TADEUSZ CZACHÓRSKI

Materials for the seminar

Queueing Theory and Diffusion Approximation with their Applications

28 May – 10 June 2026

Centre for Stochastic Modelling and Simulation
School of Computer Science
University of Petroleum and Energy Studies, Dehradun

Contents

Introduction

Queueing theory, as a core component of both operations research and the theory of stochastic processes, provides a rigorous framework for analyzing systems in which random demand competes for limited service capacity. Beyond its traditional role in telecommunications, it is widely applied in financial markets to model order arrivals and execution delays in stock trading systems, as well as in commercial environments such as shopping centers and service facilities to optimize customer flow and resource allocation. In urban infrastructure, queueing models are used to study pedestrian dynamics in underground passages and transportation hubs, helping to mitigate congestion and improve safety. In healthcare systems, they support the design and management of patient flow, enabling more effective allocation of medical resources and reduction of waiting times. Overall, queueing theory offers a unifying analytical approach for studying congestion phenomena across a broad range of engineered and socio-economic systems.

A basic model in queueing theory considers a stream of customers (or jobs) arriving to a service facility consisting of one or more servers. The fundamental stochastic inputs of the model are the arrival process, which characterizes the pattern of customer arrivals over time, and the service-time distribution, which describes how long each customer requires service. Based on these inputs, the objective is to analyze key performance measures such as the waiting time experienced by customers and the queue length, either in terms of their expected values or their full probability distributions. In systems with finite capacity, an additional important aspect is the possibility of losses (or blocking), whereby arriving customers are rejected when the system is full, leading to the notion of blocking probability and effective arrival rate. Depending on the level of detail required and the available information, one may adopt simpler models—such as Markovian systems with exponential assumptions—or more general and realistic formulations with arbitrary distributions and additional structural features (e.g., finite capacity, priorities, or time dependence). This flexibility allows queueing models to balance analytical tractability with descriptive accuracy.

Queueing theory has long been a cornerstone of performance analysis in computer systems and communication networks. Its origins can be traced to the pioneering work of *Agner Krarup Erlang* in the early twentieth century, who introduced probabilistic models to analyse congestion in telephone networks. Erlang's formulas for loss and delay laid the foundation for a discipline that would grow in both mathematical depth and practical importance.

In the decades that followed, queueing theory was significantly advanced through the formalisation of stochastic processes and the development of analytical frameworks. A major milestone was the introduction of the now-standard classification of queueing systems by *David G. Kendall*, whose notation ($M/M/1$, $G/G/1$, etc.) provided a unifying language for describing a wide range of models. At the same time, fundamental theoretical results—such as the Pollaczek–Khinchine formula and the work of *Félix Pollaczek* and *Aleksandr Khinchin*—extended the applicability of queueing models beyond purely Markovian assumptions.

With the rise of computer systems in the second half of the twentieth century, queueing theory found new domains of application. Researchers such as *Hisashi Kobayashi* contributed significantly to the analysis of computer and communication systems, bridging the gap between abstract stochastic models and engineering practice. The development of product-form networks by *Gordon* and *Newell*, and later the seminal work of *Frank Kelly*, enabled the tractable analysis of complex interconnected systems. These advances were complemented by the emergence of efficient computational techniques, including Mean Value Analysis, which allowed practitioners to evaluate large systems without resorting to exhaustive state-space enumeration.

Further progress came with the introduction of more flexible and expressive models. The work of *Erol Gelenbe*, particularly on diffusion approximation and G-networks, expanded the modelling capabilities of queueing theory by incorporating features such as negative customers and triggers, opening new avenues for representing adaptive and intelligent systems. At the same time, the increasing variability and complexity of real-world traffic led to the development of non-Markovian models and approximation techniques capable of capturing more realistic behaviour.

In recent decades, the focus has shifted towards methods that remain tractable at large scales and under dynamic conditions. Diffusion approximations, rooted in the theory of stochastic processes, provide powerful tools for analysing systems in heavy-traffic regimes by approximating them with continuous stochastic models. Fluid-flow approximations, in turn, treat traffic as a continuous deterministic flow, enabling scalable analysis of systems with thousands or even hundreds of thousands of nodes.

The methods presented in this book reflect this evolution and demonstrate their continuing relevance in modern applications. Analytical techniques such as performance bounds, Mean Value Analysis, and Markovian models provide fundamental insights into system behaviour and enable precise evaluation under well-defined assumptions. At the same time, the increasing complexity and scale of contemporary systems—particularly computer networks—have motivated the development of advanced approaches capable of capturing realistic dynamics.

In this context, the detailed treatment of diffusion and fluid-flow approximations constitutes a central contribution of this work. These approaches extend classical models by enabling the analysis of transient states and time-dependent behaviour, which are critical for understanding real-world systems operating under dynamic conditions. While steady-state analysis remains important, many practical scenarios—such as congestion episodes, traffic bursts, and adaptive control mechanisms—require tools that can accurately describe system evolution over time.

It is worth emphasising that a significant portion of the results presented in the chapters devoted to diffusion and fluid-flow models originates from the author's own research, conducted in collaboration with colleagues over many years. This gives the book not only a comprehensive character but also a distinctive perspective grounded in original contributions to the field.

By combining classical methods with modern approximation techniques, this book offers a coherent framework for the evaluation of complex systems. The approaches discussed herein are essential for the practical analysis and design of computer systems and networks, providing both theoretical understanding and tools applicable to real-world problems.

This work will be of interest to researchers, practitioners, and graduate students seeking a rigorous yet applicable treatment of queueing models and their role in performance evaluation.

This book provides a comprehensive overview of analytical and approximate methods for modelling and evaluating queueing systems, with a particular focus on their application to

communication networks and large-scale systems.

We begin with **performance bounds**, which establish fundamental limits on system behaviour. These bounds provide quick estimates of key performance metrics such as delay, throughput, and queue lengths without requiring detailed system analysis. They are especially useful for validating models and identifying feasible operating regions.

Next, the book introduces **Mean Value Analysis (MVA)**, a recursive technique for computing average performance measures in closed queueing networks. MVA enables efficient evaluation of systems with multiple service centres and circulating customers, providing insights into resource utilisation and response times.

The discussion then turns to **single-station Markovian models**, including classical $M/M/1$, $M/M/c$, and related systems. These models rely on the memoryless property of exponential distributions, allowing for exact analytical solutions. They serve as a foundation for understanding stochastic dynamics such as queue evolution, stability conditions, and steady-state distributions. Next, we discuss **network models with a product solution**, including the most general BCMP model.

We extend this framework to **non-Markovian models**, where service or interarrival times follow general distributions. Although exact solutions are typically not available, various approximation techniques and numerical methods allow us to capture more realistic system behaviour, including variability and correlation effects.

The book then presents **diffusion approximations**, first from a theoretical perspective and then through practical applications. Diffusion models approximate queueing processes by continuous stochastic processes (typically reflected Brownian motion), enabling the analysis of systems under heavy traffic conditions. These methods provide accurate estimates of delay distributions, queue lengths, and transient behaviour in regimes where classical models become intractable.

Finally, we examine **fluid-flow approximations**, in which packet traffic is treated as a continuous deterministic flow. This approach is particularly well suited for large-scale systems, such as communication networks with thousands of nodes. Fluid models enable scalable analysis of congestion control mechanisms (e.g., TCP), routing dynamics, and long-term system performance. While they sacrifice stochastic detail, they offer tractable and computationally efficient insights into system-wide behaviour.

Taken together, these methods form a hierarchy of modelling approaches, ranging from exact stochastic models to large-scale deterministic approximations. The choice of method depends on the required level of accuracy, system size, and the phenomena of interest. By combining these approaches, the book provides a versatile toolkit for analysing modern complex systems, balancing mathematical rigour with practical applicability.

Chapter 1

Operational Approach and Bounds

1.1 Basic notions

This section presents the so-called *operational approach* [11, 27] to queueing systems. The term emphasizes that all quantities considered here are easy to measure in a real system, that their values are valid only for a given observation period, and that no assumptions are made about their probabilistic nature.

Assume that a system is observed during a time interval T. We introduce several quantities related to its operation:

- number A of customers (jobs, tasks) arriving during the observation interval,
- number C of customers completed during this interval,
- total time S within period T during which the system is busy, i.e. there is work to do; during the remaining time $T - S$ the system is idle.

We can now compute the input traffic intensity (mean number of customers arriving per unit time)

$$\lambda = A/T$$

and the system throughput (mean number of customers leaving the system after completing service)

$$X = C/T.$$

The mean service time of a customer is

$$\bar{B} = S/C.$$

The bar above selected symbols in our notation indicates that they represent mean values.

We assume that the system is stable, i.e. that the mean interval between arrivals is greater than the mean service time. In this case the system is able to serve all customers, and they do not accumulate indefinitely in the system. Therefore, over a sufficiently long observation period,

$$\lambda = X.$$

Example 1.1

If during the observation period $T = 1000$ *sec the central processing unit (CPU) was busy for*

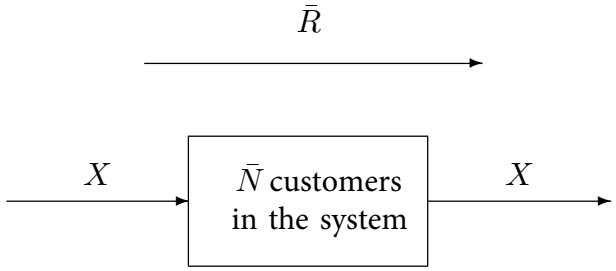


Figure 1.1: Little's law: $\bar{N} = X\bar{R}$

$S = 300$ *sec and* $C = 150$ *tasks were completed, then the CPU throughput was* $X = 150/1000 = 0.15$ *jobs/sec, its utilization was* $U = 300/1000 = 0.3 = 30\%$*, and the mean service time was* $\bar{B} = 300sec/150 = 2$ *sec.*

■

Note that

$$U = \frac{S}{T} = \frac{S}{C}\frac{C}{T} = \bar{B}X, \tag{1.1}$$

that is, utilization is equal to the product of throughput and mean service time. This relation is called the **utilization law**.

Let $\bar{R}$ denote the *response time* of the system, i.e. the mean time a customer spends in the system, either waiting or being served. Equivalently, it is the mean time between the arrival and departure instants.

If during period T there were on average $\bar{N}$ customers in the system, then their total sojourn time in the system was $\bar{N}T$. On the other hand, if C jobs were completed during T and each of them spent on average $\bar{R}$ units of time in the system, then

$$\bar{N}T = C\bar{R},$$

or

$$\bar{N} = \frac{C}{T}\bar{R} = X\bar{R}\,. \tag{1.2}$$

This relation, introduced more formally in [**?**], is called **Little's law** (or **Little's formula**). We may apply Little's law either to a single server with a queue or to an entire queueing network. In the latter case, the number of customers in the network is the sum of the numbers of customers at all stations; see Fig. 1.2c.

We may also apply Little's law to the server alone, excluding its queue; see Fig. 1.2a. In this case the mean number of customers in the system is

$$\bar{N} = \frac{1 \times S + 0 \times (T - S)}{T} = \frac{S}{T} = U,$$

and the response time is simply the service time. Thus, we recover formula (1.1).

Example 1.2

The throughput of a system, determined over an observation period of $T = 5$ *hours, is* $X = 3$ *jobs/sec. During the same period, the observed mean number of customers in the system was 4. What was the response time?*

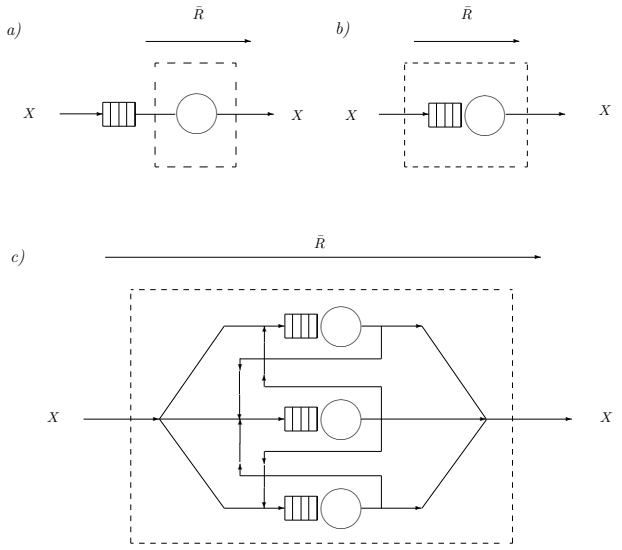


Figure 1.2: Little's law for a server, an entire service system, and a queueing network

Using Little's law, we obtain

$$\bar{R} = \frac{\bar{N}}{X} = \frac{4\,\text{jobs}}{3\,\text{jobs/sec}} = 1.33\,\text{sec}\,.$$

■

Consider now a system composed of M interconnected stations. As customers circulate through the network, they visit different stations with different frequencies. Let V_i denote the mean number of visits to station i during the total residence time of a customer in the network.

Since a customer is seen only once at the entrance to the network and visits station i on average V_i times, for $i = 1, \ldots, M$, the throughput of station i is V_i times greater than the throughput of the entire network:

$$X_i = V_i X.$$

This relation is called the **forced flow law**. The total mean service time received by a customer at station i during all its visits is

$$\bar{D}_i = V_i \bar{B}_i.$$

The total service demand in the network is therefore

$$\bar{D} = \sum_{i=1}^{M} \bar{D}_i.$$

Let us now consider the time-sharing system shown in Fig. 1.3. It is very similar to the first queueing model used to evaluate a computer system [111]. The system consists of a number of terminals and a computing subsystem represented here by a central processing unit (CPU) and disks (in the original Sherr model there was only one server).

We assume that each user terminal sends only one task at a time (so there are no queues at the terminals) to be processed and receives the result after the response time of the computing subsystem, whose mean value is $\bar{R}$. The user then analyzes the result for a period called the

thinking time, whose mean value is $\bar{Z}$, and sends the next task for processing. Thus, each task alternates between two phases, with mean durations $\bar{R}$ and $\bar{Z}$.

The number N of tasks circulating in the system is equal to the number of active terminals. A portion $\bar{N}_Z$ of them is at terminals, while the remaining portion $\bar{N}_R$ is being processed, so that

$$N = \bar{N}_Z + \bar{N}_R.$$

Applying Little's law to both parts of the system, we may write

$$\bar{N}_Z = X\bar{Z}, \qquad \bar{N}_R = X\bar{R},$$

hence

$$N = (\bar{R} + \bar{Z})X,$$

and therefore

$$\bar{R} = \frac{N}{X} - \bar{Z}\,.$$

The route followed by tasks within the computing subsystem is determined by events occurring during task execution. Each time a task needs to access a disk (to read or write data), it relinquishes the CPU and joins the queue of the selected disk; at that moment, the CPU is assigned to the next task waiting in the CPU queue. The same mechanism is represented in the model: a customer corresponding to the task moves from the CPU to the queue of the appropriate disk. The service time at the disk corresponds to the duration of the read/write operation. After that, the task may continue its computation; in the model, the customer returns from the disk to the CPU and joins its queue, waiting for its turn to be processed again. Thus, the service time per visit to the CPU is the uninterrupted computation time between two successive I/O operations. During the final visit to the CPU, the task completes execution and returns to its terminal.

Using the laws formulated above, we are sometimes able to determine, from measurements taken in one part of the system, quantities characterizing other parts of the system, as illustrated by the following example.

Example 1.3

In the system under consideration there are $N = 25$ active terminals; hence 25 tasks are either being processed or are at terminals in the thinking phase. The mean thinking time is $\bar{Z} = 18$ sec. Measurements show that the utilization of disk i is $U_i = 0.30$, and the mean service time per visit to this disk is $\bar{B}_i = 0.025$ sec. Completion of a task requires, on average, 20 *read/write operations at this disk. What is the mean response time $\bar{R}$ of the system?*

The throughput of disk i is

$$X_i = \frac{U_i}{\bar{B}_i} = \frac{0.30}{0.025} = 12\,\frac{1}{\text{sec}}\,,$$

while the throughput of the whole time-sharing system is

$$X = \frac{X_i}{V_i} = \frac{12}{20}\,\frac{1}{\text{sec}} = 0.6\,\frac{1}{\text{sec}}\,.$$

Hence the response time is

$$\bar{R} = \frac{25}{0.6}\,\text{sec} - 18\,\text{sec} = 23.7\,\text{sec}\,.$$

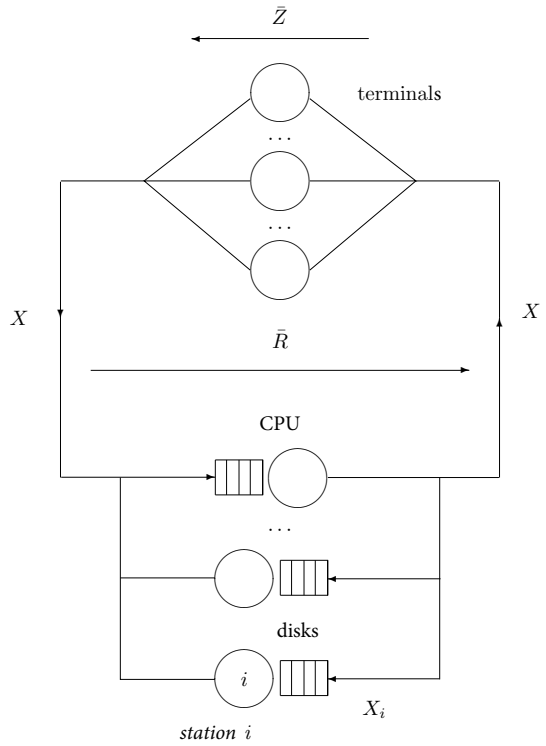


Figure 1.3: A model of a time-sharing system

■

In the next example we introduce the notion of the **bottleneck**. The bottleneck is the station with the highest utilization factor. It works more intensively than the other stations, its total service demand is the largest,

$$\bar{D}_{\max} = \max_i \bar{D}_i,$$

and therefore it tends to have the longest queue and the longest response time. It is the station that limits the performance of the whole system, that is, the weakest point of the system. If we want to improve the performance of the entire system, the first step should be to improve its bottleneck. Improving other parts of the system is of little use, since it will have only a minor effect on overall performance.

Example 1.4

In a time-sharing system represented by the model in Fig. 1.3, $N = 15$ *users are active. The CPU (station 1) cooperates with two disks: a slow disk (station 2) and a fast disk (station 3). The mean service times at the stations are* $\bar{B}_1 = 0.05$ *sec,* $\bar{B}_2 = 0.07$ *sec,* $\bar{B}_3 = 0.02$ *sec. The mean numbers of visits to the stations are* $V_1 = 20$, $V_2 = 12$, $V_3 = 7$, *and the terminal thinking time is* $\bar{Z} = 15$ *sec.*

- *Which of the system resources is the most heavily loaded? What is the ratio of the station utilizations?*

  The most heavily loaded station is the one that performs the greatest total amount of work per customer, i.e. the one with the largest total service demand $\bar{D}_i$. We compute

$$\begin{aligned} \bar{D}_1 &= \bar{B}_1 V_1 = 0.05 \text{ sec} \times 20 = 1.00 \text{ sec}, \\ \bar{D}_2 &= \bar{B}_2 V_2 = 0.07 \text{ sec} \times 12 = 0.84 \text{ sec}, \\ \bar{D}_3 &= \bar{B}_3 V_3 = 0.02 \text{ sec} \times 7 = 0.14 \text{ sec}. \end{aligned}$$

  Thus, the CPU is the bottleneck of the system.

  Although we cannot yet determine the absolute values of the station utilizations, we can determine their ratios:

$$\frac{U_i}{U_j} = \frac{X_i \bar{B}_i}{X_j \bar{B}_j} = \frac{X V_i \bar{B}_i}{X V_j \bar{B}_j} = \frac{\bar{D}_i}{\bar{D}_j} ,$$

  which gives

$$U_2 = 0.84\, U_1, \qquad U_3 = 0.14\, U_1.$$

- *What is the minimum mean response time of the system?*

  The minimum mean response time is the sum of the total service demands of all stations. We assume that the customer never has to wait: whenever it arrives at a station, all other customers are elsewhere. Therefore,

$$\bar{R}_{\min} = \sum_{i=1}^{3} \bar{D}_i = 1.00 \text{ sec} + 0.84 \text{ sec} + 0.14 \text{ sec} = 1.98 \text{ sec} .$$

- *Which device becomes the bottleneck of the system if:*

  - *we use a faster CPU with $\bar{B}_1 = 0.04$ sec?*
    Then
$$\bar{D}_1 = 0.04 \times 20 = 0.8 \text{ sec},$$
    so $\bar{D}_1 < \bar{D}_2$, and the slow disk becomes the bottleneck.
  - *the slow disk is replaced by a slightly faster one with $\bar{B}_2 = 0.06$ sec?*
    In this case the CPU clearly remains the bottleneck; the disproportion between device utilizations becomes even greater.

- *What will be the utilization factor of the fast disk if the number of active terminals N increases significantly?*

  The system will then become heavily loaded, which means that the utilization of the bottleneck will approach 1, i.e. $U_1 \approx 1$; the CPU will be busy almost continuously. Since we know that
$$U_3 = 0.14\, U_1,$$
  the utilization of the fast disk cannot exceed 14%.

- *In the system, $N = 30$ users are active. By how much should the CPU be accelerated to ensure a response time of 12 sec?*

  If the response time is $\bar{R} = 12$ sec, then the throughput of the whole system is

  $$X = \frac{N}{\bar{R} + \bar{Z}} = \frac{30}{12 + 15} \approx 1.11 \, \frac{1}{\text{sec}},$$

  and the throughput of the CPU is

  $$X_1 = V_1 X = 20 \times 1.11 = 22.22 \, \frac{1}{\text{sec}}.$$

  The maximum throughput of the current CPU, when it is busy 100% of the time, i.e. when $U_1 = 1$, is

  $$X_{1\max} = \frac{U_{1\max}}{\bar{B}_1} = \frac{1}{\bar{B}_1} = \frac{1}{0.05} = 20 \, \frac{1}{\text{sec}},$$

  which is below the required value.

  Therefore, the CPU speed should be increased by at least a factor of

  $$\frac{22.22}{20} = 1.11,$$

  that is, by about 11%.

- *In the system with $N = 30$, what should be done to ensure a mean response time of $\bar{R} = 10$ sec?*

  In this case, both the CPU and the slow disk must be accelerated, because the speed of either device alone is insufficient to ensure the required response time.

■

1.2 Asymptotic bounds for throughput and response time

Below we present lower and upper bounds on the throughput and response time of a time-sharing system represented by the model in Fig. 1.3. We expect throughput and response time to be increasing functions of N, but at this point we are not yet able to determine their exact values (this will be done in the next chapter using more sophisticated models). Therefore, we derive here simple formulas, easy to compute, that bound the actual values of $X(N)$ and $\bar{R}(N)$:

$$X_{\min}(N) \le X(N) \le X_{\max}(N)\,, \qquad \bar{R}_{\min}(N) \le \bar{R}(N) \le \bar{R}_{\max}(N)\,.$$

Two kinds of bounds are commonly used in the literature: the asymptotic bounds presented here, and the balanced-system bounds presented in the next section. The term *asymptotic* means here that we consider the system in two limiting cases: under low load and under high load. The presentation below follows [30]. The original results may be found in [85, **?**, 121].

