

[51] Van Dijk, N.M. (1989), "A simple bounding methodology for non-product queueing networks with blocking", in Queueing Networks with Blocking (Eds H.G. Perros and T. Altiok), North-Holland, 3-17.

[52] Van Dijk, N.M. (1989), "Stop = Recirculate for exponential product form queueing networks with departure blocking", To appear: Oper. Res. Letters.

[53] Van Dijk, N.M. (1990), "An insensitive product form for discrete-time communications networks", To appear: Performance '90.

[54] Van Dijk, N.M. and Akyildiz, I.F. (1990), "Networks with mixed processor sharing parallel queues and common pools", Performance '90, North-Holland, 35-49.

[55] Van Dijk, N.M. and Tijms H.G. (1986), "Insensitivity in two node blocking models with applications", in: Boxma and Cohen Eds, Teletraffic Analysis and Computer Performance Evaluation, North-Holland, Amsterdam.

[56] Van Dijk, N.M. and Veltkamp, J.P. (1988), "Product form for stochastic interference systems", Probability Engineering and Informational Sciences 2, 355-376.

[57] Van Dijk, N.M. and Van der Wal, J. (1989), "Simple bounds and monotonicity results for finite multi-server exponential tandem queues", Queueing Systems 4, 1-16.

[28] Hordijk, A. and Van Dijk, N.M. (1983), "Networks of queues. Part I: Job-local-balance and the adjoint process. Part II: General routing and service characteristics", Lecture notes in control and Informational Sciences, Springer-Verlag, 60, 158-205.

[29] Hordijk, A. and Van Dijk, N.M. (1983), "Adjoint processes, job-local-balance and insensitivity of stochastic networks", Bull. 44-th Session Int. Stat. Inst., 50, 776-788.

[30] Iversen, V.B. (1980), "The A-formula", Teleteknik 2, 64-79.

[31] Jackson, J.R. (1963), "Jobshop-like queueing systems", Mngt. Sci., 10, 131-142.

[32] Kaufman, J. (1981), "Blocking in a shared resource environment", IEEE Trans. Commun. 29, 1474-1481.

[33] Kelly F.P. (1979), "Reversibility and stochastic networks", Wiley.

[34] Kleinrock, L. (1975), "Queueing systems", Vol II, Wiley.

[35] Krzezinski, A.E. (1987), "Multiclass queueing networks with state-dependent routing", Perf. Evaluation 7, 125-144.

[36] Lam, S.S. (1977), "Queueing networks with population size constraints", IBM J. Res. Develop., 370-378.

[37] Mitrani, I. (1987), "Modelling of computer and communication systems", Cambridge Press.

[38] Nelson, R. and Kleinrock, L. (1985), "Rude-CSMA: A multihop channel access protocol", IEEE Commun. 33, 785-791.

[39] Noetzel, A.S. (1979), "A generalized queueing discipline for product form network solutions", J.A.C.M. 26 (4), 779-793.

[40] Onvural, R.O. (1989), "A note on the product form solutions of multiclass closed queueing networks with blocking", Performance Evaluation 10, 247-253.

[41] Perros, H.I. (1989), "A bibliography of papers on queueing networks with finite capacity queues", Performance Evaluation 10, 255-260.

[42] Pittel, B. (1979), "Closed exponential networks of queues with saturation: The Jackson type stationary distribution and its asymptotic analysis", Math. Operat. Res. 4, 367-378.

[43] Rubin, I. and Lee, J.K., "Performance analysis of interconnected metropolitan area circuit-switched networks", IEEE Trans. Comm. 36, 171-185.

[58] Walrand, J. (1988), "Introduction to queueing networks", Prentice-Hall.

[59] Whittle, P., (1986), "Systems in stochastic equilibrium", Wiley.

[60] Yao, D.D. and Buzacott, J.A. (1985), "Modeling a class of state dependent routing in flexible manufacturing systems", Ann. Oper. Res. 3, 153-167.

[61] Yao, D.D. and Buzacott, J.A. (1985), "Queueing models for a flexible machining station station Part ii: the method of Co phases", Eur. J. Oper. Res. 19, 241-252.

[62] Yao, D.D. and Buzacott, J.A. (1987), "Modeling a class of flexible manufacturing systems with reversible routing", Oper. Res. 35, 87-93.