Assume first that only one user is active, i.e. $N = 1$. The task issued by that user is alone in the system and therefore never has to wait for service while circulating through the network. Hence

$$\bar{R}(1) = \bar{D}.$$

By Little's law,

$$X(1) = \frac{1}{\bar{R} + \bar{Z}} = \frac{1}{\bar{D} + \bar{Z}}\,.$$

If the number of customers N increases, it is still possible (although less and less probable) that a customer entering a station finds it empty because all other customers are elsewhere. In the most favorable case, this happens at every station visited during its route, and thus

$$\bar{R}_{\min}(N) = \bar{D}.$$

The minimum response time corresponds to the maximum throughput:

$$X_{\max}(N) = \frac{N}{\bar{D} + \bar{Z}}.$$

On the other hand, an unlucky customer arriving at a station may find all other $N-1$ customers already there. If this happens at every station it visits, its response time will be maximal:

$$\bar{R}_{\max}(N) = N\bar{D},$$

and the corresponding throughput will be minimal:

$$X_{\min}(N) = \frac{N}{N\bar{D} + \bar{Z}}.$$

We may add one more condition: for each station,

$$U_i \le 1, \qquad i = 1, \ldots, M,$$

where M is the number of stations. Since

$$U_i = X_i \bar{B}_i = X(N)\bar{D}_i,$$

it follows that

$$X(N) \le \frac{1}{\bar{D}_i},$$

and in particular

$$X(N) \le \frac{1}{\bar{D}_{\max}}.$$

This means that if each customer spends, on average, at least $\bar{D}_{\max}$ units of time at the bottleneck station, then customers cannot leave the network at intervals shorter than $\bar{D}_{\max}$, and therefore the throughput cannot exceed $1/\bar{D}_{\max}$.

Taking this into account, we obtain

$$\frac{N}{N\bar{D} + \bar{Z}} \;\le\; X(N) \;\le\; \min\left\{\frac{1}{\bar{D}_{\max}}, \frac{N}{\bar{D} + \bar{Z}}\right\}. \tag{1.3}$$

Recalling that

$$X(N) = \frac{N}{\bar{R}(N) + \bar{Z}},$$

we may transform inequality (1.3) into

$$\bar{D} \;\le\; \bar{R}(N) \;\le\; \max\left\{N\bar{D}_{\max} - \bar{Z},\, N\bar{D}\right\}. \tag{1.4}$$

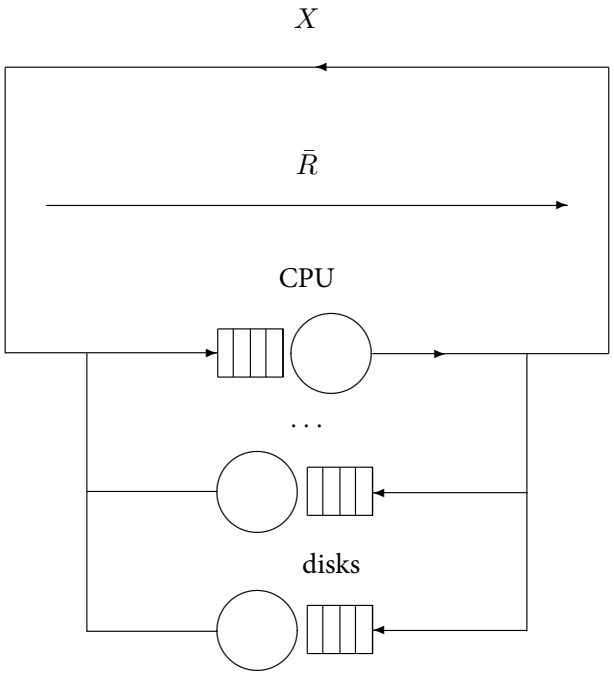


Figure 1.4: A model of a batch-loaded time-sharing system

In contrast to the *terminal-load system*, we may also consider a *batch-load system*, in which, instead of terminals operating interactively, we model a system executing exactly N jobs in parallel. Whenever a customer goes through the upper path, it is completed and immediately replaced by a new one waiting outside the system in an input queue; see Fig. 1.4. In this case $\bar{Z} = 0$, and bounds (1.3)–(1.4) become

$$\begin{aligned} \frac{1}{\bar{D}} &\leq X(N) \leq \min\left\{\frac{1}{\bar{D}_{\max}}, \frac{N}{\bar{D}}\right\}, \\ \bar{D} &\leq \bar{R}(N) \leq \max\left\{N\bar{D}_{\max},\, N\bar{D}\right\}. \end{aligned} \tag{1.5}$$

The curves representing bounds (1.3)–(1.5) for exemplary parameter values are shown in Figs. 1.5 and 1.6, together with the balanced-system bounds presented in the next section.

1.3 Bounds based on balanced systems

We now introduce another class of bounds, which are usually tighter, i.e. the gap between the lower and upper bounds is smaller, although their form is somewhat more complex. Therefore, we present them here only for the batch-load case ($\bar{Z} = 0$).

A balanced system is one in which the total service demand at each device is the same:

$$\bar{D}_1 = \bar{D}_2 = \ldots = \bar{D}_M = \frac{\bar{D}}{M}.$$

As a consequence, the utilization factors of all stations are equal,

$$U_1 = \ldots = U_M.$$

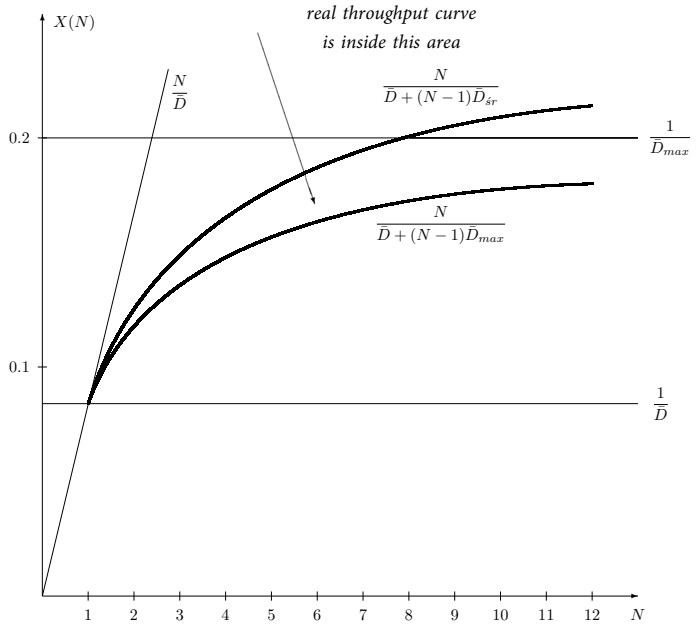


Figure 1.5: Batch load – asymptotic bounds (thin lines) and balanced-system bounds (thick lines) for throughput; data from Exercise 1.5

We will prove later that in such a system, when $\bar{Z} = 0$, the utilization is given by

$$U_i(N) = \frac{N}{N + M - 1}\,, \tag{1.6}$$

Therefore, the throughput of the system is

$$X(N) = \frac{X_i(N)}{V_i} = \frac{U_i(N)}{\bar{B}_i}\,\frac{1}{V_i} = \frac{U_i(N)}{\bar{D}_i} = \frac{N}{N + M - 1}\,\frac{1}{\bar{D}_i}\,. \tag{1.7}$$

and the mean response time is

$$\bar{R}(N) = \frac{N}{X(N)} = (N + M - 1)\bar{D}_i\,. \tag{1.8}$$

The system we want to evaluate is generally not balanced. We therefore bound its performance by two balanced systems. One of them (system A) is better than the system under consideration, i.e. it has a smaller response time and higher throughput. The other (system B) is worse, i.e. it has a longer response time and lower throughput. Both are balanced systems.

In the original system, the total service demands at the stations are $\bar{D}_1, \ldots, \bar{D}_M$ and their sum is $\bar{D}$.

In system A, the total amount of work $\bar{D}$ is redistributed equally among all stations, so that each station has the same service demand

$$\bar{D}_{ave} = \frac{\bar{D}}{M}.$$

Since the total amount of work is the same as in the original system, but it is distributed evenly among all stations (with no bottleneck), we expect the performance of this system to be better.

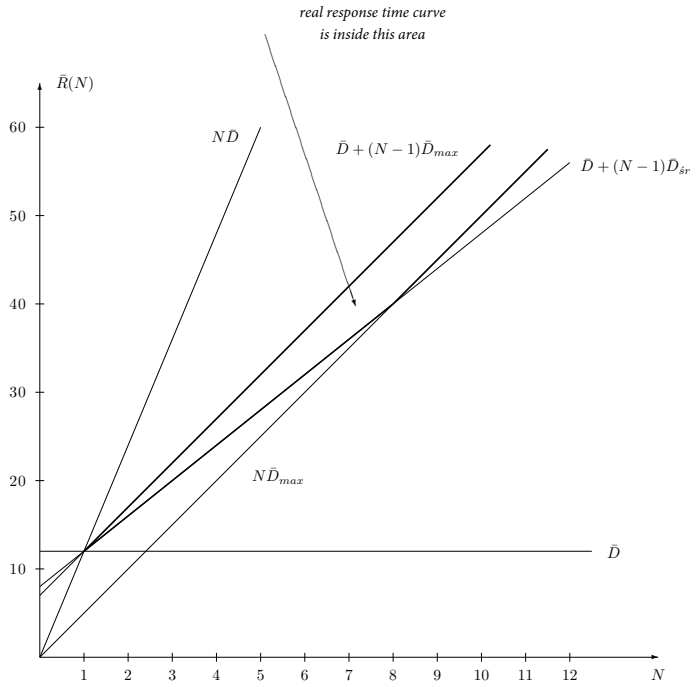


Figure 1.6: Batch load – asymptotic bounds (thin lines) and balanced-system bounds (thick lines) for response time; data from Exercise 1.5

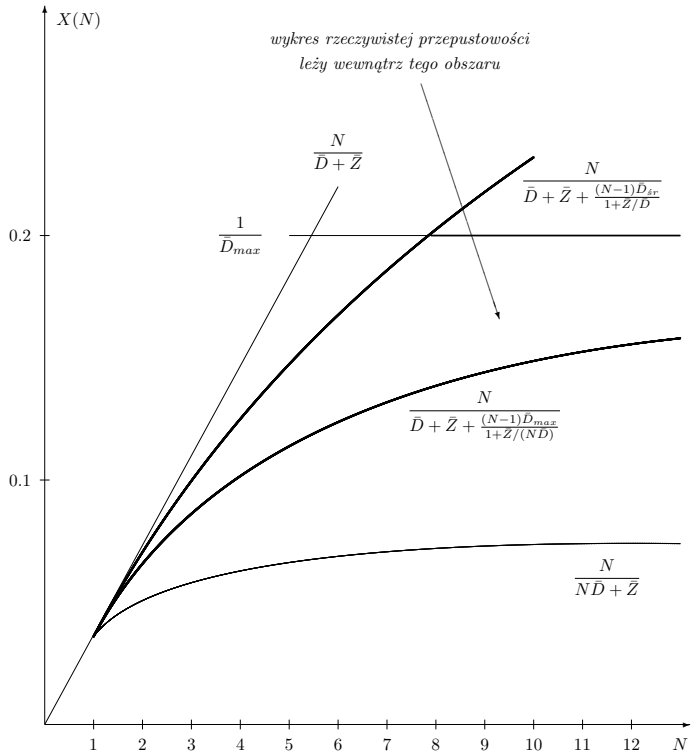


Figure 1.7: Terminal load – asymptotic bounds (thin lines) and balanced-system bounds (thick lines) for throughput; data from Exercise 1.5

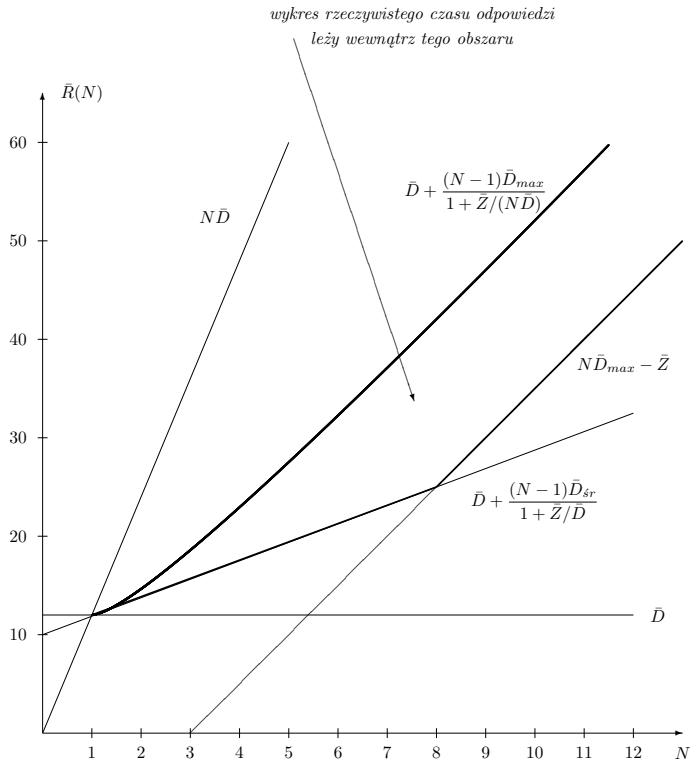


Figure 1.8: Terminal load – asymptotic bounds (thin lines) and balanced-system bounds (thick lines) for response time; data from Exercise 1.5

Let $\bar{D}_{max}$ denote the total service demand of the bottleneck station in the original system. In system B, we assign this amount of work to

$$M' = \frac{\bar{D}}{\bar{D}_{max}}.$$

It does not matter that M' may not be an integer. In this way, the total amount of work is distributed among only M' stations, while the remaining stations $i = M'+1, \ldots, M$ are idle.

We may interpret system B in two ways. On the one hand, if we ignore the idle stations and consider only the first M' stations, the system is balanced and we may apply formulas (1.7) and (1.8). On the other hand, if we consider all M stations, then the system is more unbalanced than the original one, and therefore its performance should be worse.

Thus we obtain the bounds

$$\begin{aligned} \frac{N}{N+M'-1}\frac{1}{\bar{D}_{max}} &\le X(N) \le \frac{N}{N+M-1}\frac{1}{\bar{D}_{ave}}\,, \\ \bar{D}+(N-1)\bar{D}_{max} &\ge \bar{R}(N) \ge \bar{D}+(N-1)\bar{D}_{ave}\,, \end{aligned} \tag{1.9}$$

and, recalling that $X(N) \le 1/\bar{D}_{max}$,

$$\begin{aligned} \frac{N}{\bar{D}+(N-1)\bar{D}_{max}} &\le X(N) \le \min\left\{\frac{1}{\bar{D}_{max}},\ \frac{N}{\bar{D}+(N-1)\bar{D}_{ave}}\right\}, \\ \bar{D}+(N-1)\bar{D}_{max} &\ge \bar{R}(N) \ge \max\left\{\bar{D},\ \bar{D}+(N-1)\bar{D}_{ave}\right\}. \end{aligned} \tag{1.10}$$

In the case of terminal load ($\bar{Z} > 0$), Eqs. (1.10) take the following form; see [30, 53]:

$$\begin{aligned} \frac{N}{\bar{D}+\bar{Z}+\frac{(N-1)\bar{D}_{max}}{1+\bar{Z}/(N\bar{D})}} &\le X(N) \le \min\left\{\frac{1}{\bar{D}_{max}},\ \frac{N}{\bar{D}+\bar{Z}+\frac{(N-1)\bar{D}_{ave}}{1+\bar{Z}/\bar{D}}}\right\}, \\ \bar{D}+\frac{(N-1)\bar{D}_{max}}{1+\bar{Z}/\bar{D}} &\ge \bar{R}(N) \ge \max\left\{N\bar{D}_{max}-\bar{Z},\ \bar{D}+\frac{(N-1)\bar{D}_{ave}}{1+\bar{Z}/(N\bar{D})}\right\}. \end{aligned} \tag{1.11}$$

Example 1.5

Draw the asymptotic and balanced-system bounds for a system of the type shown in Fig. 1.3, composed of three stations with total service demands $\bar{D}_1 = 5$ sec, $\bar{D}_2 = 4$ sec, and $\bar{D}_3 = 3$ sec. Consider both terminal load ($\bar{Z} = 15$ sec) and batch load.

We define

$$\bar{D} = \bar{D}_1 + \bar{D}_2 + \bar{D}_3 = 5\,\text{sec} + 4\,\text{sec} + 3\,\text{sec} = 12\,\text{sec}, \qquad \bar{D}_{\max} = 5\,\text{sec}, \qquad \bar{D}_{ave} = 4\,\text{sec}.$$

The bounds (1.10) and (1.11) are shown in Figs. 1.5–1.8.

■

Example 1.6

An interactive system of the type shown in Fig. 1.3 is composed of a CPU (station 1), a group of fast disks (station 2), and a group of slow disks (station 3). During an observation period of 60 *minutes, the following measurements were taken:*

— *number of active terminals:* 30,

— *mean user thinking time:* 15 *s*,

— *number of completed transactions:* 2000,

— *number of accesses to fast disks:* 20000,

— *number of accesses to slow disks:* 16000,

— *CPU busy time:* 1000 *sec*,

— *fast-disk busy time:* 200 *sec*,

— *slow-disk busy time:* 1600 *sec*.

Evaluate, using asymptotic bounds, the following separate modifications of the system:

1. *all files are moved to the fast disks and the slow disks are removed,*
2. *the slow disks are replaced by new fast disks of the same type as the existing ones,*
3. *the CPU is replaced by a new one that is 2.5 times faster,*
4. *the load of the existing disks is balanced — some files are transferred from one disk group to the other so that their utilizations become equal,*
5. *all three previous options are combined: we have two balanced fast-disk groups and a faster CPU.*

Using the notation introduced earlier, we have

$$T = 60\,\text{min}, \qquad \bar{Z} = 15\,\text{sec}, \qquad S_1 = 1000\,\text{sec}, \qquad S_2 = 200\,\text{sec}, \qquad S_3 = 1600\,\text{sec},$$

$$C = 2000, \qquad C_2 = 20000, \qquad C_3 = 16000.$$

The mean numbers of visits to the disks are

$$\begin{aligned} V_2 &= \frac{C_2}{C} = \frac{20000}{2000} = 10, \\ V_3 &= \frac{C_3}{C} = \frac{16000}{2000} = 8. \end{aligned}$$

The topology of the model implies that

$$V_1 = V_2 + V_3 + 1 = 19.$$

The mean total service demand at station i is $\bar{D}_i = S_i/C$:

$$\begin{aligned} \bar{D}_1 &= \frac{S_1}{C} = \frac{1000\,\text{s}}{2000} = 0.5\,\text{s}, \\ \bar{D}_2 &= \frac{S_2}{C} = \frac{200\,\text{s}}{2000} = 0.1\,\text{s}, \\ \bar{D}_3 &= \frac{S_3}{C} = \frac{1600\,\text{s}}{2000} = 0.8\,\text{s}. \end{aligned}$$

Modification	$\bar{D}_1$	$\bar{D}_2$	$\bar{D}_3$	$\bar{D}_{\max}$	$\bar{D}$	$1/\bar{D}_{\max}$	$1/(\bar{D}+\bar{Z})$
0	0.5	0.1	0.8	0.8	1.40	1.25	0.0606
1	0.5	0.18	0	0.5	0.68	2	0.0638
2	0.5	0.1	0.08	0.5	0.68	2	0.0638
3	0.2	0.1	0.8	0.8	1.1	1.25	0.0621
4	0.5	0.164	0.164	0.5	0.828	2	0.0632
5	0.2	0.09	0.09	0.2	0.38	5	0.0650

Table 1.1: System parameters for each modification

The mean service time per visit is

$$\bar{B}_i = \frac{\bar{D}_i}{V_i},$$

hence

$$\begin{aligned}
\bar{B}_1 &= \frac{\bar{D}_1}{V_1} = \frac{0.5\,\mathrm{s}}{19} = 0.026\,\mathrm{s}, \\
\bar{B}_2 &= \frac{\bar{D}_2}{V_2} = \frac{0.1\,\mathrm{s}}{10} = 0.01\,\mathrm{s}, \\
\bar{B}_3 &= \frac{\bar{D}_3}{V_3} = \frac{0.8\,\mathrm{s}}{8} = 0.1\,\mathrm{s}.
\end{aligned}$$

To plot the asymptotic bounds, we also need

$$\frac{1}{\bar{D}_{\max}} = 1.25\,\frac{1}{\mathrm{sec}}, \qquad N^* = \frac{\bar{D}+\bar{Z}}{\bar{D}_{\max}} = 20.5.$$

Let us now consider the modifications. Parameters corresponding to a modification are marked by a superscript indicating the number of that modification in square brackets. The original system is denoted as modification zero.

For clarity, let us show only selected asymptotic bounds in the figure (but for all system versions simultaneously): the upper bound for throughput and the lower bound for response time; see Fig. 1.9. As can be seen, modification [3], i.e. the introduction of a faster CPU, does not change the shape of the curves. The CPU is not the bottleneck, and speeding it up does not improve system performance.

Modifications [1], [2], and [4] yield practically the same result: they improve external-memory performance to such an extent that the CPU becomes the component limiting system performance, that is, the device determining throughput and response time. Note that modification [2] is the most expensive, since it requires the purchase of new disks, whereas modifications [1] and [4] require only the work of a system administrator. Modification [4] also has an advantage over [1], since it preserves more disk capacity.

Once disk performance has been improved, it also makes sense to speed up the CPU — this is modification [5].

■

Example 1.7

The virtual-memory system consists of:

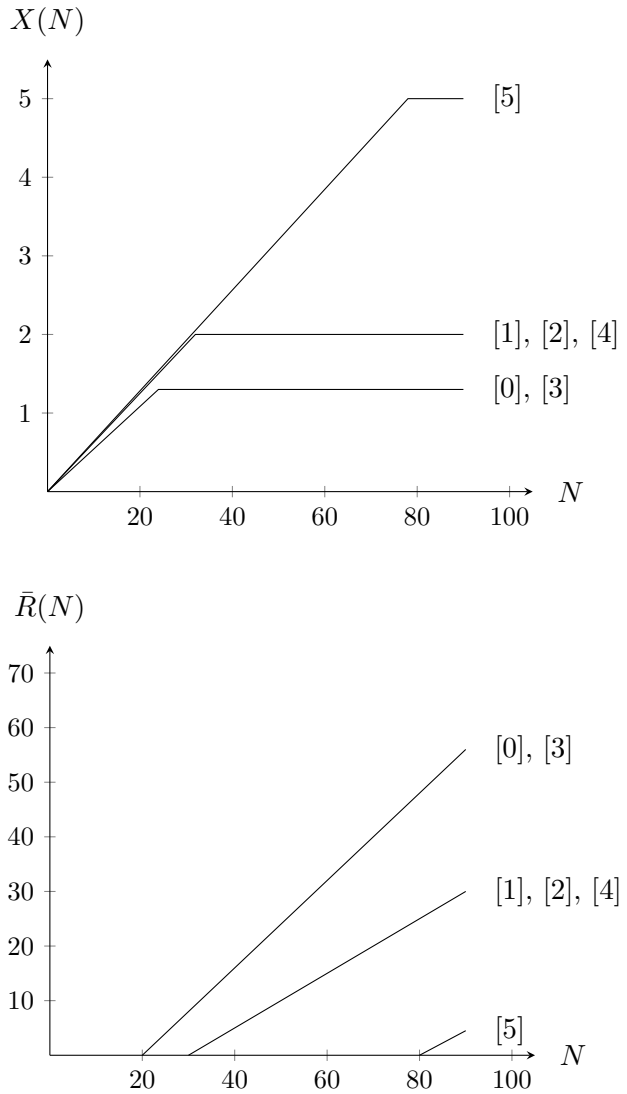


Figure 1.9: Asymptotic bounds for the system in Example 1.6 and its modification: an upper bound for $X(N)$ and a lower bound for $\bar{R}(N)$.

1. *a central processing unit,*

2. *a paging disk with fixed heads, used as an extension of main memory. Memory is divided into pages; some of them reside in main memory, while the others are stored on disk. If a program refers to a page that is not currently in main memory (a* page fault*), that page is fetched from disk and replaces another page that is no longer needed. A page fault causes the task to lose CPU time — in the model, the customer representing this task moves from the "central processing unit" station to the "paging disk" station and then returns to the end of the queue at the central processing unit,*

3. *a fixed-head disk, with the same parameters as the paging disk, used for writing files,*

4. *a moving-head disk containing files.*

Let us use the above numbers as the station numbers in the model. The service time for a single access to a fixed-head disk is $\bar{B}_2 = \bar{B}_3 = 12.5\,ms$*, while the service time for a single access to the moving-head disk is* $\bar{B}_4 = 30\ ms$*.*

Batch-type tasks are executed in the system. A typical task requires $\bar{D}_1 = 3\ s$ *of CPU time and performs, on average,* $V_3 = 150$ *accesses to files on the fixed-head disk and* $V_4 = 250$ *accesses to the moving-head disk.*

The mean time between page faults is a function of the level of multiprogramming: the more programs are running simultaneously, the smaller the amount of main memory allocated to each of them, and therefore the more frequent the page faults. The relationship between the multiprogramming level N *and the mean time between page faults, denoted by* $\bar{B}_{str}$*, has been measured and is given in the following table:*

N	1	2	3	4	5	6	7	8	9	10
$\bar{B}_{str}\,[ms]$	15.0	14.5	13.0	12.0	8.3	6.0	4.7	4.1	3.9	3.5

The utilization of the central processing unit was measured as a function of the multiprogramming level, yielding a dependence typical of a virtual-memory system: initially, CPU utilization increases with N*; subsequently, as page faults occur more and more frequently, the CPU spends an increasing fraction of its time manipulating memory pages, and its effective utilization by user programs decreases.*

N	1	2	3	4	5	6	7	8	9	10
U_1	0.286	0.302	0.350	0.375	0.370	0.361	0.329	0.310	0.291	0.275

- *For what multiprogramming level* N_{opt} *is the system throughput maximal, and what is its value?*

- *What software improvements to the system can be proposed, and how should the effect of these changes be evaluated?*

The system throughput is

$$X = \frac{X_1}{V_1} = \frac{U_1}{\bar{B}_1}\frac{1}{V_1} = \frac{U_1}{\bar{D}_1}\,,$$

and is therefore proportional to CPU utilization. Hence the optimum is attained at $N_{opt} = 4$, and

$$X_{\max} = \frac{U_{1,\max}}{\bar{D}_1} = \frac{0.375}{3} = 0.125\ \text{tasks/sec.}$$

To determine the number of file accesses per task, i.e. the numbers of customer visits to stations 3 and 4, we compute the total service demands:

$$\begin{aligned} \bar{D}_3 &= V_3 \bar{B}_3 = 150 \times 12.5\,\text{ms} = 1.875\,\text{s}, \\ \bar{D}_4 &= V_4 \bar{B}_4 = 250 \times 30\,\text{ms} = 7.5\,\text{s}. \end{aligned}$$

The number of page-fault operations required to complete a single task is a function of N:

$$V_2(N) = \frac{\bar{D}_1}{\bar{B}_{str}(N)},$$

and thus for $N = 4$ the total task processing time at station 2 is

$$\bar{D}_2(4) = V_2(4)\,\bar{B}_2 = \frac{\bar{D}_1}{\bar{B}_{str}(4)}\,\bar{B}_2 = \frac{3\,\text{s}}{0.012\,\text{s}} \times 0.0125\,\text{s} = 250 \times 0.0125\,\text{s} = 3.125\,\text{s}.$$

Comparing the values of $\bar{D}_i$, $i = 1, \ldots, 4$, we see that the bottleneck of the system is station 4. We may therefore attempt to balance the load between stations 3 and 4, as in the previous example. We transfer part of the files from the slow moving-head disk to the faster fixed-head disk so that the new values of the total service demands become equal:

$$\bar{D}_3^{[1]} = \bar{D}_4^{[1]}.$$

We compute the numbers of visits corresponding to this condition:

$$\begin{aligned} V_3^{[1]} + V_4^{[1]} &= V_3^{[0]} + V_4^{[0]}, \\ V_3^{[1]} \bar{B}_3 &= V_4^{[1]} \bar{B}_4, \end{aligned}$$

that is,

$$\begin{aligned} V_3^{[1]} + V_4^{[1]} &= 400, \\ V_3^{[1]} \times 12.5 &= V_4^{[1]} \times 30. \end{aligned}$$

Hence we obtain

$$V_3^{[1]} \approx 282, \qquad V_4^{[1]} \approx 118,$$

and therefore

$$\bar{D}_3^{[1]} = \bar{D}_4^{[1]} \approx 3.54\,\text{s}.$$

The system is now nearly balanced, and the total service demand is

$$\bar{D} = \sum_{i=1}^{4} \bar{D}_i = 3\,\text{s} + 3.125\,\text{s} + 3.54\,\text{s} + 3.54\,\text{s} = 13.16\,\text{s}.$$

We obtain

$$\bar{D}_{av} = 3.29\,\text{s}, \qquad \bar{D}_{\max} = 3.54\,\text{s},$$

and, using the balanced-system bounds,

$$\frac{N}{\bar{D} + (N-1)\bar{D}_{\max}} \;\le\; X(N) \;\le\; \frac{N}{\bar{D} + (N-1)\bar{D}_{av}},$$

we estimate the new system throughput:

$$\begin{aligned} \frac{4}{13.16 + (4-1) \times 3.54} &\leq X(4) \leq \frac{4}{13.16 + (4-1) \times 3.29}, \\ 0.168 &\leq X(4) \leq 0.174. \end{aligned}$$

We may therefore expect the system throughput to increase from 0.125 tasks/sec to approximately 0.170 tasks/sec, i.e. by about 36%.

■

Chapter 2

Mean Value Analysis

2.1 Basic algorithm

The bounds discussed so far allow us to estimate the response time and throughput of the system. But where exactly between $\bar{R}_{\min}$ and $\bar{R}_{\max}$, and $X_{\min}$ and $X_{\max}$, do the actual values $\bar{R}$ and X lie? This question is answered by the analysis of mean values presented in this chapter, known briefly as the *MVA* method (*Mean Value Analysis*), [100, 102, 101].

In an open network such as that shown in Fig. 2.1, which is in equilibrium (the network is in equilibrium if all its stations are in equilibrium, i.e. if the mean service time is shorter than the mean interarrival time of successive jobs), the throughput of any station i is determined by the external flow X passing through the network and by the number of visits V_i made by customers to that station:

$$X_i = XV_i.$$

The throughput X_i does not, however, depend on the queue length or the waiting time at that station. Let the average number of customers at station i be $\bar{N}_i$. A new customer arriving at station i sees $\bar{N}_i$ customers ahead of it; each of them will be served for a time $\bar{B}_i$ before service of the observed customer begins. The station response time, consisting of the waiting time and the service time, is therefore

$$\bar{R}_i = \bar{N}_i\bar{B}_i + \bar{B}_i = \bar{B}_i(\bar{N}_i + 1)\,.$$

We also know, from Little's law, that $\bar{N}_i = \bar{R}_i X_i$. Combining the above equations and recalling that $\bar{B}_i X_i = U_i$, where U_i is the utilization of station i, we obtain

$$\bar{N}_i = \frac{U_i}{1 - U_i} \tag{2.1}$$

and

$$\bar{R}_i = \frac{\bar{B}_i}{1 - U_i}\,. \tag{2.2}$$

The reasoning above is approximate: one of the $\bar{N}_i$ customers present at the station when a new customer arrives is already being served, and the average remaining service time generally differs from $\bar{B}_i$. We shall return to this issue in the probabilistic section; for the moment, let us only note that results (2.1) and (2.2) are exact when the service-time distribution is exponential. A condition for the existence of a solution is that for all stations, $U_i < 1$. This is evident from the form of expressions (2.1) and (2.2). Otherwise, the system is unstable: the queue of tasks

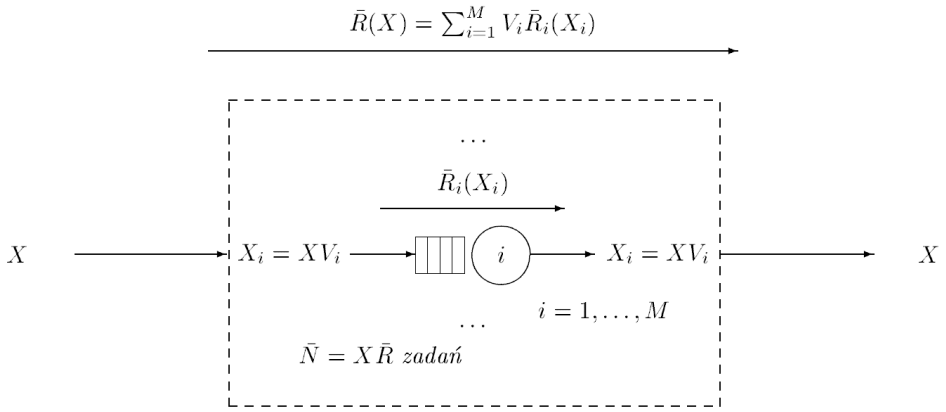


Figure 2.1: Open network with M stations.

in front of the station whose utilization factor satisfies $U_i = 1$ keeps increasing, because the station is unable to process tasks sufficiently quickly.

Knowing the external customer flow X and the numbers of visits to the stations, we calculate the values $\bar{N}_i$, and then the average number of customers in the entire network,

$$\bar{N} = \sum_{i=1}^{M} \bar{N}_i,$$

and, in accordance with Little's law, the network response time

$$\bar{R} = \bar{N}/X.$$

We can also calculate $\bar{R}$ by first determining the mean total time

$$\bar{R}_{(c)i} = V_i \bar{R}_i$$

that a customer spends at station i:

$$\bar{R}_{(c)i} = V_i \frac{\bar{B}_i}{1 - U_i} = \frac{\bar{D}_i}{1 - U_i},$$

and then summing

$$\bar{R} = \sum_{i=1}^{M} \bar{R}_{(c)i}.$$

Example 2.1

Fig. 2.2 *shows a model of a transaction-processing system. The transaction stream originates at many terminals and is practically independent of the number of transactions already present and being processed in the system. The model distinguishes four service centers, representing, respectively, the central processing unit, fast disks, slow disks, and the communication processor.*

The average service times per visit at these stations are

$$\bar{B}_1 = 30\ ms, \qquad \bar{B}_2 = 20\ ms, \qquad \bar{B}_3 = 80\ ms, \qquad \bar{B}_4 = 500\ ms.$$

The average numbers of visits per task to the individual stations are:

$$V_1 = 40\,, \quad V_2 = 30\,, \quad V_3 = 10\,, \quad V_4 = 1\,.$$

The input task flow arriving at the communication processor has value $X = 0.7$ *tasks/sec. Determine the basic performance parameters of this system. At what threshold value* X_{gr} *is the system unable to handle all transactions and thus becomes unstable?*

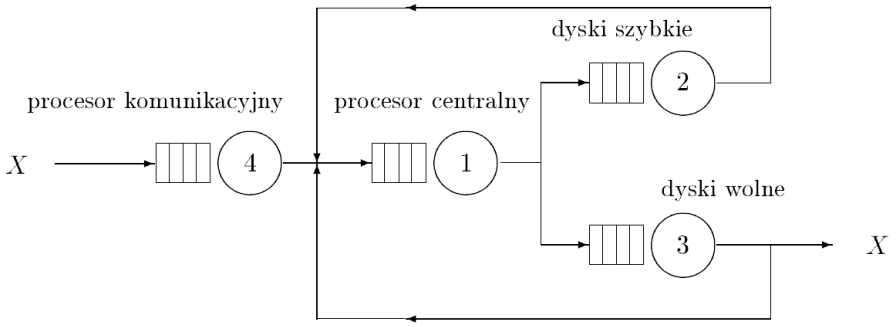


Figure 2.2: Model of a transaction-processing system, Example 2.1.

The performance parameters of the service centers and the method of their calculation are summarized in the following table:

Parameter		Station number:			
		1	2	3	4
$\bar{D}_i = V_i\,\bar{B}_i$	[s]	1.2	0.6	0.8	0.5
$X_i = V_i X$	[tasks/s]	28	21	7	0.7
$U_i = X_i\bar{B}_i = X\bar{D}_i$		0.84	0.42	0.56	0.35
$\bar{N}_i = U_i/(1-U_i)$	[tasks]	5.25	0.72	1.27	0.54
$\bar{R}_i = \bar{B}_i/(1-U_i)$	[s]	0.1875	0.0345	0.1818	0.7692
$\bar{R}_{(c)i} = \bar{D}_i/(1-U_i)$	[s]	7.5	1.0345	1.8181	0.7692

The network response time is

$$\bar{R} = \sum_{i=1}^{4} \bar{R}_{(c)i} = 7.5\ s + 1.0345\ s + 1.8181\ s + 0.7692\ s \approx 11.12\ s.$$

One can also determine $\bar{R}$ using Little's law:

$$\bar{R} = \frac{\bar{N}}{X} = \frac{\sum_{i=1}^{4} \bar{N}_i}{X} = \frac{5.25 + 0.72 + 1.27 + 0.54}{0.7\ 1/\mathrm{s}} = \frac{7.78}{0.7}\ \mathrm{s} \approx 11.12\ s.$$

It can be seen that the central processing unit is the bottleneck of the system. The total service demand of a task is greatest there; the largest part of the total response time is therefore due to the central processing unit. This is also the station at which the longest queue is found.

The stability of the system is determined by the stability of the most heavily loaded station, and the load on the stations depends linearly on the system throughput. The load on the central processing unit is currently, at $X = 0.7$ tasks/s, equal to 84%, and it will reach 100% at

$$X_{gr} = \frac{0.7}{0.84} = 0.833 \text{ tasks/s}.$$

■

Example 2.2

The path of packets from node 1 to node 4 is shown in Fig. 2.3. Node 1 sends packets to node 4 at a rate of $X = 4$ packets per unit time. The mean packet transmission time over each link is the same and equal to $\bar{B} = 0.2$ time units.

Determine the packet flow rates between the individual nodes and the average packet transmission time between nodes 1 and 4. The probabilities of selecting particular packet routes are as follows: $r_{1,2} = 0.25$, $r_{1,3} = 0.75$, $r_{2,3} = 0.5$, $r_{2,4} = 0.5$, $r_{3,4} = 1$. The intensity of local traffic (additional to the traffic under consideration) is shown in Fig. 2.3.

We calculate the traffic intensities between successive nodes, expressed in packets per unit time:

$$\begin{aligned}
X_{1-2} &= X r_{1-2} + X^{loc}_{1-2} = 4 \times 0.25 + 2 = 3, \\
X_{2-3} &= X r_{1-2}\, r_{2-3} + X^{loc}_{2-3} = 4 \times 0.25 \times 0.5 + 2 = 2.5, \\
X_{2-4} &= X r_{1-2}\, r_{2-4} + X^{loc}_{2-4} = 4 \times 0.25 \times 0.5 + 4 = 4.5, \\
X_{1-3} &= X r_{1-3} + X^{loc}_{1-3} = 4 \times 0.75 + 1 = 4, \\
X_{3-4} &= X r_{1-3}\, r_{3-4} + X^{loc}_{3-4} = 4 \times 0.75 \times 1.0 + 0.5 = 3.5.
\end{aligned}$$

Next, we calculate the utilizations of the links,

$$U_{i-j} = X_{i-j} \bar{B},$$

thus obtaining

$$U_{1-2} = 0.6, \qquad U_{2-3} = 0.5, \qquad U_{2-4} = 0.9, \qquad U_{1-3} = 0.8, \qquad U_{3-4} = 0.7.$$

The response times for the individual links are

$$\bar{R}_{i-j} = \frac{\bar{B}}{1 - U_{i-j}},$$

hence

$$\bar{R}_{1-2} = 0.5, \qquad \bar{R}_{2-3} = 0.4, \qquad \bar{R}_{2-4} = 2.0, \qquad \bar{R}_{1-3} = 1.0, \qquad \bar{R}_{3-4} = 0.667.$$

The probability that a packet takes the route $1 - 2 - 4$ is

$$p_{1-2-4} = r_{1-2} r_{2-4} = 0.125,$$

similarly,

$$p_{1-2-3-4} = 0.125, \qquad p_{1-3-4} = 0.75.$$

Taking this into account, we calculate the average response time for the connection between nodes 1 and 4:

$$\begin{aligned}\bar{R} &= p_{1-2-4}(\bar{R}_{1-2}+\bar{R}_{2-4}) \\ &\quad +p_{1-2-3-4}(\bar{R}_{1-2}+\bar{R}_{2-3}+\bar{R}_{3-4}) \\ &\quad +p_{1-3-4}(\bar{R}_{1-3}+\bar{R}_{3-4}) \\ &= 0.125(0.5+2.0)+0.125(0.5+0.4+0.667)+0.75(1.0+0.667) \\ &\approx 1.76.\end{aligned}$$

■

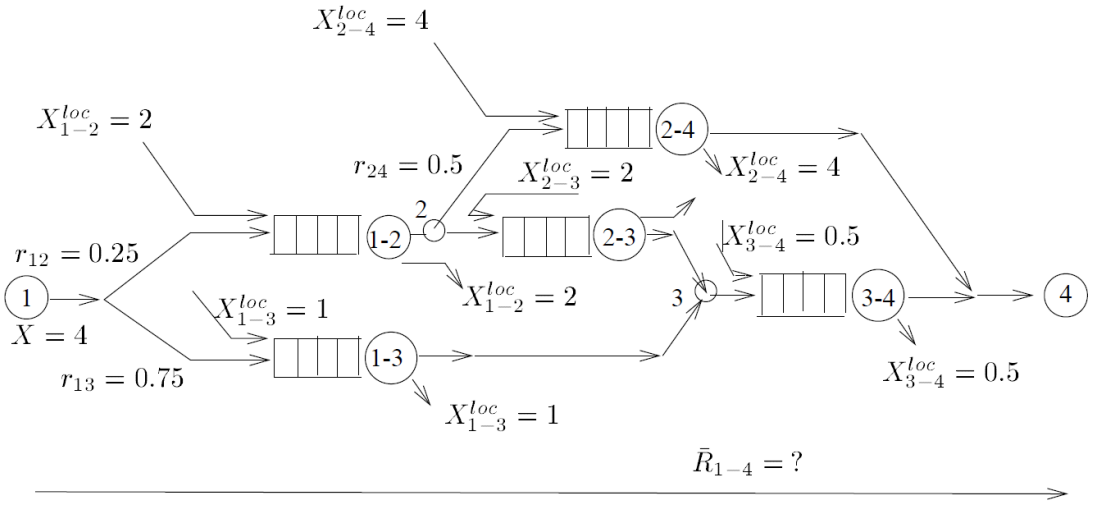


Figure 2.3: Packet flow through the network in Example 2.2

In a closed network, shown in Fig. 2.4, the average number of clients at a station (as well as its utilization and throughput) depends on the total number N of clients in the network,

$$\bar{N}_i = \bar{N}_i(N).$$

A client arriving at station i finds on average $\bar{N}_i(N-1)$ other clients there, that is, as many as would be present on average at the station if the client under consideration were removed from the network. In other words, when a new client arrives at station i, there are on average $\bar{N}_i(N-1)$ other customers already present.

Proofs of this fundamental relationship for the *MVA* algorithm are given in [110, 71]. The response time at station i is therefore

$$\bar{R}_i(N) = \bar{B}_i[1+\bar{N}_i(N-1)]\,, \qquad i=1,\ldots,M. \tag{2.3}$$

At stations representing terminals, where there is no queue and therefore no waiting time,

$$\bar{R}_i(N) = \bar{B}_i\,. \tag{2.4}$$

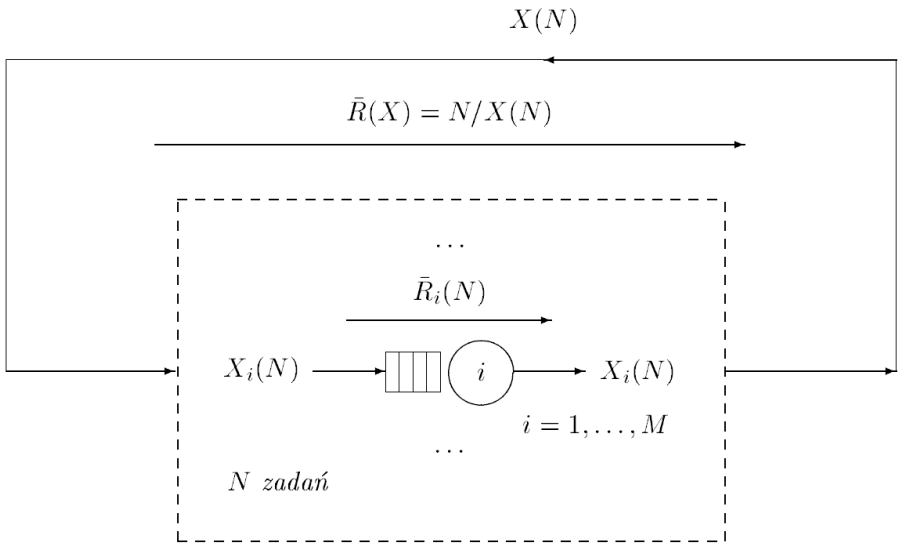


Figure 2.4: Closed network of M stations

Knowing $\bar{R}_i(N)$ for $i = 1, \ldots, M$, we can in turn calculate the network throughput. We choose a branch whose throughput is regarded as the network throughput $X(N)$ and, using Little's law applied to the entire network as viewed from this branch, we obtain

$$X(N) = \frac{N}{\sum_{i=1}^{M} V_i \bar{R}_i(N)} = \frac{N}{\sum_{i=1}^{M} \bar{R}_{(c)i}(N)}\,, \tag{2.5}$$

where V_i is the average number of visits to station i per single cycle through the branch with respect to which the network throughput is defined, and

$$\bar{R}_{(c)i}(N) = V_i \bar{R}_i(N)$$

is the total time spent at station i between successive visits to the selected branch.

If station i has a queue, then

$$\bar{R}_{(c)i}(N) = \bar{D}_i[1 + \bar{N}_i(N-1)],$$

whereas if it is a station where tasks do not queue, then

$$\bar{R}_{(c)i}(N) = \bar{D}_i.$$

The throughput of station i is V_i times greater than the network throughput:

$$X_i(N) = V_i X(N).$$

Let us now apply Little's law separately to each station:

$$\bar{N}_i(N) = X_i(N)\bar{R}_i(N)\,, \qquad i = 1, \ldots, M. \tag{2.6}$$

In this way, knowing the value of $\bar{N}_i(N-1)$, one can calculate the value of $\bar{N}_i(N)$. The utilization of station i is

$$U_i(N) = \bar{B}_i X_i(N)\,.$$

The Mean Value Analysis algorithm consists of recursively calculating expressions (2.3)–(2.6), assuming that there are $n = 1, 2, \ldots, N$ customers in the network. The initial condition is

$$\bar{N}_i(0) = 0, \qquad i = 1, \ldots, M,$$

and consequently

$$\bar{R}_i(1) = \bar{B}_i,$$

since if there is only one customer in the network, that customer always arrives at an empty station.

Equation (2.3), which relates response time to the average number of customers, applies not only to a station with the standard queueing discipline, in which the order of service corresponds to the order of arrival of requests, also known as the *FIFO* (*First-In-First-Out*) or *FCFS* (*First-Come-First-Served*) discipline, but also to a *PS* (*Processor Sharing*) station, in which the average service time is proportional to the number of clients at the station, as well as to a *LIFO* (*Last-In-First-Out*) station with the following queue discipline: service of a newly arriving client begins immediately, interrupting the current service, which is resumed when the station becomes available again.

We shall justify this in more detail in Chapter 5, which describes the probabilistic model of a similarly defined network.

Stations for which equation (2.4) applies, i.e. stations whose response time is always equal to the service time, are called *IS* (*Infinite Server*) stations. This means that we may assume an infinite number of parallel service channels, so that service of every customer begins immediately upon arrival, regardless of how many customers are already present at the station.

Example 2.3

Let us return to the model for which the asymptotic and balanced-system bounds on throughput and response time were computed in Example 1.5. The model consists of a group of terminals, a central processing unit, and two disks, and its structure is shown in Fig. 1.3. We assume terminal load ($\bar{Z} = 15$ s), and the total task service demands at the stations are: $\bar{D}_1 = 5$ s, $\bar{D}_2 = 4$ s, $\bar{D}_3 = 3$ s. Calculate the mean response time and system throughput for $N \in [1, 12]$ active terminals.

For our model, the equations of the *MVA* method take the form

$$\begin{aligned} \bar{R}_{(c)i}(N) &= \bar{D}_i[1 + \bar{N}_i(N-1)]\,, \qquad i = 1, 2, 3, \\ X(N) &= \frac{N}{\bar{Z} + \sum_{i=1}^{3} \bar{R}_{(c)i}(N)}\,, \\ \bar{N}_i(N) &= X(N)\bar{R}_{(c)i}(N)\,, \qquad i = 1, 2, 3. \end{aligned}$$

We use these equations to perform calculations for $N = 1, 2, \ldots, 12$. Once the throughput $X(N)$ has been found for a given N, the utilizations of the stations are obtained from

$$U_i(N) = X(N)\bar{D}_i.$$

On average, there are

$$\bar{N}_0(N) = X(N)\bar{Z}$$

tasks at the terminals. The system response time

$$\bar{R}(N) = \sum_{i=1}^{3} \bar{R}_{(c)i}(N)$$

can also be calculated as

$$\bar{R}(N) = \frac{\sum_{i=1}^{3} \bar{N}_i(N)}{X(N)} = \frac{N - \bar{N}_0(N)}{X(N)} \,.$$

- $N = 1$:

$$\begin{aligned}
\bar{R}_{(c)1}(1) &= \bar{D}_1[1 + \bar{N}_1(0)] = \bar{D}_1 = 5 \,, \\
\bar{R}_{(c)2}(1) &= \bar{D}_2[1 + \bar{N}_2(0)] = \bar{D}_2 = 4 \,, \\
\bar{R}_{(c)3}(1) &= \bar{D}_3[1 + \bar{N}_3(0)] = \bar{D}_3 = 3 \,, \\
\bar{R}(1) &= \sum_{i=1}^{3} \bar{R}_{(c)i}(1) = 5 + 4 + 3 = 12 \,, \\
X(1) &= \frac{1}{\bar{Z} + \bar{R}(1)} = \frac{1}{15 + 12} = \frac{1}{27} = 0.0370 \,, \\
\bar{N}_1(1) &= X(1)\bar{R}_{(c)1}(1) = \frac{5}{27} = 0.1852 \,, \\
\bar{N}_2(1) &= X(1)\bar{R}_{(c)2}(1) = \frac{4}{27} = 0.1481 \,, \\
\bar{N}_3(1) &= X(1)\bar{R}_{(c)3}(1) = \frac{3}{27} = 0.1111 \,.
\end{aligned}$$

The average number of tasks at the terminals is

$$\bar{N}_0(1) = X(1)\bar{Z} = \frac{15}{27} = 0.5556.$$

For $N = 1$, the station utilizations

$$U_i(1) = X(1)\bar{D}_i$$

are equal to the mean numbers of customers at these stations, that is, to the proportions of time during which the single customer in the network is present at the corresponding stations:

$$U_i(1) = \bar{N}_i(1).$$

- $N = 2$:

$$\begin{aligned}
\bar{R}_{(c)1}(2) &= \bar{D}_1[1 + \bar{N}_1(1)] = 5[1 + 0.1852] = 5.9259\,, \\
\bar{R}_{(c)2}(2) &= \bar{D}_2[1 + \bar{N}_2(1)] = 4[1 + 0.1481] = 4.5926\,, \\
\bar{R}_{(c)3}(2) &= \bar{D}_3[1 + \bar{N}_3(1)] = 3[1 + 0.1111] = 3.3333\,, \\
\bar{R}(2) &= \sum_{i=1}^{3} \bar{R}_{(c)i}(2) = 5.9259 + 4.5926 + 3.3333 = 13.8518\,, \\
X(2) &= \frac{2}{\bar{Z} + \bar{R}(2)} = \frac{2}{15 + 13.8518} = 0.0693\,, \\
\bar{N}_0(2) &= X(2)\bar{Z} = 0.0693 \times 15 = 1.0395\,, \\
\bar{N}_1(2) &= X(2)\bar{R}_{(c)1}(2) = 0.0693 \times 5.9259 = 0.4108\,, \\
\bar{N}_2(2) &= X(2)\bar{R}_{(c)2}(2) = 0.0693 \times 4.5926 = 0.3184\,, \\
\bar{N}_3(2) &= X(2)\bar{R}_{(c)3}(2) = 0.0693 \times 3.3333 = 0.2311\,.
\end{aligned}$$

  The station utilizations are

$$\begin{aligned}
U_1(2) &= X(2)\bar{D}_1 = 0.0693 \times 5 = 0.3466\,, \\
U_2(2) &= X(2)\bar{D}_2 = 0.0693 \times 4 = 0.2773\,, \\
U_3(2) &= X(2)\bar{D}_3 = 0.0693 \times 3 = 0.2080\,.
\end{aligned}$$

- $N = 3$:

$$\begin{aligned}
\bar{R}_{(c)1}(3) &= \bar{D}_1[1 + \bar{N}_1(2)] = 5[1 + 0.4108] = 7.0539\,, \\
\bar{R}_{(c)2}(3) &= \bar{D}_2[1 + \bar{N}_2(2)] = 4[1 + 0.3184] = 5.2736\,, \\
\bar{R}_{(c)3}(3) &= \bar{D}_3[1 + \bar{N}_3(2)] = 3[1 + 0.2311] = 3.6933\,, \\
\bar{R}(3) &= \sum_{i=1}^{3} \bar{R}_{(c)i}(3) = 7.0539 + 5.2736 + 3.6933 = 16.0208\,, \\
X(3) &= \frac{3}{\bar{Z} + \sum_{i=1}^{3} \bar{R}_{(c)i}(3)} = \frac{3}{15 + 16.0208} = 0.0967\,, \\
\bar{N}_0(3) &= X(3)\bar{Z} = 0.0967 \times 15 = 1.4505\,, \\
\bar{N}_1(3) &= X(3)\bar{R}_{(c)1}(3) = 0.0967 \times 7.0539 = 0.6822\,, \\
\bar{N}_2(3) &= X(3)\bar{R}_{(c)2}(3) = 0.0967 \times 5.2736 = 0.5100\,, \\
\bar{N}_3(3) &= X(3)\bar{R}_{(c)3}(3) = 0.0967 \times 3.6933 = 0.3572\,.
\end{aligned}$$

  The station utilizations are

$$\begin{aligned}
U_1(3) &= X(3)\bar{D}_1 = 0.0967 \times 5 = 0.4835\,, \\
U_2(3) &= X(3)\bar{D}_2 = 0.0967 \times 4 = 0.3868\,, \\
U_3(3) &= X(3)\bar{D}_3 = 0.0967 \times 3 = 0.2901\,.
\end{aligned}$$

We proceed similarly for $N = 4, \ldots, 12$. Table 2.1 contains some of the calculated results.

Figs. 2.5 and 2.6 show the graphs of $X(N)$ and $\bar{R}(N)$ against the background of the asymptotic and balanced-system bounds.

N	$\bar{R}$	X	$\bar{N}_1$	$\bar{N}_2$	$\bar{N}_3$	U_1	U_2	U_3
1	12.0000	0.0370	0.1852	0.1482	0.1111	0.1852	0.1482	0.1111
2	13.8517	0.0693	0.4108	0.3184	0.2311	0.3466	0.2773	0.2080
3	16.0205	0.0967	0.6822	0.5100	0.3572	0.4836	0.3868	0.2901
4	18.5233	0.1193	1.0036	0.7207	0.4858	0.5966	0.4773	0.3580
5	21.3583	0.1375	1.3777	0.9465	0.6130	0.6876	0.5501	0.4126
6	24.5125	0.1519	1.8052	1.1823	0.7348	0.7592	0.6074	0.4555
7	27.9600	0.1629	2.2855	1.4224	0.8480	0.8147	0.6518	0.4888
8	31.6605	0.1715	2.8165	1.6613	0.9505	0.8573	0.6858	0.5144
9	35.5788	0.1779	3.3955	1.8942	1.0412	0.8897	0.7118	0.5338
10	39.6779	0.1829	4.0195	2.1173	1.1200	0.9145	0.7316	0.5487
11	43.9273	0.1867	4.6850	2.3276	1.1872	0.9337	0.7467	0.5600
12	48.2978	0.1896	5.3889	2.5234	1.2440	0.9479	0.7583	0.5687

Table 2.1: Calculation results for Example 2.2

■

When deriving the balanced-system bounds for batch load, we assumed without proof the following expression for the station utilization:

$$U_i(N) = \frac{N}{M+N-1}\,,$$

where, as usual, N is the number of tasks and M is the number of stations. We can now derive this relation easily.

In a balanced system,

$$\bar{D}_1 = \bar{D}_2 = \cdots = \bar{D}_M = \frac{\bar{D}}{M},$$

and the average number of tasks at each station is the same:

$$\bar{N}_1(N) = \bar{N}_2(N) = \cdots = \bar{N}_M(N) = \frac{N}{M}.$$

We determine the total response time at a station, which is the same for all stations:

$$\bar{R}_{(c)i}(N) = \bar{D}_i[1 + \bar{N}_i(N-1)] = \frac{\bar{D}}{M}\left[1 + \frac{N-1}{M}\right]\,,$$

and the network throughput (we are considering batch load here, so $\bar{Z} = 0$):

$$X(N) = \frac{N}{\sum_{i=1}^{M} \bar{R}_{(c)i}(N)} = \frac{N}{M \cdot \frac{\bar{D}}{M}\left[1 + \frac{N-1}{M}\right]} = \frac{NM}{\bar{D}(M+N-1)}\,.$$

The utilization of a station is therefore

$$U_i(N) = X(N)\bar{D}_i = \frac{NM}{\bar{D}(M+N-1)} \cdot \frac{\bar{D}}{M} = \frac{N}{N+M-1}\,.$$

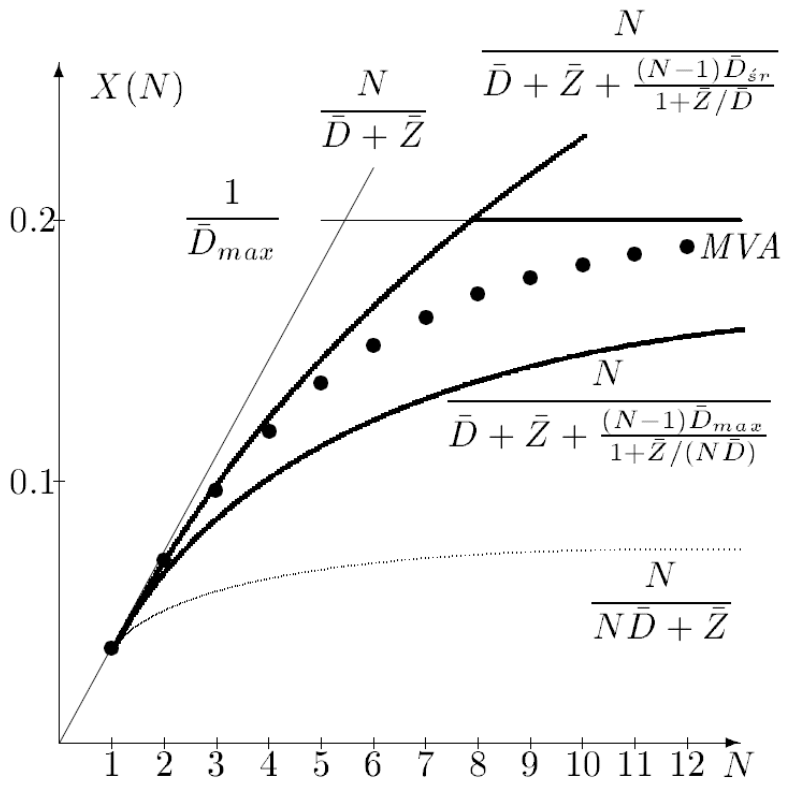


Figure 2.5: Throughput $X(N)$ calculated using *MVA* (dots) and its asymptotic and balanced-system estimates, data from Example 2.2

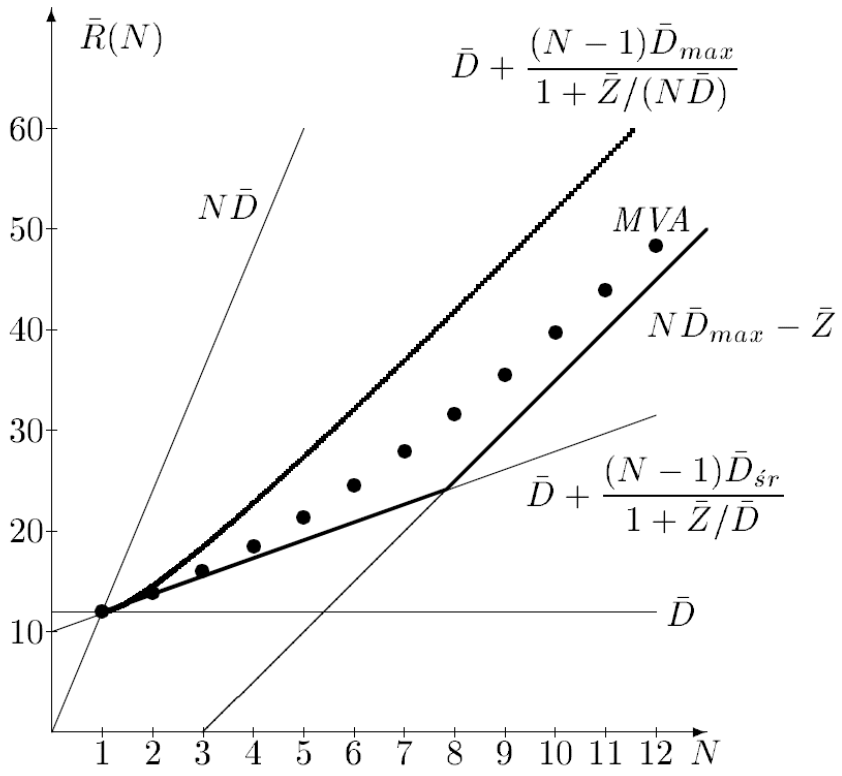


Figure 2.6: Response time $\bar{R}(N)$ calculated using *MVA* (dots) and its asymptotic and balanced-system estimates, data from Example 2.2

Example 2.4

Figure 2.6 shows a section of a computer network through which packets are transmitted (switched). The black dots denote the nodes of this network, and the arrows between them denote packet-transmission paths. At this point, we are not concerned with the operation of the entire network, but only with packet transmission along the path A–B–C. This path, considered in isolation from the rest of the network, is called a virtual path. Its model is shown in Fig. 2.8a.

Packets arrive at node A at a rate of X_A packets per unit time and form a queue there, waiting to be processed, i.e. transmitted to the next node. Transmission over link A–B corresponds to the first service station, and transmission over link B–C corresponds to the second station. We want to determine the transmission time over path A–B–C, the queue lengths at the nodes, and the actual path throughput.

In the model, we must take into account the transmission-window mechanism: at most N packets may be present on the A–B–C path at any given moment. If this limit is reached, subsequent packets arriving at node A are discarded. How can this be represented in the model? We transform the model from Fig. 2.8a into a closed network containing an additional, third station with mean service time $\bar{B}_3 = 1/X_A$; see Fig. 2.8b. There are N customers circulating in this network. As long as fewer than N customers are present at stations 1 and 2 together, station 3 is active and customers move from it to station 1 at rate X_A. If the total number of customers in stations 1 and 2 reaches N, then station 3 is empty and no new customers may enter. The model therefore respects the constraints imposed by the windowing mechanism. It remains to solve the closed-network model using the MVA algorithm.

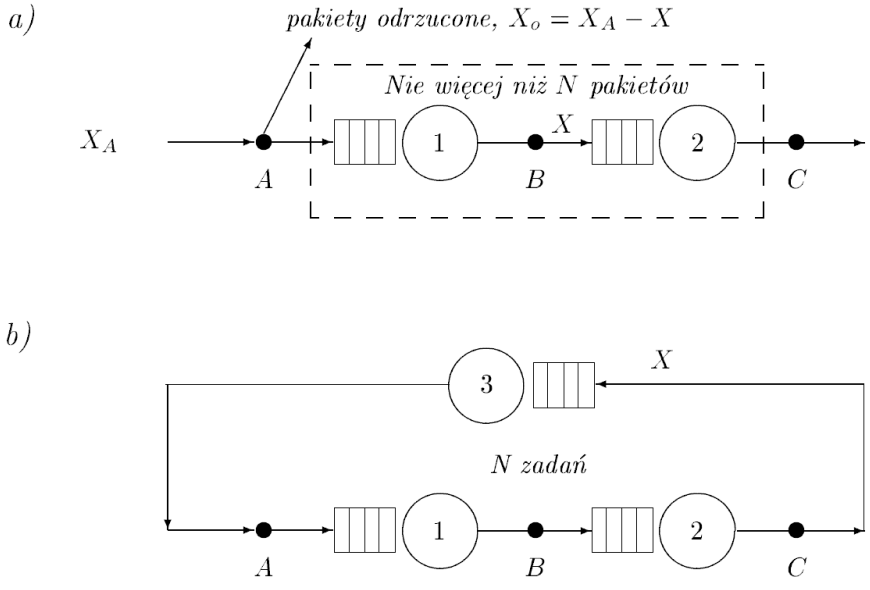


Figure 2.7: Packet transmission in Example 2.4

Let us assume that the transmission times over links A–B and B–C are equal:

$$\bar{B}_1 = \bar{B}_2.$$

In the model, $V_1 = V_2 = V_3 = 1$, and the *MVA* equations for any N take the form:

$$\begin{aligned}
\bar{R}_1(N) &= \bar{R}_2(N) = \bar{B}_1[1 + \bar{N}_1(N-1)]\,, \\
\bar{R}_3(N) &= \frac{1}{X_A}[1 + \bar{N}_3(N-1)]\,, \\
X(N) &= \frac{N}{\sum_{i=1}^{3} \bar{R}_i(N)}\,, \\
\bar{N}_1(N) &= \bar{N}_2(N) = X(N)\bar{R}_1(N)\,, \\
\bar{N}_3(N) &= X(N)\bar{R}_3(N)\,.
\end{aligned}$$

The link utilization is

$$U(N) = U_1(N) = U_2(N) = X(N)\bar{B}_1.$$

If the windowing mechanism is not active, i.e. no packets are discarded ($X = X_A$), then the open-network model consisting of two service stations in series gives

$$U_1 = U_2 = X_A\bar{B}_1, \qquad \bar{N}_1 = \bar{N}_2 = \frac{U_1}{1-U_1}, \qquad \bar{R}_1 = \bar{R}_2 = \frac{\bar{B}_1}{1-U_1}\,.$$

Let us assume

$$\bar{B}_1 = 0.15\,\text{s}, \qquad X_A = 3\,\text{packets/s}.$$

Table 2.2 contains the values

$$\bar{N} = \bar{N}_1 + \bar{N}_2$$

of the mean number of packets on route A–C, the mean transmission time

$$\bar{R} = \bar{R}_1 + \bar{R}_2,$$

the throughput X, and the link utilization

$$U = U_2 = U_1$$

as functions of the window size N, as well as the case in which the windowing mechanism is disabled ($N \to \infty$). The smaller the window size, the more frequently packets are discarded and, consequently, the lower the utilization and throughput of the links. At the same time, the transmission time becomes shorter.

■

Example 2.5

The virtual path is represented in Fig. **??** *by a sequence of M identical service stations. The windowing mechanism limits the number of packets in transit to N. Determine the value of N for which the link capacity, defined as the ratio of throughput to response time, is maximized.*

We express the path throughput, in accordance with Little's law, as

$$X(N) = \frac{N}{\bar{R}(N)}.$$

N	$\bar{R}$	$\bar{N}$	X	U
2	0.3711	0.8435	2.2732	0.3410
4	0.4676	1.3093	2.8001	0.4200
6	0.5156	1.5191	2.9463	0.4419
8	0.5354	1.5987	2.9862	0.4479
10	0.5424	1.6252	2.9966	0.4495
∞	0.5455	1.6364	3.0000	0.4500

Table 2.2: Calculation results for Example 2.3

Since we assume that all service stations have identical properties and that the transmission times over all links are the same,

$$\bar{B}_i = \bar{B}, \qquad i = 1, \ldots, M,$$

the average number of tasks seen at each station when a new task arrives is

$$\bar{N}_i = \frac{N-1}{M}.$$

Hence the response time for the entire path is

$$\bar{R}(N) = M\bar{R}_i(N) = M\bar{B}\left[1 + \frac{N-1}{M}\right] = \bar{B}[M+N-1].$$

We now calculate the link capacity:

$$q(N) = \frac{X(N)}{\bar{R}(N)} = \frac{N}{\bar{R}^2(N)} = \frac{N}{\bar{B}^2[M+N-1]^2}.$$

From the condition

$$\frac{d\,q(N)}{d\,N} = 0$$

we obtain

$$N = M - 1.$$

■

2.2 Multiple classes of clients

So far we have assumed that all clients behave in the same way: they have the same mean service times and the same mean numbers of visits to the stations. This assumption is not always justified. The utilization of the central processing unit and the number of disk accesses differ for numerical computations, text-editing sessions, database queries, and other kinds of workloads. If the tasks performed in the system are not all of a similar nature, it is advisable to divide them into classes within which their behavior may be regarded as uniform.

In such a model, we distinguish classes of clients, each of which is characterized by its own service times and its own route through the network of stations.

Assume that there are K classes of clients in the model. A quantity characteristic of class k will be denoted by a superscript in parentheses (to avoid confusion with an exponent). For example, $\bar{B}_i^{(k)}$ denotes the mean service time of a customer of class k at station i, and $V_i^{(k)}$ denotes the mean number of visits made by a customer of class k to station i.

2.2.1 Open network

The input stream is described by the vector

$$\boldsymbol{X} = (X^{(1)}, X^{(2)}, \ldots, X^{(K)}).$$

The throughput of station i for tasks of class k is

$$X_i^{(k)} = X^{(k)} V_i^{(k)} ,$$

the utilization factor of station i associated with serving tasks of class k is

$$U_i^{(k)} = X_i^{(k)} \bar{B}_i^{(k)} = X^{(k)} \bar{D}_i^{(k)} ,$$

and the total load on station i due to serving all classes of customers is

$$U_i = \sum_{k=1}^{K} U_i^{(k)} .$$

For the network to remain stable, the utilizations of all stations must be less than one, i.e.

$$\max_i U_i < 1.$$

Suppose that, at queueing stations operating under the natural discipline, the mean service times are the same for all customer classes:

$$\bar{B}_i^{(k)} = \bar{B}_i^{(l)} = \bar{B}_i.$$

This is a serious limitation of the model, but only under this assumption can we proceed as in the case of a single customer class and assume that the waiting time for service is equal to the average number of customers of all classes present at that station multiplied by their common mean service time. It follows that

$$\bar{R}_i^{(k)} = \bar{B}_i^{(k)} \left[1 + \sum_{l=1}^{K} \bar{N}_i^{(l)} \right] .$$

Note that the total service demands for customers of different classes at a given station may nevertheless differ, because the numbers of visits made by customers of different classes to that station may differ:

$$V_i^{(k)} \neq V_i^{(l)},$$

and hence

$$\bar{D}_i^{(k)} \neq \bar{D}_i^{(l)}.$$

If we work only with total service demands and total response times, the above restriction is not immediately visible.

The total response time of a customer of class k at station i is

$$\bar{R}_{(c)i}^{(k)} = \bar{D}_i^{(k)} \left[1 + \sum_{l=1}^{K} \bar{N}_i^{(l)} \right] .$$

The expression in brackets is the same for all customer classes, so we may write

$$\frac{\bar{R}_{(c)i}^{(l)}}{\bar{D}_i^{(l)}} = \frac{\bar{R}_{(c)i}^{(k)}}{\bar{D}_i^{(k)}} ,$$

and therefore

$$\begin{aligned} \bar{R}_{(c)i}^{(k)} &= \bar{D}_i^{(k)} \left[1 + \sum_{l=1}^{K} \bar{N}_i^{(l)} \right] \\ &= \bar{D}_i^{(k)} \left[1 + \sum_{l=1}^{K} \bar{R}_{(c)i}^{(l)} X^{(l)} \right] \\ &= \bar{D}_i^{(k)} \left[1 + \frac{\bar{R}_{(c)i}^{(k)}}{\bar{D}_i^{(k)}} \sum_{l=1}^{K} \bar{D}_i^{(l)} X^{(l)} \right] , \end{aligned}$$

from which we obtain

$$\bar{R}_{(c)i}^{(k)} = \frac{\bar{D}_i^{(k)}}{1 - \sum_{l=1}^{K} \bar{D}_i^{(l)} X^{(l)}} = \frac{\bar{D}_i^{(k)}}{1 - \sum_{l=1}^{K} U_i^{(l)}} = \frac{\bar{D}_i^{(k)}}{1 - U_i} ,$$

and, using Little's law,

$$\bar{N}_i^{(k)} = X^{(k)} \bar{R}_{(c)i}^{(k)} = \frac{X^{(k)} \bar{D}_i^{(k)}}{1 - U_i} = \frac{U_i^{(k)}}{1 - U_i} .$$

Example 2.6

Fig. 2.10 shows a two-station open network serving two classes of customers. Station 1, representing the central processing unit, is modeled as a PS station; station 2, representing the disk, has the natural queueing discipline. Assume the following data:

- *input traffic intensities:* $X^{(1)} = 0.5$, $X^{(2)} = 1$;
- *mean service times:* $\bar{B}_1^{(1)} = 0.01$ *s*, $\bar{B}_1^{(2)} = 0.1$ *s*, $\bar{B}_2^{(1)} = 0.02$ *s*, $\bar{B}_2^{(2)} = 0.02$ *s;*
- *numbers of visits to the stations:* $V_1^{(1)} = 40$, $V_1^{(2)} = 4$, $V_2^{(1)} = 39$, $V_2^{(2)} = 3$.

Calculate the mean queue lengths and response times of the network.

We calculate:

- total service demands at the stations:

$$\begin{aligned} \bar{D}_1^{(1)} &= \bar{B}_1^{(1)} V_1^{(1)} = 0.01\,\mathrm{s} \times 40 = 0.4\,\mathrm{s}, \\ \bar{D}_1^{(2)} &= \bar{B}_1^{(2)} V_1^{(2)} = 0.1\,\mathrm{s} \times 4 = 0.4\,\mathrm{s}, \\ \bar{D}_2^{(1)} &= \bar{B}_2^{(1)} V_2^{(1)} = 0.02\,\mathrm{s} \times 39 = 0.78\,\mathrm{s}, \\ \bar{D}_2^{(2)} &= \bar{B}_2^{(2)} V_2^{(2)} = 0.02\,\mathrm{s} \times 3 = 0.06\,\mathrm{s}; \end{aligned}$$

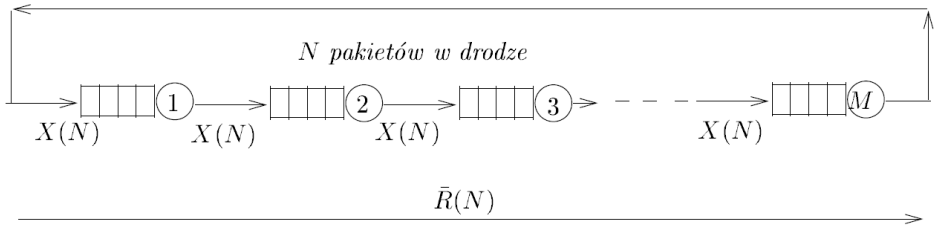


Figure 2.8: Virtual path with N packets between sender and receiver in Example 2.5

- utilization factors of the stations due to first- and second-class customers:

$$\begin{aligned} U_1^{(1)} &= X^{(1)}\bar{D}_1^{(1)} = 0.5\,\frac{1}{\mathrm{s}} \times 0.4\,\mathrm{s} = 0.2, \\ U_1^{(2)} &= X^{(2)}\bar{D}_1^{(2)} = 1.0\,\frac{1}{\mathrm{s}} \times 0.4\,\mathrm{s} = 0.4, \\ U_2^{(1)} &= X^{(1)}\bar{D}_2^{(1)} = 0.5\,\frac{1}{\mathrm{s}} \times 0.78\,\mathrm{s} = 0.39, \\ U_2^{(2)} &= X^{(2)}\bar{D}_2^{(2)} = 1.0\,\frac{1}{\mathrm{s}} \times 0.06\,\mathrm{s} = 0.06; \end{aligned}$$

  hence the total utilizations of the stations are

$$\begin{aligned} U_1 &= U_1^{(1)} + U_1^{(2)} = 0.2 + 0.4 = 0.6, \\ U_2 &= U_2^{(1)} + U_2^{(2)} = 0.39 + 0.06 = 0.45; \end{aligned}$$

- the mean numbers of first- and second-class customers at the stations:

$$\begin{aligned} \bar{N}_1^{(1)} &= \frac{U_1^{(1)}}{1-U_1} = \frac{0.2}{1-0.6} = 0.5, \\ \bar{N}_1^{(2)} &= \frac{U_1^{(2)}}{1-U_1} = \frac{0.4}{1-0.6} = 1.0, \\ \bar{N}_2^{(1)} &= \frac{U_2^{(1)}}{1-U_2} = \frac{0.39}{1-0.45} = 0.709, \\ \bar{N}_2^{(2)} &= \frac{U_2^{(2)}}{1-U_2} = \frac{0.06}{1-0.45} = 0.109; \end{aligned}$$

- the mean total response times for customers of each class at each station:

$$\begin{aligned} \bar{R}_{(c)1}^{(1)} &= \frac{\bar{D}_1^{(1)}}{1-U_1} = \frac{0.4}{1-0.6} = 1.0, \\ \bar{R}_{(c)1}^{(2)} &= \frac{\bar{D}_1^{(2)}}{1-U_1} = \frac{0.4}{1-0.6} = 1.0, \\ \bar{R}_{(c)2}^{(1)} &= \frac{\bar{D}_2^{(1)}}{1-U_2} = \frac{0.78}{1-0.45} = 1.418, \\ \bar{R}_{(c)2}^{(2)} &= \frac{\bar{D}_2^{(2)}}{1-U_2} = \frac{0.06}{1-0.45} = 0.109; \end{aligned}$$

hence the network response times for customers of each class are

$$\begin{aligned}\bar{R}^{(1)} &= \bar{R}^{(1)}_{(c)1} + \bar{R}^{(1)}_{(c)2} = 1.0 + 1.418 = 2.418,\\ \bar{R}^{(2)} &= \bar{R}^{(2)}_{(c)1} + \bar{R}^{(2)}_{(c)2} = 1.0 + 0.109 = 1.109.\end{aligned}$$

We can also calculate the network response times using Little's law:

$$\begin{aligned}\bar{R}^{(1)} &= \frac{\bar{N}^{(1)}_1 + \bar{N}^{(1)}_2}{X^{(1)}} = \frac{0.5 + 0.709}{0.5} = 2.418,\\ \bar{R}^{(2)} &= \frac{\bar{N}^{(2)}_1 + \bar{N}^{(2)}_2}{X^{(2)}} = \frac{1 + 0.109}{1} = 1.109.\end{aligned}$$

■

2.2.2 Closed network

The *MVA* algorithm for a closed network with multiple customer classes is

a simple generalisation of the procedure for a single class. We continue to denote by M the number of terminals, and by K the number of classes. The population of clients in the network is fixed and described by the vector $\boldsymbol{N} = [N^{(1)}, N^{(2)}, \ldots, N^{(K)}]$, where $N^{(k)}$, $k = 1, 2, \ldots, K$, is the number of customers of class k present in the network.

A customer of class k arriving at station i in a closed network with a population of $\boldsymbol{N}$ finds *at that moment* an average of as many customers as many as there are on average, if we reduce the network population by one customer of his class, and thus for the population, which we shall denote as $\boldsymbol{N} - \mathbf{1}^{(k)}$, where $\mathbf{1}^{(k)} = [0, \ldots, 0, 1, 0, \ldots, 0]$ is a K-dimensional vector whose k-th component is equal to one, and the others are zero.

$$\bar{R}^{(k)}_i(\boldsymbol{N}) = \bar{B}^{(k)}_i[1 + \bar{N}_i(\boldsymbol{N} - \mathbf{1}^{(k)})]\,, \qquad \begin{array}{l} i = 1, \ldots, M,\\ k = 1, \ldots, K.\end{array} \tag{2.7}$$

$\bar{N}_i = \sum_{k=1}^{K} \bar{N}^{(k)}_i$ is the average number of customers of all classes combined present at station i. As with open queues, we must assume that under the natural policy, all customer classes have the same average service time $\bar{B}_i$ at station i, but a different overall average service time $\bar{D}^{(k)}_i$. In the case of $LIFO$ and PS policies, equation (2.7) also holds and different service times $B^{(k)}_i$ are permitted for each class. For stations *IS*

$$\bar{R}^{(k)}_i(\boldsymbol{N}) = \bar{B}^{(k)}_i\,. \tag{2.8}$$

The total response time at a station (between successive passages of a customer through branch /x, where the network throughput is measured) is, respectively,

$$\bar{R}^{(k)}_{(c)i}(\boldsymbol{N}) = \bar{D}^{(k)}_i[1 + \bar{N}_i(\boldsymbol{N} - \mathbf{1}^{(k)})] \quad \text{for the } \textit{FIFO, LIFO, PS} \text{ queues}\,, \tag{2.9}$$

$$\bar{R}^{(k)}_{(c)i}(\boldsymbol{N}) = \bar{D}^{(k)}_i \quad \text{for } \textit{IS} \text{ queues}\,. \tag{2.10}$$

Figure 2.9: Intermediate populations for the model with a population of $(2, 2)$ customers

The equations determining the network throughput and the average number of customers of a given class at the stations — equivalents of equations (2.5,2.6) for a single class of customers — take the form

$$X^{(k)}(\boldsymbol{N}) = \frac{N^{(k)}}{\sum_{i=1}^{M} \bar{R}_{(c)i}^{(k)}(\boldsymbol{N})}, \qquad k = 1, \ldots, K, \tag{2.11}$$

$$\bar{N}_i^{(k)}(\boldsymbol{N}) = X^{(k)}(\boldsymbol{N}) \bar{R}_{(c)i}^{(k)}(\boldsymbol{N}), \qquad \begin{array}{l} i = 1, \ldots, M, \\ k = 1, \ldots, K. \end{array} \tag{2.12}$$

The presence of many classes greatly increases the computational complexity of the algorithm. The number of intermediate populations between $\mathbf{0}$ and $\boldsymbol{N}$, for which the model must be solved, grows very rapidly. Fig. 2.10 shows the intermediate states for the model with population $\boldsymbol{N} = (2, 2)$.

For illustration, let us assume that the *MVA* algorithm for a closed network of 10 stations containing 50 customers of a single class requires approx. 500 arithmetic operations and the storage of approx. 10 intermediate data items , whereas the algorithm for the same network containing also 50 customers, grouped into 5 classes of 10 customers each, requires over 8 million operations and the storage of over 145,000 intermediate data items [30].

The required computation time is proportional to the value

$$MK \prod_{k=1}^{K} (N^{(k)} + 1),$$

whilst the memory required increases with the expression

$$M \prod_{k=1, k \neq k_{max}}^{K} (N^{(k)} + 1),$$

where k_{max} is the index of the most numerous class. In some practical cases (telecommunications network models), these requirements are too excessive and the *MVA* algorithm must be used in an approximate form, discussed in subsection 2.3 .

2.2.3 Mixed network

A mixed network is a multi-class network which, for certain classes (let us call them closed classes), behaves as a closed network, while for others (referred to here as open classes) it behaves as an open network — Fig. 2.12.

The procedure in this case is as follows.

1. Let us denote the set of closed classes by $\mathcal{Z}$, and the set of open classes by $\mathcal{O}$. For all stations in the network, we first calculate their utilization due to each of the open classes:

$$U_i^{(k)} = X^{(k)} \bar{D}_i^{(k)}, \qquad \begin{array}{l} i = 1, \ldots, M, \\ k \in \mathcal{O}. \end{array}$$

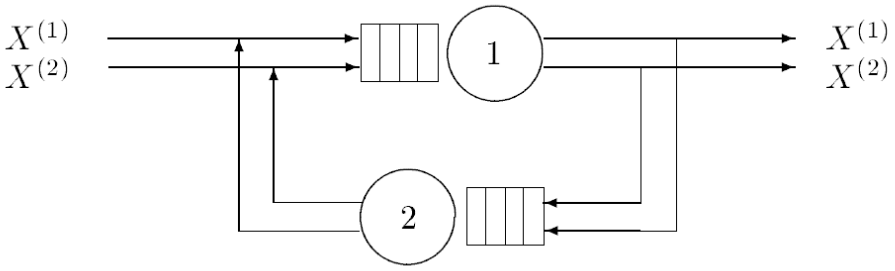


Figure 2.10: Two-class model in Example 2.6

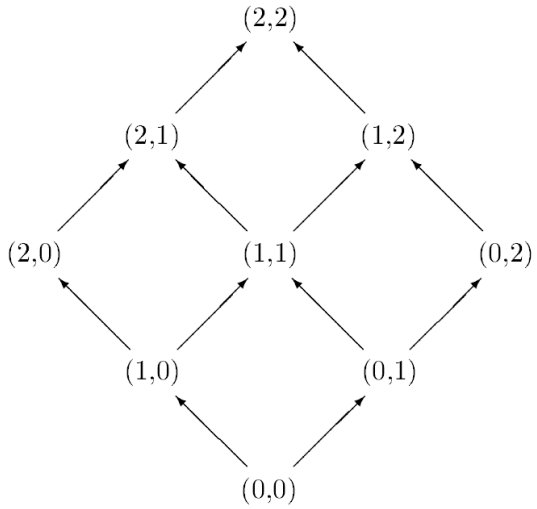


Figure 2.11: Intermediate populations for the model with population $(2, 2)$ customers

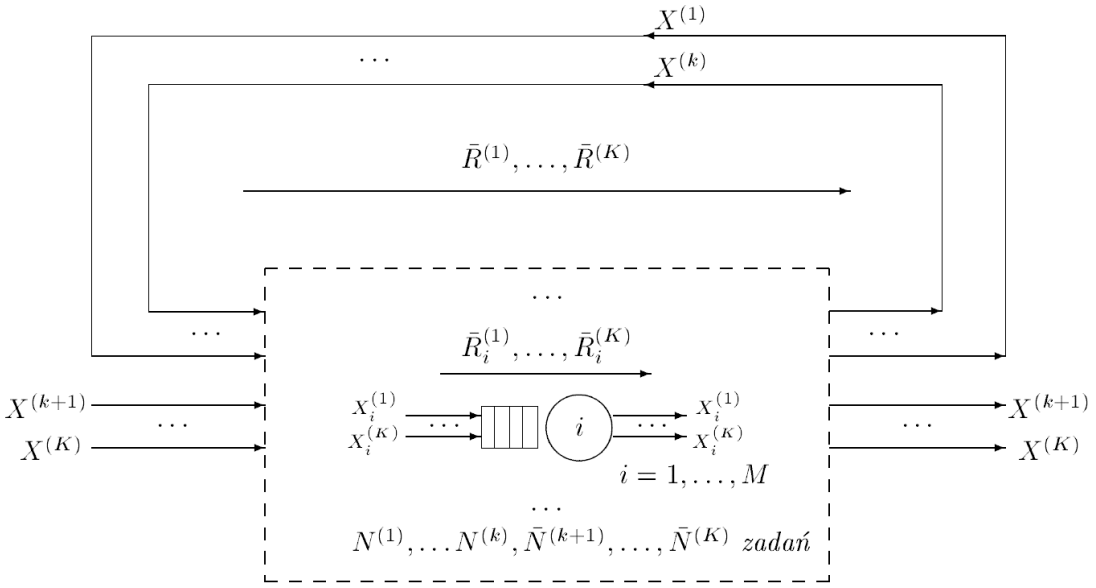


Figure 2.12: Mixed network with open and closed customer classes

We then calculate the total utilization due to all open classes:

$$U_{\mathcal{O}i} = \sum_{k \in \mathcal{O}} U_i^{(k)} , \qquad i = 1, \ldots, M.$$

2. Next, we solve the model for the closed classes. The presence of open classes, which do not explicitly appear in the model at this stage, is taken into account indirectly. Since a fraction of the service capacity (equal to $U_{\mathcal{O}i}$) is used to serve customers from the open classes, the effective service demand for the closed classes must be increased accordingly.

   The new total mean service demand for the closed classes is

$$\bar{D}_i^{(k*)} = \frac{\bar{D}_i^{(k)}}{1 - U_{\mathcal{O}i}} , \qquad \begin{array}{l} i = 1, \ldots, M, \\ k \in \mathcal{Z} . \end{array}$$

   The workstation throughputs, queue lengths, and response times for the closed classes, calculated according to the model described in the previous subsection, are then correct.

3. Finally, for the open classes we calculate the response times while taking into account the presence of closed-class customers at the stations:

$$\begin{aligned} \bar{R}_{(c)i}^{(k)} &= \frac{\bar{D}_i^{(k)}[1 + \bar{N}_{\mathcal{Z}i}]}{1 - U_{\mathcal{O}i}} , \\ \bar{N}_i^{(k)} &= X^{(k)} \bar{R}_{(c)i}^{(k)} , \qquad \begin{array}{l} i = 1, \ldots, M, \\ k \in \mathcal{O} . \end{array} \end{aligned}$$

   Here, $\bar{N}_{\mathcal{Z}i}$ denotes the mean number of customers of all closed classes present at station i.

Example 2.7

Let us return to Example 2.5, in which the model has the form of an open network, shown in Fig. 2.10, serving two classes of customers. To these two open customer classes, with the same parameters as in Example 2.5, let us add a third, closed class — yielding the model shown in Fig. 2.12a — for which the mean service times and the numbers of visits to the first and second stations are:

$$\bar{B}_1^{(3)} = 0.02\ s, \qquad \bar{B}_2^{(3)} = 0.02\ s, \qquad V_1^{(3)} = 5, \qquad V_2^{(3)} = 4.$$

Analyze the resulting mixed network, assuming the presence of $N^{(3)} = 5$ third-class customers.

The first step of the mixed-network algorithm, i.e. the preliminary calculations for the open classes (Fig. 2.12b), was already carried out in Example 2.5. We therefore already know the values of

$$U_{\mathcal{O}i} = U_i^{(1)} + U_i^{(2)}\ :$$

$$U_{\mathcal{O}1} = U_1^{(1)} + U_1^{(2)} = 0.6, \qquad U_{\mathcal{O}2} = U_2^{(1)} + U_2^{(2)} = 0.45.$$

We may now proceed to the second step and rescale the service demands of the closed-class customer:

$$\bar{D}_1^{(3)*} = \frac{\bar{D}_1^{(3)}}{1 - U_{\mathcal{O}1}} = \frac{0.1\,\mathrm{s}}{1 - 0.6} = 0.25\,\mathrm{s}, \qquad \bar{D}_2^{(3)*} = \frac{\bar{D}_2^{(3)}}{1 - U_{\mathcal{O}2}} = \frac{0.08\,\mathrm{s}}{1 - 0.45} = 0.1455\,\mathrm{s}.$$

We then apply the *MVA* algorithm to the closed network containing only the third customer class — Fig. 2.12c.

The results of the recursive calculations

$$\begin{aligned}
\bar{R}_{(c)1}^{(3)}(N^{(3)}) &= \bar{D}_1^{(3)*}[1 + \bar{N}_1^{(3)}(N^{(3)} - 1)]\,, \\
\bar{R}_{(c)2}^{(3)}(N^{(3)}) &= \bar{D}_2^{(3)*}[1 + \bar{N}_2^{(3)}(N^{(3)} - 1)]\,, \\
X^{(3)}(N^{(3)}) &= \frac{N^{(3)}}{\bar{R}_{(c)1}^{(3)}(N^{(3)}) + \bar{R}_{(c)2}^{(3)}(N^{(3)})}\,, \\
\bar{N}_1^{(3)}(N^{(3)}) &= \bar{R}_{(c)1}^{(3)}(N^{(3)}) X^{(3)}(N^{(3)})\,, \\
\bar{N}_2^{(3)}(N^{(3)}) &= \bar{R}_{(c)2}^{(3)}(N^{(3)}) X^{(3)}(N^{(3)})\,.
\end{aligned}$$

for $N^{(3)} = 1, \ldots, 5$ are shown in Table 2.3.

After determining $\bar{N}_{\mathcal{Z}i}$:

$$\bar{N}_{\mathcal{Z}1} = \bar{N}_1^{(3)}, \qquad \bar{N}_{\mathcal{Z}2} = \bar{N}_2^{(3)},$$

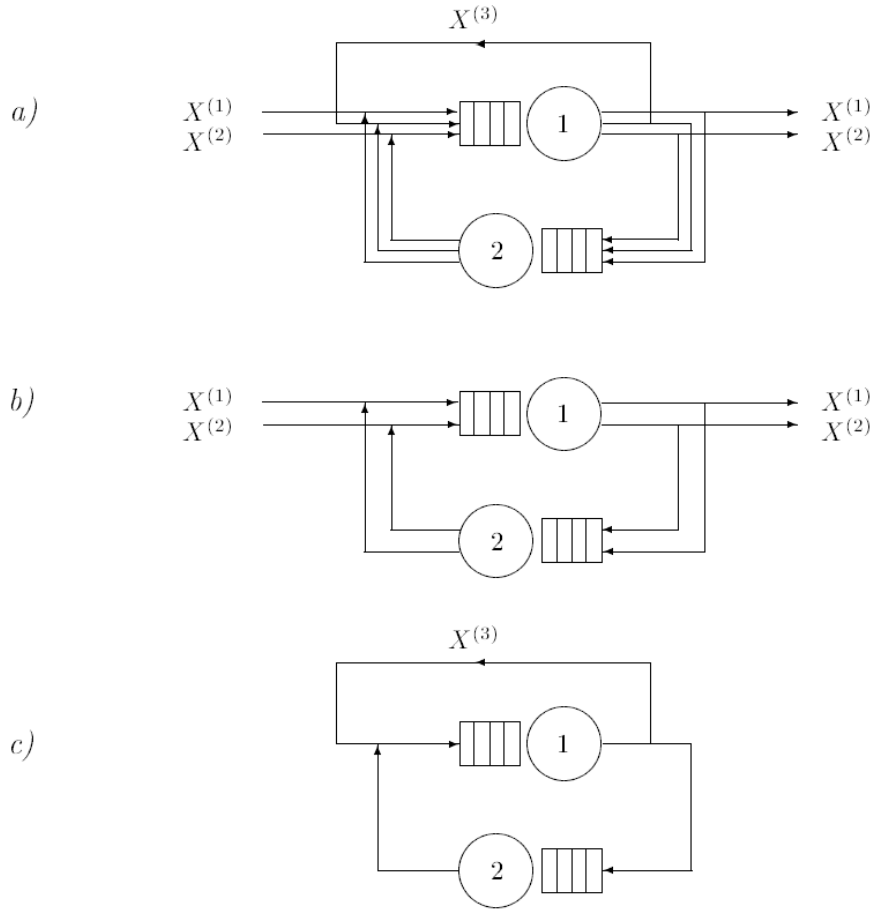


Figure 2.13: Model in Example 2.7

$N^{(3)}$	$\bar{R}^{(3)}_{(c)1}$	$\bar{R}^{(3)}_{(c)2}$	$X^{(3)}$	$\bar{N}^{(3)}_1$	$\bar{N}^{(3)}_2$	$U^{(3)}_1$	$U^{(3)}_2$
1	0.2500	0.1455	2.5287	0.6322	0.3678	0.6322	0.3678
2	0.4080	0.1990	3.2949	1.3445	0.6555	0.8237	0.4793
3	0.5861	0.2408	3.6279	2.1264	0.8736	0.9070	0.5277
4	0.7816	0.2725	3.7964	2.9659	1.0341	0.9487	0.5520
5	0.9915	0.2959	3.8840	3.8508	1.1492	0.9710	0.5649

Table 2.3: Calculation results for the closed class in Example 2.5

we proceed to step 3 of the algorithm and calculate

$$\begin{aligned}
\bar{R}^{(1)}_{(c)1} &= \frac{\bar{D}^{(1)}_1[1+\bar{N}^{(3)}_1]}{1-U_{\mathcal{O}1}} = \frac{0.4\times[1+3.8508]}{1-0.6} = 4.8508\,,\\
\bar{R}^{(2)}_{(c)1} &= \frac{\bar{D}^{(2)}_1[1+\bar{N}^{(3)}_1]}{1-U_{\mathcal{O}1}} = \frac{0.4\times[1+3.8508]}{1-0.6} = 4.8508\,,\\
\bar{R}^{(1)}_{(c)2} &= \frac{\bar{D}^{(1)}_2[1+\bar{N}^{(3)}_2]}{1-U_{\mathcal{O}2}} = \frac{0.78\times[1+1.1492]}{1-0.45} = 3.0480\,,\\
\bar{R}^{(2)}_{(c)2} &= \frac{\bar{D}^{(2)}_2[1+\bar{N}^{(3)}_2]}{1-U_{\mathcal{O}2}} = \frac{0.06\times[1+1.1492]}{1-0.45} = 0.2345\,.
\end{aligned}$$

Hence,

$$\begin{aligned}
\bar{N}^{(1)}_1 &= X^{(1)}\bar{R}^{(1)}_{(c)1} = 0.5\times 4.8508 = 2.4254\,,\\
\bar{N}^{(2)}_1 &= X^{(2)}\bar{R}^{(2)}_{(c)1} = 1.0\times 4.8508 = 4.8508\,,\\
\bar{N}^{(1)}_2 &= X^{(1)}\bar{R}^{(1)}_{(c)2} = 0.5\times 3.0480 = 1.5240\,,\\
\bar{N}^{(2)}_2 &= X^{(2)}\bar{R}^{(2)}_{(c)2} = 1.0\times 0.2345 = 0.2345\,.
\end{aligned}$$

■

2.3 Approximate algorithms

In telecommunications network models, each customer class corresponds to a single virtual path together with its associated window-based packet flow control mechanism, see Example 2.3. The size of the class is equal to the window size. There may be hundreds or thousands of such paths, and the computational complexity of an exact *MVA* algorithm for closed networks is, in this case, too high.

Traversing all intermediate populations between $\mathbf{0}$ and $\boldsymbol{N}$ requires computational effort and memory usage that grow very rapidly with the number of classes and their populations. Therefore, approximate *MVA* algorithms for closed networks [14, 49, 109] avoid such a procedure and attempt to estimate the mean value $\bar{N}^{(l)}_i(\boldsymbol{N}-\mathbf{1}^{(k)})$, i.e. the number of customers of class l present at station i at the moment a customer of class k arrives, on the basis of the value $\bar{N}^{(l)}_i(\boldsymbol{N})$. We do not know this latter value either, but estimate it iteratively.

As a starting point (the values at iteration zero), we assume an initial distribution of customers across the stations, $\bar{N}^{(k)[0]}_i$, e.g. by assuming that the number $N^{(k)}$ of customers of class k present in the network is distributed evenly across all stations:

$$\bar{N}^{(k)[0]}_i = \frac{N^{(k)}}{M}\,, \qquad \begin{array}{l} i=1,\ldots,M,\\ k=1,\ldots,K,\end{array}$$

or by assuming that the more customers there are at a station, the greater its total service demand:

$$\bar{N}^{(k)[0]}_i = N^{(k)}\frac{\bar{D}^{(k)}_i}{\sum_{j=1}^{M}\bar{D}^{(k)}_j}\,, \qquad \begin{array}{l} i=1,\ldots,M,\\ k=1,\ldots,K.\end{array}$$

We also assume a relationship between $\bar{N}_i^{(l)}(\boldsymbol{N})$ and $\bar{N}_i^{(l)}(\boldsymbol{N}-\mathbf{1}^{(k)})$, e.g.

$$\bar{N}_i^{(l)}(\boldsymbol{N}-\mathbf{1}^{(k)}) = \begin{cases} \bar{N}_i^{(l)}(\boldsymbol{N}) & \text{if} \quad l \neq k\,, \\ \bar{N}_i^{(k)}(\boldsymbol{N}) - \gamma_i^{(k)} & \text{if} \quad l = k\,, \end{cases}$$

where

$$\gamma_i^{(k)} = \frac{\bar{D}_i^{(k)}}{\sum_{j=1}^{M} \bar{D}_j^{(k)}}$$

and then perform the calculations in the *MVA* loop based on equations (2.7)–(2.12).

The approximate *MVA* algorithm for closed networks therefore takes the form:

1. Set the initial conditions:

$$\bar{N}_i^{(k)[0]} = \gamma_i^{(k)} N^{(k)}, \qquad \begin{array}{l} i = 1, \ldots, M, \\ k = 1, \ldots, K; \end{array}$$

2. Compute, in the next (jth) iteration (values obtained in the jth iteration are denoted by the superscript $[j]$, to distinguish them from the class index):

$$\bar{R}_{(c)i}^{(k)[j]}(\boldsymbol{N}) \;=\; \begin{cases} \bar{D}_i^{(k)} \left[1 + \sum_{l=1}^{K} \bar{N}_i^{(l)[j-1]}(\boldsymbol{N}) - \gamma_i^{(k)}\right] & \text{for } \textit{FIFO, LIFO, PS}, \\ \bar{D}_i^{(k)} & \text{for } \textit{IS}; \end{cases} \qquad \begin{array}{l} i = 1, \ldots, M, \\ k = 1, \ldots, K; \end{array}$$

3. Calculate

$$X^{(k)[j]}(\boldsymbol{N}) = \frac{N^{(k)}}{\sum_{i=1}^{M} \bar{R}_{(c)i}^{(k)[j]}(\boldsymbol{N})}\,, \qquad k = 1, \ldots, K;$$

4. Calculate

$$\bar{N}_i^{(k)[j]}(\boldsymbol{N}) = X^{(k)[j]}(\boldsymbol{N}) \bar{R}_{(c)i}^{(k)[j]}(\boldsymbol{N})\,, \qquad \begin{array}{l} i = 1, \ldots, M, \\ k = 1, \ldots, K; \end{array}$$

5. Check whether a fixed point of the iteration has been reached, i.e. whether the values calculated in successive iterations, $\bar{R}_{(c)i}^{(k)[j]}(\boldsymbol{N})$, $X^{(k)[j]}(\boldsymbol{N})$, and $\bar{N}_i^{(k)[j]}(\boldsymbol{N})$, differ by less than a predetermined small value ϵ. If so, the procedure terminates; otherwise, $j \leftarrow j+1$ and we return to step 2.

After only one pass through steps 1–4, the results are already close to the exact values, and practical convergence is achieved after only a few iterations j.

Example 2.8

In a closed network consisting of two service stations ($M = 2$), there are $N = 1000$ customers, and the mean service times are $\bar{B}_1 = 0.1$ and $\bar{B}_2 = 0.9$ time units. Using the approximate MVA method, determine the response time and the mean number of tasks present at both stations.

For the exact *MVA* algorithm:

$$\bar{R}_i(1000) = \bar{B}_i[1 + \bar{N}_i(999)], \qquad i = 1, 2.$$

For the approximate algorithm:

$$\begin{aligned} \bar{R}_i(1000) &\approx \bar{B}_i[1 + \bar{N}_i(1000) - \gamma_i] \\ &= \bar{B}_i \left[1 + \bar{N}_i(1000) - \frac{\bar{B}_i}{\bar{B}_1 + \bar{B}_2}\right]. \end{aligned}$$

The value of $\bar{N}_1(1000)$ is unknown. As a first approximation, we may assume $\bar{N}_i^{[0]} = N/M$, i.e.

$$\bar{N}_1^{[0]}(1000) = \bar{N}_2^{[0]}(1000) = \frac{1000}{2} = 500,$$

but it is more reasonable to choose

$$\bar{N}_i^{[0]} = \frac{\bar{D}_i}{\bar{D}} N,$$

from which

$$\bar{N}_1^{[0]}(1000) = \frac{\bar{B}_1}{\bar{B}_1 + \bar{B}_2} N = 100, \qquad \bar{N}_2^{[0]}(1000) = \frac{\bar{B}_2}{\bar{B}_1 + \bar{B}_2} N = 900.$$

First iteration:

$$\begin{aligned} \bar{R}_1^{[1]}(1000) &\approx \bar{B}_1[1 + \bar{N}_1^{[0]}(1000) - \gamma_1] \\ &= 0.1[1 + 100 - 0.1] = 10.09, \\ \bar{R}_2^{[1]}(1000) &\approx \bar{B}_2[1 + \bar{N}_2^{[0]}(1000) - \gamma_2] \\ &= 0.9[1 + 900 - 0.9] = 810.09. \end{aligned}$$

$$X^{[1]}(1000) = \frac{1000}{\bar{R}_1^{[1]}(1000) + \bar{R}_2^{[1]}(1000)} = \frac{1000}{10.09 + 810.09} = 1.2192446.$$

$$\begin{aligned} \bar{N}_1^{[1]}(1000) &= X^{[1]}(1000)\, \bar{R}_1^{[1]}(1000) = 1.2192446 \times 10.09 = 12.302178, \\ \bar{N}_2^{[1]}(1000) &= X^{[1]}(1000)\, \bar{R}_2^{[1]}(1000) = 1.2192446 \times 810.09 = 987.69782. \end{aligned}$$

Second iteration:

$$\begin{aligned} \bar{R}_1^{[2]}(1000) &\approx \bar{B}_1[1 + \bar{N}_1^{[1]}(1000) - \gamma_1] \\ &= 0.1[1 + 12.302178 - 0.1] = 1.3202178, \\ \bar{R}_2^{[2]}(1000) &\approx \bar{B}_2[1 + \bar{N}_2^{[1]}(1000) - \gamma_2] \\ &= 0.9[1 + 987.69782 - 0.9] = 889.01804. \end{aligned}$$

$$X^{[2]}(1000) = \frac{1000}{\bar{R}_1^{[2]}(1000) + \bar{R}_2^{[2]}(1000)} = \frac{1000}{1.3202178 + 889.01804} = 1.1231686.$$

$$\begin{aligned}
\bar{N}_1^{[2]}(1000) &= X^{[2]}(1000)\,\bar{R}_1^{[2]}(1000) = 1.1231686 \times 1.3202178 = 1.4828272, \\
\bar{N}_2^{[2]}(1000) &= X^{[2]}(1000)\,\bar{R}_2^{[2]}(1000) = 1.1231686 \times 889.01804 = 998.51717.
\end{aligned}$$

Third iteration:

$$\begin{aligned}
\bar{R}_1^{[3]}(1000) &\approx \bar{B}_1[1 + \bar{N}_1^{[2]}(1000) - \gamma_1] \\
&= 0.1[1 + 1.4828272 - 0.1] = 0.23828272, \\
\bar{R}_2^{[3]}(1000) &\approx \bar{B}_2[1 + \bar{N}_2^{[2]}(1000) - \gamma_2] \\
&= 0.9[1 + 998.51717 - 0.9] = 898.75545.
\end{aligned}$$

$$X^{[3]}(1000) = \frac{1000}{\bar{R}_1^{[3]}(1000) + \bar{R}_2^{[3]}(1000)} = \frac{1000}{0.23828272 + 898.75545} = 1.1123424.$$

$$\begin{aligned}
\bar{N}_1^{[3]}(1000) &= X^{[3]}(1000)\,\bar{R}_1^{[3]}(1000) = 1.1123424 \times 0.23828272 = 0.2650547, \\
\bar{N}_2^{[3]}(1000) &= X^{[3]}(1000)\,\bar{R}_2^{[3]}(1000) = 1.1123424 \times 898.75545 = 999.73495.
\end{aligned}$$

Etc.
One can predict that

$$\bar{N}_1 \approx \frac{0.1}{0.9} = 0.1111, \qquad \bar{N}_2 = 1000 - \bar{N}_1.$$

■

Chapter 3

Markov simple models

3.1 Probabilistic models – introduction

From this point onward, we assume that the interarrival time between successive customers at a station is a random variable described by the distribution $A(x)$, and that the service time is also a random variable described by the distribution $B(x)$.

This assumption is realistic: apart from special cases (e.g. periodically launched control or monitoring programs), task arrivals occur at random times, and their execution times, which may depend on input data, can also be treated as random variables.

This is illustrated by the following example of execution time analysis for a simple program, taken from *The Art of Computer Programming* [63]. The example also provides an opportunity to review some concepts related to random variables and their distributions. A more systematic discussion of concepts such as probability distributions, probability density functions, moments, generating functions, and the distributions used later in this book is given in Appendix A.

Example 3.1

The following example presents an analysis of the execution time of the MAX program, which selects the largest of n numbers randomly distributed in a one-dimensional array B. The program, written in Pascal, is shown below.

```
 program MAX(input, output);
label 1,2,3,4,5,6;
const n = 100;
var j,k,m: integer;
    B: array [1 .. n] of integer;
begin
   1: j := n; k := n − 1; m := B[n];
   2: while (k > 0) do
      begin
   3: if B[k] > m then
         begin
   4:       j := k;     m := B[k];
         end;
   5: k := k − 1
      end;
   6: writeln(j, m)     end.
```

step	number of executions	number of instructions
1	1	3
2	n	1
3	$n-1$	1
4	X_n	2
5	$n-1$	1
6	1	1

Table 3.1: Number of executions of individual steps of the *MAX* program

The array is scanned from the end. Initially, variable m is assigned the value of the last element $B[n]$. Then m is compared with the successive elements $B[k]$, for $k = n-1, \ldots, 1$. Whenever $B[k]$ is greater than m, the value of $B[k]$ is assigned to m, and the index k is stored in variable j. In this way, after all elements have been examined, the largest value is stored in m.

The numbered program fragments are listed in Table 3.1. Step 4 — assigning a new value to variable m — is executed X_n times. The index n indicates that X_n depends on the length of the scanned array.

The value of X_n depends on the arrangement of the numbers in array B. Thus, X_n is a discrete random variable taking values from the set $\{0, 1, \ldots, n-1\}$.

In the boundary cases:

$$X_n = 0, \quad \text{if} \quad B[n] = \max_k \; B[k] \;,$$

$$X_n = n-1, \quad \text{if} \quad B[1] > B[2] > \cdots > B[n] \,.$$

For simplicity, we assume that all numbers are distinct, i.e. $B[i] \neq B[j]$ whenever $i \neq j$. We also assume that the arrangement of the numbers is completely random, meaning that each permutation is equally likely.

Let $p_n(k)$ denote the probability that, during the scan of the array, the current maximum value must be updated exactly k times:

$$p_n(k) = P[X_n = k] = \frac{\text{number of permutations of } n \text{ elements for which } X_n = k}{n!} \,.$$

Let A denote the event that the maximum value is located in $B[1]$, so that it is assigned to m during the last pass through the loop. In this case, step *4* is executed and the value of X_n increases by one:

$$X_n = X_{n-1} + 1 \,.$$

Let $\bar{A}$ denote the complementary event: the maximum value has already been found earlier, and therefore step *4* is not executed during the final iteration of the loop. In this case:

$$X_n = X_{n-1} \,.$$

Clearly,

$$P(A) + P(\bar{A}) = 1 \,.$$

Since each element of the array is equally likely to contain the maximum value,

$$P(A) = \frac{1}{n} \,, \qquad P(\bar{A}) = \frac{n-1}{n} \,.$$

Let us write the conditional-probability formula:

$$P(X_n = k) = P(X_n = k \mid A)P(A) + P(X_n = k \mid \bar{A})P(\bar{A}) \,,$$

that is,

$$p_n(k) = p_{n-1}(k-1)\,\frac{1}{n} + p_{n-1}(k)\,\frac{n-1}{n} \,. \tag{3.1}$$

For $n = 1$, step *4* is not executed, so $X_1 = 0$, $p_1(0) = 1$, and $p_1(1) = 0$. The above relations allow us to compute $p_n(k)$ recursively.

We can also determine the generating function

$$G_{X_n}(z) = \sum_{k\geq 0} p_n(k) z^k$$

of the random variable X_n. For $n = 1$, it has the form

$$G_{X_1}(z) = p_1(0)z^0 + p_1(1)z = 1.$$

Multiplying both sides of equation (3.3) by z^k, for $k = 1, 2, \ldots$, and summing over k, we obtain

$$\sum_{k=1}^{\infty} p_n(k) z^k = \frac{z}{n} \sum_{k=1}^{\infty} p_{n-1}(k-1) z^{k-1} + \frac{n-1}{n} \sum_{k=1}^{\infty} p_{n-1}(k)\, z^k \,,$$

that is,

$$G_{X_n}(z) - p_n(0) = \frac{z}{n}\, G_{X_{n-1}}(z) + \frac{n-1}{n} \left[G_{X_{n-1}}(z) - p_{n-1}(0) \right] \,.$$

Taking into account the known values

$$p_n(0) = \frac{1}{n} \qquad \text{and} \qquad p_{n-1}(0) = \frac{1}{n-1},$$

we obtain

$$G_{X_n}(z) = G_{X_{n-1}}(z)\, \frac{z+n-1}{n} \,. \tag{3.2}$$

Hence,

$$G_{X_n}(z) = \frac{z+n-1}{n} \frac{z+n-2}{n-1} \cdots \frac{z+1}{2}\, G_{X_1}(z) = \frac{(z+n-1)(z+n-2)\cdots(z+1)}{n\,!} \,.$$

The probability distribution $p_n(k)$ can be obtained directly from $G_{X_n}(z)$ (see Appendix A) as

$$p_n(k) = \frac{1}{k!}\, \frac{\partial^k G_{X_n}(z)}{\partial z^k}\bigg|_{z=0} \,.$$

One may also use the identity

$$z(z+1)\ldots(z+n-1) = \sum_{k=0}^{n} \begin{bmatrix} n \\ k \end{bmatrix} z^k \,,$$

to represent $G_{X_n}(z)$ as

$$G_{X_n}(z) = \frac{(z+n-1)(z+n-2)\cdots(z+1)}{n\,!} = \frac{1}{n!} \sum_{k=0}^{n} \begin{bmatrix} n \\ k \end{bmatrix} z^{k-1} \,,$$

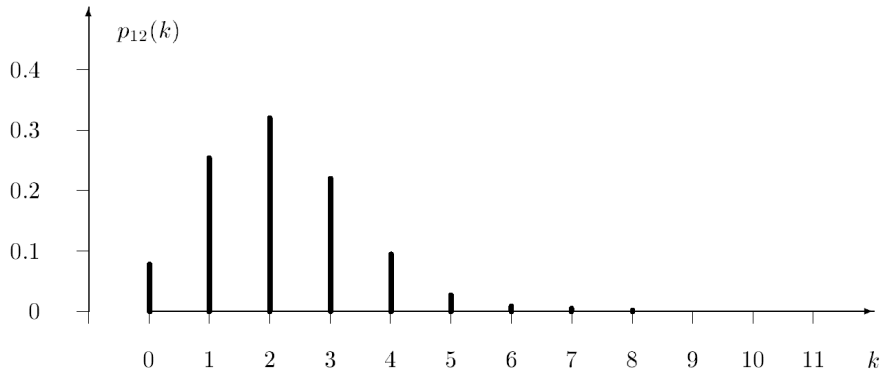


Figure 3.1: Distribution of the number of executions of step *4* in the *MAX* program for $n = 12$. The mean of this distribution is $E[X_{12}] = \sum_{i=2}^{12} \frac{1}{i} \approx 2.10$, and the variance is $\mathrm{var}[X_{12}] = \sum_{k=2}^{12} \frac{1}{k}(1 - \frac{1}{k}) \approx 1.5382$.

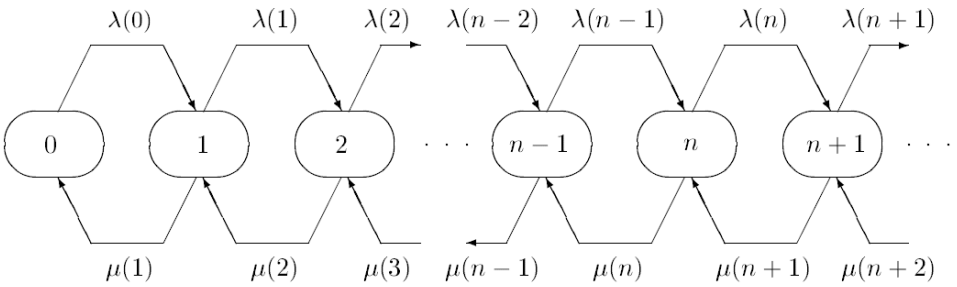


Figure 3.2: Transition graph for $M(n)/M(n)/1$ queueing system.

that is,

$$p_n(k) = \frac{1}{n!} \begin{bmatrix} n \\ k+1 \end{bmatrix} .$$

The symbols $\begin{bmatrix} n \\ k \end{bmatrix}$ denote the Stirling numbers of the first kind, whose values can be found in the relevant tables, e.g. [63].

Fig. 3.1 shows an example of the distribution of probabilities $p_n(k)$ for $n = 12$.

The generating function allows us to calculate the mean value (and all other moments of the distribution of the random variable X_n) directly:

$$E[X_n] = \left. \frac{d\, G_{X_n}(z)}{d\, z} \right|_{z=1} ,$$

however, it is more convenient to use the recursive relation (3.4):

$$\frac{d\, G_{X_n}(z)}{d\, z} = \frac{1}{n}\, G_{X_{n-1}}(z) + \frac{z+n-1}{n}\, \frac{d\, G_{X_{n-1}}(z)}{d\, z} ,$$

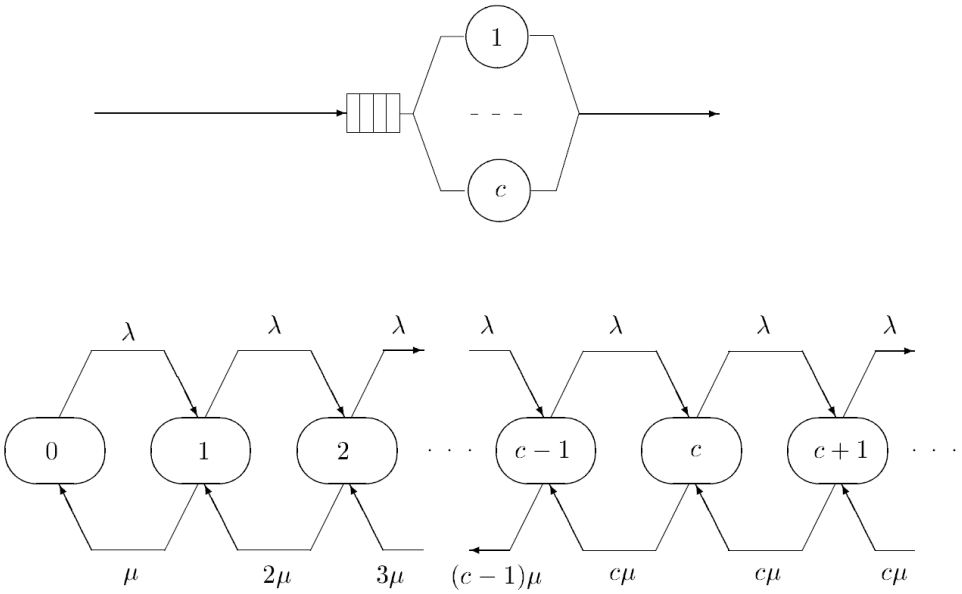


Figure 3.3: Transition graph for $M/M/c$ queueing system.

and therefore

$$E[X_n] = \left.\frac{d\,G_{X_n}(z)}{d\,z}\right|_{z=1} = \frac{1}{n}\,G_{X_{n-1}}(1) + \left.\frac{d\,G_{X_{n-1}}(z)}{d\,z}\right|_{z=1} = \frac{1}{n} + E[X_{n-1}]\,,$$

that is, taking into account that $E[X_1] = 0$,

$$E[X_n] = \sum_{i=2}^{n} \frac{1}{i} \approx \ln n\,.$$

Similarly, the variance of X_n can be derived without computing it directly. Note that the generating function $G_{X_n}(z)$ has the same form as the generating function of another random variable.

Let Y_k denote a Bernoulli random variable (see Appendix A) with parameter $p = 1/k$. Its generating function is

$$G_{Y_k}(z) = (1-p)z^0 + pz = \left(1 - \frac{1}{k}\right) + \frac{1}{k}\,z = \frac{k-1+z}{k}\,.$$

If W is the sum of the independent random variables $Y_2, Y_3, \ldots, Y_n$,

$$W = \sum_{k=2}^{n} Y_k,$$

then

$$G_W(z) = \prod_{k=2}^{n} G_{Y_k}(z) = \prod_{k=2}^{n} \frac{z+k-1}{k} = G_{X_n}(z)\,,$$

that is, X_n and W have the same distribution. Since the variables Y_k are independent,

$$\mathrm{var}[X_n] = \sum_{k=2}^{n} \mathrm{var}[Y_k] = \sum_{k=2}^{n} \frac{1}{k}\left(1 - \frac{1}{k}\right) .$$

To summarize: the program execution time is a random variable

$$S = S_1 + S_2 X_n,$$

where S_1 is a constant corresponding to the execution time of steps *1÷3* and *5÷6*, S_2 is the constant execution time of a single performance of step *4*, and X_n is a random variable whose generating function has been determined above.

■

Let us write the conditional-probability formula:

$$P(X_n = k) = P(X_n = k \mid A)P(A) + P(X_n = k \mid \bar{A})P(\bar{A}) ,$$

that is,

$$p_n(k) = p_{n-1}(k-1)\,\frac{1}{n} + p_{n-1}(k)\,\frac{n-1}{n} . \tag{3.3}$$

For $n = 1$, step *4* is not executed, so $X_1 = 0$, $p_1(0) = 1$, and $p_1(1) = 0$. The above relations allow us to compute $p_n(k)$ recursively.

We can also determine the generating function

$$G_{X_n}(z) = \sum_{k\geq 0} p_n(k) z^k$$

of the random variable X_n. For $n = 1$, it has the form

$$G_{X_1}(z) = p_1(0)z^0 + p_1(1)z = 1.$$

Multiplying both sides of equation (3.3) by z^k, for $k = 1, 2, \ldots$, and summing over k, we obtain

$$\sum_{k=1}^{\infty} p_n(k) z^k = \frac{z}{n} \sum_{k=1}^{\infty} p_{n-1}(k-1) z^{k-1} + \frac{n-1}{n} \sum_{k=1}^{\infty} p_{n-1}(k)\, z^k ,$$

that is,

$$G_{X_n}(z) - p_n(0) = \frac{z}{n}\, G_{X_{n-1}}(z) + \frac{n-1}{n}\, \left[G_{X_{n-1}}(z) - p_{n-1}(0)\right] .$$

Taking into account the known values

$$p_n(0) = \frac{1}{n} \qquad \text{and} \qquad p_{n-1}(0) = \frac{1}{n-1},$$

we obtain

$$G_{X_n}(z) = G_{X_{n-1}}(z)\,\frac{z+n-1}{n} . \tag{3.4}$$

Hence,

$$G_{X_n}(z) = \frac{z+n-1}{n}\,\frac{z+n-2}{n-1} \cdots \frac{z+1}{2}\, G_{X_1}(z) = \frac{(z+n-1)(z+n-2)\cdots(z+1)}{n\,!} .$$

The probability distribution $p_n(k)$ can be obtained directly from $G_{X_n}(z)$ (see Appendix **??**) as

$$p_n(k) = \frac{1}{k!} \left. \frac{\partial^k G_{X_n}(z)}{\partial z^k} \right|_{z=0} .$$

One may also use the identity

$$z(z+1)\dots(z+n-1) = \sum_{k=0}^{n} \begin{bmatrix} n \\ k \end{bmatrix} z^k ,$$

to represent $G_{X_n}(z)$ as

$$G_{X_n}(z) = \frac{(z+n-1)(z+n-2)\cdots(z+1)}{n\,!} = \frac{1}{n!} \sum_{k=0}^{n} \begin{bmatrix} n \\ k \end{bmatrix} z^{k-1} ,$$

that is,

$$p_n(k) = \frac{1}{n!} \begin{bmatrix} n \\ k+1 \end{bmatrix} .$$

The symbols $\left[\begin{smallmatrix} n \\ k \end{smallmatrix}\right]$ denote Stirling numbers of the first kind, whose values can be found in the relevant tables, e.g. [63].

Fig. 3.1 shows an example of the distribution of probabilities $p_n(k)$ for $n = 12$.

The generating function allows us to calculate the mean value (and all other moments of the distribution of the random variable X_n) directly:

$$E[X_n] = \left. \frac{d\, G_{X_n}(z)}{d\, z} \right|_{z=1} ,$$

however, it is more convenient to use the recursive relation (3.4):

$$\frac{d\, G_{X_n}(z)}{d\, z} = \frac{1}{n}\, G_{X_{n-1}}(z) + \frac{z+n-1}{n}\, \frac{d\, G_{X_{n-1}}(z)}{d\, z} ,$$

and therefore

$$E[X_n] = \left. \frac{d\, G_{X_n}(z)}{d\, z} \right|_{z=1} = \frac{1}{n}\, G_{X_{n-1}}(1) + \left. \frac{d\, G_{X_{n-1}}(z)}{d\, z} \right|_{z=1} = \frac{1}{n} + E[X_{n-1}] ,$$

that is, taking into account that $E[X_1] = 0$,

$$E[X_n] = \sum_{i=2}^{n} \frac{1}{i} \approx \ln n .$$

Similarly, the variance of X_n can be derived without computing it directly. Note that the generating function $G_{X_n}(z)$ has the same form as the generating function of another random variable.

Let Y_k denote a Bernoulli random variable (see Appendix A) with parameter $p = 1/k$. Its generating function is

$$G_{Y_k}(z) = (1-p)z^0 + pz = \left(1 - \frac{1}{k}\right) + \frac{1}{k}\, z = \frac{k-1+z}{k} .$$

If W is the sum of the independent random variables $Y_2, Y_3, \ldots, Y_n$,

$$W = \sum_{k=2}^{n} Y_k,$$

then

$$G_W(z) = \prod_{k=2}^{n} G_{Y_k}(z) = \prod_{k=2}^{n} \frac{z+k-1}{k} = G_{X_n}(z)\,,$$

that is, X_n and W have the same distribution. Since the variables Y_k are independent,

$$\mathrm{var}[X_n] = \sum_{k=2}^{n} \mathrm{var}[Y_k] = \sum_{k=2}^{n} \frac{1}{k}\left(1 - \frac{1}{k}\right)\,.$$

To summarize: the program execution time is a random variable

$$S = S_1 + S_2 X_n,$$

where S_1 is a constant corresponding to the execution time of steps *1÷3* and *5÷6*, S_2 is the constant execution time of a single execution of step *4*, and X_n is a random variable whose generating function has been determined above.

■

For the characteristics of a service station, Kendall's notation is used [57]: the system is described by a sequence of symbols $A/B/c/K/H/Z$. The symbols placed in positions A and B denote the types of distributions governing the interarrival times of successive customers and the customer service times, respectively.

In particular: M denotes a Markovian (exponential) distribution, D a deterministic distribution, E_k an Erlang distribution of order k, H_k a hyperexponential distribution of order k, C a Cox distribution, and G an arbitrary distribution. (These distributions are described in Appendix A.)

The parameter c determines the number of parallel service channels, i.e. the maximum number of customers that can be served simultaneously. The parameter K specifies the queue-length limit: if there are already K customers at the station, subsequent arrivals are rejected. The parameter H denotes the population size, i.e. the number of customers that may request service. The final parameter, Z, specifies the queue discipline, e.g. *FIFO*, *LIFO*, *IS*, or *PS*.

If the source population is infinite, if there is no queue-length limit, or if the queue discipline is the default one, then H, K, or Z is omitted from the notation, respectively.

Let us recall a few basic concepts related to the Markov processes used later in this chapter.

A **stochastic process** $X(t)$ is a family of random variables defined on the same probability space and indexed by a parameter t (in our case, time). Thus, for each admissible value $t = t_i$ there is a distribution function

$$F_X(x, t_i) = P[X(t_i) \le x].$$

The values taken by the process form its state space, which may be continuous or discrete. If the state space is discrete, the process is called a *discrete-state process*. Similarly, the indexing parameter t may take continuous or discrete values; in these cases we speak of continuous-time or discrete-time processes, respectively. Discrete-time processes are called *chains*.

The statistical dependence between the values of $X(t)$ at different times should, in general, be described by the joint distribution function

$$\begin{aligned} F_{\mathbf{X}}(\mathbf{x};\mathbf{t}) &= F_{X_1,X_2,\ldots,X_n}(x_1,x_2,\ldots,x_n;t_1,t_2,\ldots,t_n) \\ &= P[X(t_1)\le x_1, X(t_2)\le x_2,\ldots,X(t_n)\le x_n] \end{aligned}$$

defined for all n and all admissible vectors

$$\mathbf{x}=(x_1,x_2,\ldots,x_n), \qquad \mathbf{t}=(t_1,t_2,\ldots,t_n).$$

In general, this is an impossible task, so one usually introduces additional assumptions about the process. For example, one may restrict attention to stationary processes, for which

$$F_{\mathbf{X}}(\mathbf{x};\mathbf{t}) = F_{\mathbf{X}}(\mathbf{x};\mathbf{t}+\tau),$$

where

$$\mathbf{t}+\tau=(t_1+\tau,t_2+\tau,\ldots,t_n+\tau).$$

In what follows, we will be interested in a particular class of processes: **Markov processes**, for which

$$\begin{aligned} &P\left[X(t)\le x \mid X(t_n)=x_n, X(t_{n-1})=x_{n-1},\ldots,X(t_0)=x_0\right] \\ &\quad = P[X(t)\le x \mid X(t_n)=x_n] \end{aligned}$$

for all $t>t_n>t_{n-1}>\ldots>t_0$.

This equation expresses the fundamental property of a Markov process: its future evolution depends only on its current state and not on the history of how that state was reached. This property is often called *memorylessness*.

In particular, we will deal with *Markov chains*, i.e. processes with a discrete state space and continuous time. We will also assume homogeneous processes, for which

$$P[X(t)\le x \mid X(t_n)=x_n]$$

depends only on the time difference $(t-t_n)$ and not on the specific values of t and t_n.

Let a continuous-time Markov chain take values from the set $\{x_1,x_2,\ldots,x_n,\ldots\}$. If $X(t)=x_i$, we say that the chain is in state i. Let us define

$$P_i(t)=P[X(t)=x_i] \qquad \text{and} \qquad p_{ij}(s,u)=P[X(u)=x_j \mid X(s)=x_i].$$

For a homogeneous Markov chain,

$$p_{ij}(t,t+\tau)=p_{ij}(\tau)\,.$$

For a very small time interval $\tau=\Delta t$, we assume that $p_{ij}(\tau)$ is approximately linear:

$$p_{ij}(\Delta t)=q_{ij}\Delta t\,.$$

The coefficient q_{ij} is called the transition rate (or transition intensity) from state i to state j. Multiplying it by the small interval Δt gives the approximate probability of transition from state i to state j during that interval.

The probability that the chain is in state i at time $t + \Delta t$ can therefore be written as

$$\begin{aligned} P_i(t+\Delta t) &= \sum_{j,\, j\neq i} P_j(t) q_{ji} \Delta t + P_i(t) \prod_{j,\, j\neq i} [1 - q_{ij}\Delta t] \\ &= \sum_{j,\, j\neq i} P_j(t) q_{ji} \Delta t + P_i(t) \left[1 - \sum_{j,\, j\neq i} q_{ij}\Delta t + O(\Delta t) \right] . \end{aligned}$$

After rearranging terms, we obtain

$$\frac{P_i(t+\Delta t) - P_i(t)}{\Delta t} = \sum_{j,\, j\neq i} P_j(t) q_{ji} - P_i(t) \sum_{j,\, j\neq i} q_{ij} \, .$$

Let us define

$$q_{ii} = - \sum_{j,\, j\neq i} q_{ij}.$$

Then, in the limit as $\Delta t \to 0$, we obtain

$$\frac{dP_i(t)}{dt} = \sum_j P_j(t) q_{ji}. \tag{3.5}$$

In matrix notation,

$$\frac{d\,\mathbf{P}(t)}{dt} = \mathbf{Q}^T \mathbf{P}(t) \, . \tag{3.6}$$

Here, $\mathbf{P}(t)$ is the state-probability vector, and the matrix $\mathbf{Q}$ is the transition-rate matrix (or intensity matrix), also called the *infinitesimal generator*.

The equations (3.5) are supplemented by the normalisation condition

$$\sum_j P_j(t) = 1.$$

In steady state, when the state probabilities do not depend on time, $\mathbf{P}(t) = \mathbf{P}$, equation (3.6) becomes

$$\mathbf{0} = \mathbf{Q}^T \mathbf{P} \, . \tag{3.7}$$

The dimensions of $\mathbf{Q}$ and $\mathbf{P}$ may be finite or infinite, depending on the number of states of the process. Note that the sum of the elements in each row of the matrix $\mathbf{Q}$ (or, equivalently, each column of $\mathbf{Q}^T$) is equal to zero.

The time spent in any state has an exponential distribution with parameter

$$\lambda_i = -q_{ii}.$$

The exponential distribution is the only continuous distribution with the memoryless property, i.e. the probability of remaining in a given state for an additional time interval t does not depend on the time t_1 already spent in that state:

$$\frac{\lambda_i e^{-\lambda_i (t_1+t)}}{\int_{t_1}^{\infty} \lambda_i e^{-\lambda_i t}\, dt} = \frac{\lambda_i e^{-\lambda_i (t_1+t)}}{e^{-\lambda_i t_1}} = \lambda_i e^{-\lambda_i t} \, . \tag{3.8}$$

Therefore, the transition rate out of a state does not depend on how long the process has already remained in that state.

3.2 Models $M(n)/M(n)/1$ of systems with FIFO service

Let us assume that customers arrive at the service station at intervals that are exponentially distributed random variables. The parameter λ of this distribution depends on the number n of customers already present in the station. The density function is

$$a(x) = \lambda(n)e^{-\lambda(n)x},$$

and the distribution function is

$$A(x) = 1 - e^{-\lambda(n)x}.$$

A process in which events occur at exponentially distributed intervals is called a Poisson process.

Let us also assume that service times are exponentially distributed, with density

$$b(x) = \mu(n)e^{-\mu(n)x}$$

and distribution function

$$B(x) = 1 - e^{-\mu(n)x},$$

where the parameter μ may also depend on the number of customers present in the station.

We denote this model by $M(n)/M(n)/1$ to emphasise the dependence of both λ and μ on n. The model with constant parameters λ and μ, independent of n, is denoted by $M/M/1$.

If the arrival process is Poisson and the service times are exponentially distributed, then the process

$$N(t),$$

whose value represents the number of customers in the station at time t, is a Markov chain.

Let

$$p(n, t; n_0) = P[N(t) = n \mid N(0) = n_0].$$

For simplicity, we omit the initial condition in the notation.

If at time $t + \Delta t$ there are no customers in the system, this means that at time t either:

- the system was empty and no customer arrived during the interval Δt; the probability of an arrival to an empty system during Δt is $\lambda(0)\Delta t$, so the probability that no customer arrives is
$$1 - \lambda(0)\Delta t;$$
- or there was one customer in the system and that customer's service was completed; the probability of service completion during Δt, when there is one customer in the system, is
$$\mu(1)\Delta t.$$

If at time $t + \Delta t$ there are $n \geq 1$ customers in the system, then at time t either:

- there were already n customers, no new customer arrived, and no customer departed; the probability of no arrival is
$$1 - \lambda(n)\Delta t,$$
and the probability of no departure is
$$1 - \mu(n)\Delta t;$$

- there were $n+1$ customers and one customer departed; the probability of this event is

$$\mu(n+1)\Delta t;$$

- there were $n-1$ customers and one new customer arrived; the probability of this event is

$$\lambda(n-1)\Delta t.$$

Therefore,

$$\begin{aligned} p(0,t+\Delta t) &= p(0,t)\left[1-\lambda(0)\Delta t\right]+p(1,t)\mu(1)\Delta t, \\ p(n,t+\Delta t) &= p(n,t)\left[1-\lambda(n)\Delta t\right]\left[1-\mu(n)\Delta t\right]+p(n+1,t)\mu(n+1)\Delta t \\ &\quad +p(n-1,t)\lambda(n-1)\Delta t, \qquad n\geq 1. \end{aligned} \tag{3.9}$$

Subtracting $p(n,t)$ from both sides, dividing by Δt, and letting $\Delta t \to 0$, we obtain

$$\begin{aligned} \frac{dp(0,t)}{dt} &= -p(0,t)\lambda(0)+p(1,t)\mu(1), \\ \frac{dp(n,t)}{dt} &= -p(n,t)\left[\lambda(n)+\mu(n)\right]+p(n+1,t)\mu(n+1) \\ &\quad +p(n-1,t)\lambda(n-1), \qquad n\geq 1. \end{aligned} \tag{3.10}$$

In matrix notation, the matrices $\mathbf{Q}^T$ and $\mathbf{P}$ have the form

$$\mathbf{Q}^T=\begin{bmatrix} -\lambda(0) & \mu(1) & 0 & 0 & 0 & 0 & \cdots \\ \lambda(0) & -\mu(1)-\lambda(1) & \mu(2) & 0 & 0 & 0 & \cdots \\ 0 & \lambda(1) & -\mu(2)-\lambda(2) & \mu(3) & 0 & 0 & \cdots \\ 0 & 0 & \lambda(2) & -\mu(3)-\lambda(3) & \mu(4) & 0 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

and

$$\mathbf{P}^T(t)=\begin{bmatrix} p(0,t) & p(1,t) & p(2,t) & p(3,t) & \cdots \end{bmatrix}.$$

In steady state,

$$\lim_{t\to\infty} p(n,t)=p(n),$$

and therefore

$$\frac{dp(n,t)}{dt}=0.$$

The differential equations become algebraic equations:

$$\begin{aligned} 0 &= -p(0)\lambda(0)+p(1)\mu(1), \\ 0 &= -p(n)\left[\lambda(n)+\mu(n)\right]+p(n+1)\mu(n+1)+p(n-1)\lambda(n-1), \\ &\quad n\geq 1. \end{aligned} \tag{3.11}$$

Since the state of the system is uniquely determined by the number of customers present, the state with n customers in the system will simply be called state n.

The graph in Fig. 3.2 illustrates the transitions between the states of the system according to equations (3.11). In this process, transitions are possible only between neighbouring states:

from state n one may move only to state $n-1$ or to state $n+1$. This type of process is called a *birth-death process*.

Equations (3.11) are equilibrium equations representing the balance of probability flows: in steady state, the probability flow entering state n,

$$p(n+1)\mu(n+1) + p(n-1)\lambda(n-1),$$

is equal to the probability flow leaving this state,

$$p(n)\big[\lambda(n) + \mu(n)\big].$$

By solving the recurrence relation (3.11) recursively, we obtain

$$\begin{aligned} p(1) &= p(0)\,\frac{\lambda(0)}{\mu(1)}\,, \\ p(2) &= p(0)\,\frac{\lambda(0)\lambda(1)}{\mu(1)\mu(2)}\,, \\ \dots &\quad \dots \\ p(n) &= p(0)\,\frac{\lambda(0)\cdots\lambda(n-1)}{\mu(1)\cdots\mu(n)}\,, \\ \dots &\quad \dots \end{aligned} \tag{3.12}$$

Taking into account the normalisation condition

$$\sum_{n=0}^{\infty} p(n) = 1,$$

we obtain

$$p(0) = \frac{1}{1 + \sum_{n=1}^{\infty} \prod_{i=0}^{n-1} \frac{\lambda(i)}{\mu(i+1)}}\,.$$

The condition for the existence of a steady-state solution is that the sum

$$\sum_{n=1}^{\infty} \prod_{i=0}^{n-1} \frac{\lambda(i)}{\mu(i+1)}$$

be finite; in that case, the probability that the system is empty is positive.

By specifying the functions $\lambda(n)$ and $\mu(n)$ in solution (3.12), we can analyse different types of service systems, as illustrated in the following subsections.

3.2.1 System $M/M/1$

Let us assume that the parameters of the distributions $A(x)$ and $B(x)$ do not depend on the number of customers in the queue:

$$\lambda(n) = \lambda, \qquad \mu(n) = \mu.$$

Then, from equation (3.12), we obtain

$$p(n) = p(0)\,\frac{\lambda(0)\cdots\lambda(n-1)}{\mu(1)\cdots\mu(n)} = p(0)\left(\frac{\lambda}{\mu}\right)^n\,.$$

Let us denote

$$\frac{\lambda}{\mu} = \varrho.$$

If $\varrho < 1$, then

$$\sum_{n=1}^{\infty} \prod_{i=0}^{n-1} \frac{\lambda(i)}{\mu(i+1)} = \sum_{n=1}^{\infty} \varrho^n = \frac{\varrho}{1-\varrho},$$

hence

$$p(0) = 1 - \varrho,$$

and therefore

$$p(n) = (1-\varrho)\varrho^n \,. \tag{3.13}$$

The probability that the system is busy is

$$P[N \geq 1] = \varrho,$$

so ϱ is the utilisation factor of the system:

$$U \equiv \varrho.$$

The mean number of customers at the station is

$$\begin{aligned} E[N] &= \sum_{n=0}^{\infty} n\, p(n) = \sum_{n=0}^{\infty} n(1-\varrho)\varrho^n = (1-\varrho)\varrho \sum_{n=1}^{\infty} n\varrho^{n-1} \\ &= (1-\varrho)\varrho \frac{\partial}{\partial \varrho} \sum_{n=1}^{\infty} \varrho^n = (1-\varrho)\varrho \frac{\partial}{\partial \varrho} \frac{\varrho}{1-\varrho} = \frac{\varrho}{1-\varrho} \,. \end{aligned} \tag{3.14}$$

In steady state, the stochastic process $N(t)$ becomes a random variable. To emphasise that the mean number of customers at the station is now computed as the expectation of a random variable with a given distribution, we use the notation $E[N]$ rather than $\bar{N}$, which denoted a value determined from measurements. Similarly, instead of the symbol $\bar{R}$ for the response time (mean time spent at the station), we use the symbol $E[R]$. The system throughput in steady state is

$$X = \frac{1}{E[A]} = \lambda.$$

The mean service time at the station, previously denoted by $\bar{B}$, is now written as

$$E[B] = \frac{1}{\mu}.$$

These symbols are commonly used in the literature.

The mean number of customers at the station is the sum of the mean number of customers waiting in the queue and the mean number of customers in service. The latter quantity has the value

$$E[N_B] = 0 \cdot p(0) + 1 \cdot \sum_{n=1}^{\infty} p(n) = \varrho,$$

and thus the mean number of customers in the queue is

$$E[K] = E[N] - \varrho = \frac{\varrho^2}{1-\varrho}.$$

The system response time and the waiting time in the queue can be calculated using Little's law, introduced in the previous operational chapter, but which can also be proved within the probabilistic model of a service station [75]. It holds under very general assumptions — in particular for the $G/G/c$ model, and therefore also for the $M/M/1$ model:

$$\begin{aligned} E[R] &= \frac{E[N]}{\lambda} = \frac{1/\mu}{1-\varrho}\,, \\ E[W] &= \frac{E[K]}{\lambda} = \frac{\varrho/\mu}{1-\varrho}\,. \end{aligned} \tag{3.15}$$

Of course,

$$E[R] = E[W] + \frac{1}{\mu}.$$

Note that the formulas for the mean queue length, mean number of customers in the system, mean waiting time, and mean response time all have the factor $1-\varrho$ in the denominator, which reflects the stability condition $\varrho < 1$ under which these formulas are valid. Also note the nonlinear dependence of these quantities on the system load. The same increment $\Delta\varrho$ in the load, when ϱ is close to one, causes a much greater deterioration of performance than when ϱ is close to zero.

Distribution of the response time in the $M/M/1$ system:
Let $f_R(x)$ denote the density function of the random variable R, the response time at the station, and let $f_{R_n}(x)$ denote the density of the response time under the condition that the arriving customer finds exactly n customers at the station. The response time is the sum of the service times of all customers already present in the system and the service time of the arriving customer. The density of a random variable that is the sum of independent random variables is the convolution of the corresponding densities. Thus, $f_{R_n}(x)$ is the $(n+1)$-fold convolution of the service-time density:

$$f_{R_n}(x) = \underbrace{b(x) * b(x) * \cdots * b(x)}_{(n+1)\ times} = \underbrace{\mu e^{-\mu x} * \mu e^{-\mu x} * \cdots * \mu e^{-\mu x}}_{(n+1)\ times}\,.$$

We apply the Laplace transform, computing

$$\overline{f}_{R_n}(s) = \int_0^\infty e^{-sx} f_{R_n}(x)\,dx.$$

The rules and properties of this transform are recalled in Appendix B; in particular, the transform of a convolution is equal to the product of the transforms of the convolved functions:

$$\overline{f}_{R_n}(s) = \left[\overline{b}(s)\right]^{n+1} = \left(\frac{\mu}{\mu+s}\right)^{n+1}\,.$$

The Laplace transform of

$$f_R(x) = \sum_{n=0}^{\infty} f_{R_n}(x) p(n)$$

is

$$\begin{aligned}\overline{f}_R(s) &= \sum_{n=0}^{\infty} \overline{f}_{R_n}(s)p(n) = \sum_{n=0}^{\infty}(1-\varrho)\varrho^n \left(\frac{\mu}{\mu+s}\right)^{n+1} \\ &= (1-\varrho)\frac{\mu}{\mu+s}\sum_{n=0}^{\infty}\left(\varrho\frac{\mu}{\mu+s}\right)^n \\ &= (1-\varrho)\frac{\mu}{\mu+s}\cdot\frac{1}{1-\varrho\mu/(\mu+s)} = \frac{\mu-\lambda}{\mu-\lambda+s}\,.\end{aligned}$$

Therefore, the response-time distribution is exponential with parameter $\mu-\lambda$:

$$f_R(x) = (\mu-\lambda)e^{-(\mu-\lambda)x}\,, \qquad F_R(x) = 1-e^{-(\mu-\lambda)x}\,.$$

The expected value of a random variable X with density $f_X(x)$ and Laplace transform $\overline{f}_X(s)$ can be calculated as

$$E[X] = -\left.\frac{d\,\overline{f}_X(s)}{ds}\right|_{s=0}\,,$$

cf. Appendix B. Hence the mean response time is

$$E[R] = -\left.\frac{d\,\overline{f}_R(s)}{ds}\right|_{s=0} = \frac{1}{\mu-\lambda}\,.$$

We now want to calculate the quantile $r_{q/100}$ of the qth percentile, i.e. the value such that

$$P[R \le r_{q/100}] = \frac{q}{100}.$$

Since

$$F_R(x) = 1-e^{-(\mu-\lambda)x},$$

we obtain

$$r_{q/100} = \frac{1}{\mu(1-\varrho)}\ln\left(\frac{100}{100-q}\right).$$

Distribution of the waiting time in the $M/M/1$ system:
Proceeding similarly to the definition of $f_R(x)$, let us denote by $f_{W_n}(x)$ the waiting-time distribution in the case where the arriving customer finds n customers ahead:

$$f_{W_n}(x) = \begin{cases} \delta(x) & \text{if } n=0, \\ \underbrace{b(x)*b(x)*\cdots*b(x)}_{n\ times} = \underbrace{\mu e^{-\mu x}*\mu e^{-\mu x}*\cdots*\mu e^{-\mu x}}_{n\ times} & \text{if } n\ge 1\,, \end{cases}$$

that is,

$$\overline{f}_{W_n}(s) = \begin{cases} 1 & \text{if } n=0, \\ \left(\dfrac{\mu}{\mu+s}\right)^n & \text{if } n\ge 1\,. \end{cases}$$

Hence,

$$\begin{aligned}\overline{f}_W(s) &= 1 \cdot p(0) + \sum_{n=1}^{\infty} \overline{f}_{W_n}(s)\, p(n) \\ &= (1-\varrho) + \sum_{n=1}^{\infty} (1-\varrho)\varrho^n \left(\frac{\mu}{\mu+s}\right)^n \\ &= 1 - \varrho + \varrho\, \frac{\mu-\lambda}{\mu-\lambda+s}\,,\end{aligned}$$

and, after inverting the transform,

$$f_W(x) = (1-\varrho)\delta(x) + \varrho(\mu-\lambda)e^{-(\mu-\lambda)x}\,,$$

where $\delta(x)$ is the Dirac delta distribution:

$$\delta(x) = 0 \text{ for } x \neq 0, \qquad \delta(x) = +\infty \text{ for } x = 0.$$

The cumulative distribution function of the waiting time is therefore

$$F_W(x) = \int_0^x f_W(u)\, du = 1 - \varrho e^{-(\mu-\lambda)x},$$

and the quantile $w_{q/100}$ has the value

$$w_{q/100} = \max\left\{0,\ \frac{1}{\mu(1-\varrho)} \ln\left(\frac{100\varrho}{100-q}\right)\right\}.$$

Output stream of the $M/M/1$ system:
Let $d(x)$ denote the distribution of the time between successive customers leaving the station.

If a departing customer leaves the station busy (which happens with probability ϱ), then the next departure occurs after a time equal to the service time of the next customer. If the departing customer leaves the station empty, then the time until the next departure is the sum of the waiting time for the next arrival and that customer's service time.

In the case of a Poisson input stream, because of the memoryless property of the exponential distribution, the distribution of the time until the next arrival is the same as the interarrival-time distribution $A(x)$. Therefore,

$$d(x) = \varrho\, b(x) + (1-\varrho)\, a(x) * b(x)\,. \tag{3.16}$$

Substituting

$$a(x) = \lambda e^{-\lambda x}, \qquad b(x) = \mu e^{-\mu x},$$

into (3.16), we obtain

$$d(x) = \lambda e^{-\lambda x}.$$

Thus, for the $M/M/1$ system,

$$d(x) \equiv a(x),$$

that is, the departure stream is again a Poisson stream with parameter λ [10].

Transient states in the $M/M/1$ system:
In the case where $\lambda(n) = \lambda$ and $\mu(n) = \mu$, the system of equations (3.10) has a known solution. Applying the Laplace transform to both sides of the system yields a system of algebraic equations from which one can derive the transform

$$\overline{p}(n, s)$$

of the probability $p(n, t)$, whose inverse transform is [12]

$$\begin{aligned} p(n,t) &= e^{-(\lambda+\mu)t}\left[\varrho^{\frac{n-i}{2}} I_{n-i}(at) + \varrho^{\frac{n-i-1}{2}} I_{n+i+1}(at)\right. \\ &\quad \left. +(1-\varrho)\varrho^n \sum_{j=n+i+2}^{\infty} \varrho^{-j/2} I_j(at)\right], \end{aligned} \tag{3.17}$$

where

$$a = 2\mu\sqrt{\varrho},$$

and $I_k(x)$ is the modified Bessel function of the first kind of order k:

$$I_k(x) = \sum_{m=0}^{\infty} \frac{\left(\frac{x}{2}\right)^{k+2m}}{(k+m)!\, m!}.$$

This solution corresponds to the initial condition

$$p(n, 0) = \delta_{in},$$

where δ_{in} is the Kronecker delta:

$$\delta_{in} = 1 \text{ for } i = n, \qquad \delta_{in} = 0 \text{ for } i \neq n,$$

that is, at time $t = 0$ there are exactly i customers in the system.

As can be seen, the analysis of transient states is not easy, even in the simplest case of the $M/M/1$ system. The computational difficulties become especially pronounced when the station load is high, because values of $p(n, t)$ for large n must then be taken into account.

Therefore, despite the existence of an exact solution, approximations are often used. For example, one may work with the Laplace transform of the distribution $p(n, t; i)$, obtaining expressions in the variable s whose inversion is simpler than (3.17) [66]. Bessel functions may also be replaced by functions that are easier to evaluate numerically [54].

If one is interested only in the evolution of the mean queue length in the nonstationary case, this quantity may be approximated by solving a first-order differential equation [103]. An analysis of flows in a larger computer network based on such an approximation is presented in [31].

3.2.2 System $M/M/c$

If the station is fed by a Poisson stream with parameter λ, has c parallel service channels, and in each channel the service time has an exponential distribution with parameter μ,

$$b(x) = \mu e^{-\mu x},$$

then we can use the solution (3.12), in which

$$\mu(n) = \begin{cases} n\mu & \text{for } n \le c, \\ c\mu & \text{for } n \ge c\,. \end{cases}$$

The graph of transitions between system states is shown in Fig. 3.3.

The distribution of the number of customers at the station has the form

$$p(n) = \begin{cases} p(0) \prod_{i=0}^{n-1} \frac{\lambda}{(i+1)\mu} = p(0) \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!} & \text{for } n \le c, \\ p(0) \left(\frac{\lambda}{\mu}\right)^n \frac{1}{c!} \frac{1}{c^{n-c}} & \text{for } n \ge c\,. \end{cases}$$

The normalisation condition determining the value of $p(0)$ is

$$p(0) = \left[1 + \sum_{n=1}^{c-1} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!} + \sum_{n=c}^{\infty} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{c!} \frac{1}{c^{n-c}}\right]^{-1},$$

that is,

$$p(0) = \left[1 + \sum_{n=1}^{c-1} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!} + \left(\frac{\lambda}{\mu}\right)^c \frac{1}{c!} \frac{1}{1 - \frac{\lambda}{c\mu}}\right]^{-1}.$$

The condition for the existence of a steady state and the validity of the above solution is

$$\frac{\lambda}{c\mu} < 1.$$

The probability p_w that an arriving task will have to wait is the probability that there are at least c tasks in the system:

$$p_w = \sum_{n=c}^{\infty} p(n) = p(0) \left(\frac{\lambda}{\mu}\right)^c \frac{1}{c!} \frac{1}{1 - \frac{\lambda}{c\mu}}\,.$$

This is known as the Erlang C formula.

We calculate the average queue length:

$$\begin{aligned} E[K] &= \sum_{n=c}^{\infty} (n-c) p(n) \\ &= p(0) \left(\frac{\lambda}{\mu}\right)^c \frac{1}{c!} \sum_{n=c}^{\infty} (n-c) \left(\frac{\lambda}{c\mu}\right)^{n-c} \\ &= \frac{p_w \left(\frac{\lambda}{c\mu}\right)}{1 - \frac{\lambda}{c\mu}}\,. \end{aligned}$$

The average number of tasks in service is

$$E[N_B] = \sum_{n=0}^{c-1} n\, p(n) + c \sum_{n=c}^{\infty} p(n) = \frac{\lambda}{\mu}\,.$$

The average number of tasks in the system is therefore

$$E[N] = E[K] + E[N_B]\,.$$

The average waiting time and response time are calculated using Little's law:

$$E[W] = \frac{E[K]}{\lambda}, \qquad E[R] = \frac{E[N]}{\lambda}\,.$$

The waiting-time distribution is

$$F_W(x) = 1 - p_w e^{-c\mu\left(1-\frac{\lambda}{c\mu}\right)x}\,.$$

In the case of an infinite number of channels, we obtain the $M/M/\infty$ model, in which

$$p(n) = p(0)\frac{\lambda^n}{\mu^n n!} \qquad \text{for } n \geq 1\,.$$

The normalisation condition has the form

$$p(0)\left[1 + \frac{\lambda}{\mu} + \frac{1}{2!}\left(\frac{\lambda}{\mu}\right)^2 + \cdots\right] = 1\,.$$

The series in brackets is the expansion of the function $e^{\lambda/\mu}$, hence

$$p(n) = e^{-\lambda/\mu}\frac{(\lambda/\mu)^n}{n!} \qquad \text{for } n \geq 0\,.$$

The expected number of customers in this system is

$$E[N] = \sum_{n=0}^{\infty} n\,p(n) = \frac{\lambda}{\mu}\,.$$

Using Little's law, we formally obtain the response time

$$E[R] = \frac{E[N]}{\lambda} = \frac{1}{\mu}\,,$$

which is equal to the average service time.

The average waiting time in the queue is zero. This model is therefore useful for representing pure service delay without queueing.

Example 3.2

There is one computer in the laboratory. During an 8-hour working day, it is used on average by 10 people. A single user session lasts, on average, 30 minutes. We assume that the number of laboratory employees is sufficiently large that successive requests can be treated as independent and that the intervals between requests can be approximated by an exponential distribution.

We wish to calculate the average waiting time for access to the station and to determine how this time changes if the laboratory purchases a second identical computer.

In the case of a single station, we use the $M/M/1$ model. Since

$$\frac{1}{\mu} = 30\,\text{min}, \qquad \frac{1}{\lambda} = \frac{8 \times 60\,\text{min}}{10} = 48\,\text{min}, \qquad \varrho = \frac{\lambda}{\mu} = \frac{5}{8},$$

we obtain

$$E[W] = \frac{\varrho/\mu}{1-\varrho} = \frac{\frac{5}{8} \times 30\,\text{min}}{1-\frac{5}{8}} = 50\,\text{min}\,.$$

In the case of two stations, we use the $M/M/2$ model. From the general solution we obtain

$$p(n) = \begin{cases} p(0) & \text{for } n = 0, \\ p(0)\dfrac{\lambda}{\mu} & \text{for } n = 1, \\ 2p(0)\left(\dfrac{\lambda}{2\mu}\right)^n & \text{for } n \geq 2\,, \end{cases}$$

where

$$p(0) = \left[1 + \frac{\lambda}{\mu} + 2\sum_{n=2}^{\infty}\left(\frac{\lambda}{2\mu}\right)^n\right]^{-1} = \frac{1-\lambda/(2\mu)}{1+\lambda/(2\mu)}\,.$$

The average waiting time is determined from Little's law applied to the queue:

$$E[W] = \frac{E[K]}{\lambda}\,.$$

Since only customers in excess of two must wait, we have

$$E[K] = \sum_{n=3}^{\infty}(n-2)p(n),$$

hence

$$E[W] = \frac{1}{\lambda}\sum_{n=3}^{\infty}(n-2)p(n) = \frac{1}{\lambda}\sum_{n=1}^{\infty} n \cdot 2p(0)\left(\frac{\lambda}{2\mu}\right)^{n+2} = \frac{2p(0)}{\lambda}\frac{\left(\frac{\lambda}{2\mu}\right)^3}{\left(1-\frac{\lambda}{2\mu}\right)^2}\,.$$

After substituting the data, we obtain

$$E[W] \approx 3.25\,\text{min},$$

so the purchase of a second station reduces the waiting time by more than a factor of 15. ■

3.2.3 System $M/M/1/K$

Customers arrive according to a Poisson process with constant rate λ. There is one service channel with exponentially distributed service time, and the number of customers in the system is limited to K. Whenever there are already K customers in the system, newly arriving requests are lost:

$$\begin{aligned} \lambda(n) &= \begin{cases} \lambda & \text{for } n < K, \\ 0 & \text{for } n \geq K\,, \end{cases} \\ \mu(n) &= \mu \qquad \text{for } n = 1, \ldots, K\,. \end{aligned}$$

Applying (3.12), we obtain

$$p(n) = p(0)\left(\frac{\lambda}{\mu}\right)^n,$$

where

$$p(0) = \left[1 + \sum_{n=1}^{K}\left(\frac{\lambda}{\mu}\right)^n\right]^{-1} = \frac{1-\frac{\lambda}{\mu}}{1-\left(\frac{\lambda}{\mu}\right)^{K+1}}\,.$$

In Fig. 3.4, we see the transition graph between the states of this system.

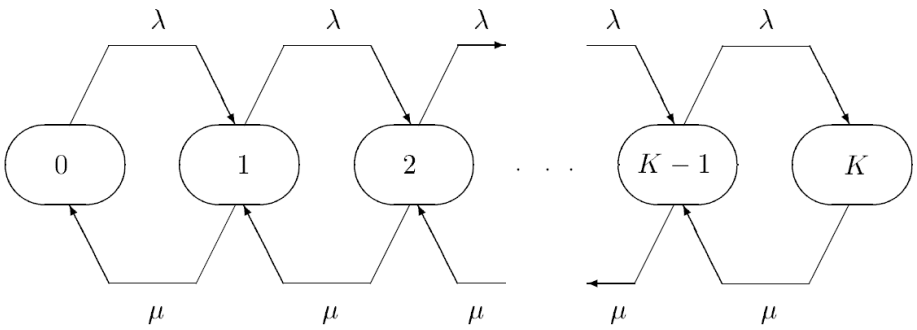


Figure 3.4: Transition graph of the $M/M/1/K$ system

The finite-capacity queueing model was originally introduced to describe the operation of a telephone exchange serving a limited population of K subscribers. In computer-system models, it is used whenever requests are stored in a buffer of limited capacity.

The mean time required to analyse and forward a packet at the interface between two networks is $1/\mu = 2$ ms. We assume that the distribution of this time is exponential. Packets arrive according to a Poisson process with intensity $\lambda = 100$ packets per second. The interface buffer can hold at most K packets. We wish to calculate the probability that the buffer is full and packets may be lost.

For the data

$$\lambda = 100, \qquad \mu = 500, \qquad \frac{\lambda}{\mu} = 0.2,$$

we calculate $p(K)$ as a function of the buffer capacity K:

K	2	3	4	5	10	15
$p(K)$	$3.17 \cdot 10^{-2}$	$6.39 \cdot 10^{-3}$	$1.28 \cdot 10^{-3}$	$2.56 \cdot 10^{-4}$	$8.19 \cdot 10^{-8}$	$2.62 \cdot 10^{-11}$

For $K = 2$ we obtain

$$p(0) = \frac{1-0.2}{1-(0.2)^3} = 0.806, \qquad p(1) = 0.2\,p(0) = 0.161, \qquad p(2) = 0.2\,p(1) = 0.032.$$

Hence

$$E[N] = \sum_{n=0}^{2} n\,p(n) = 0.225\,.$$

The effective arrival rate λ_{eff} (after deducting losses) and the response time are therefore

$$\lambda_{\mathit{eff}} = [1 - p(2)]\lambda = 0.968\,\lambda = 96.8\,1/\mathrm{s}, \qquad E[R] = \frac{E[N]}{\lambda_{\mathit{eff}}} = \frac{0.225}{96.8\,1/\mathrm{s}} = 2.33\,\mathrm{ms}.$$

For $K = \infty$ (infinite buffer, no losses),

$$E[N] = \frac{0.2}{1 - 0.2} = 0.25, \qquad E[R] = \frac{E[N]}{\lambda} = \frac{0.25}{100\,1/\mathrm{s}} = 2.5\ \mathrm{ms}.$$

3.2.4 System $M/M/1/H$

We assume that the customer population is finite and equal to H. After completing service, each customer remains outside the queue for a time determined by an exponential distribution with parameter λ, after which it returns and requests service again. The service time has an exponential distribution with parameter μ:

$$\begin{aligned} \lambda(n) &= \begin{cases} \lambda(H-n) & \text{for } n \le H, \\ 0 & \text{for } n > H\,, \end{cases} \\ \mu(n) &= \mu \qquad \text{for } n \ge 1\,. \end{aligned}$$

Then

$$p(n) = p(0) \prod_{i=0}^{n-1} \frac{\lambda(H-i)}{\mu} = p(0) \left(\frac{\lambda}{\mu}\right)^n \frac{H!}{(H-n)!}\,, \tag{3.18}$$

where

$$p(0) = \left[\sum_{n=0}^{H} \left(\frac{\lambda}{\mu}\right)^n \frac{H!}{(H-n)!}\right]^{-1}. \tag{3.19}$$

Fig. 3.5 shows the transitions between the states of this system.

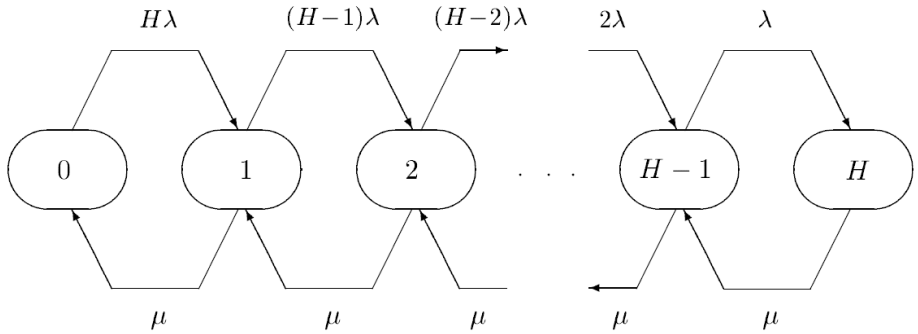


Figure 3.5: Transitions between the states of the $M/M/1//H$ system

The $M/M/1/H$ model is known in queueing theory as the machine-repair model: customers represent working machines, their time spent outside the queue is the time of trouble-free operation of the machine, the service station is a repair workshop, and the service time is the repair time.

Historically, this was the first queueing model used to analyse a computer system. A.L. Sherr [111] only changed the interpretation of the model: customers became active terminals generating transactions, and the service station became a computer system viewed as a whole. The results — the system load and its response time as a function of the number of users H — were sufficiently close to measurements ("It is surprising how many features that seem important for the operation of a real time-sharing computer system can be omitted in this model, which nevertheless correctly predicts the average response time of the system," wrote Sherr). This initiated the development and use of queueing models in the analysis of computer systems.

Four workstations form a local network. Each station runs independent programs, communicating occasionally with other stations. The local network is of the slotted ring type, which means that the communication bandwidth is divided equally among all stations that are transmitting messages at a given moment.

Let us assume that the interval between the end of one transmission and the beginning of the next transmission has an exponential distribution with mean $E[B_1] = 50$ *ms.*

The nominal bandwidth of the network is 10 Mbit/s, but protocol overhead reduces the effective bandwidth to 800 Kbit/s. The transmission time of a message can be approximated by an exponential distribution; the average message length is 1000 bytes.

What is the average message transmission time? What is the probability that at least two stations send messages simultaneously?

The average transmission time of a single message is

$$E[B_2] = \frac{1000 \times 8}{800 \times 10^3}\ \mathrm{s} = 10\ \mathrm{ms}.$$

Let the number of stations transmitting messages at a given moment determine the state of the Markov-process model. The model parameters are

$$\lambda = \frac{1}{E[B_1]} = 20\ 1/\mathrm{s}, \qquad \mu = \frac{1}{E[B_2]} = 100\ 1/\mathrm{s}.$$

Since the transmission time is proportional to the number of transmitting stations, the probability of completing one of the ongoing transmissions does not depend on their number. Therefore, we use the $M/M/1/4$ model rather than the $M/M/4/4$ model.

According to (3.19) and (3.18), we obtain:

n	0	1	2	3	4
$p(n)$	0.3983	0.3187	0.1912	0.0765	0.0153

The conditional probability that exactly $i = 1, \ldots, 4$ stations are transmitting, given that the network is active, is

$$p'(i) = \frac{p(i)}{1 - p(0)}\,.$$

Hence the average message transmission time is

$$E[B] = E[B_2] \sum_{n=1}^{4} n\, p'(n) = E[B_2] \times 1.6088 \approx 16.09\ \mathrm{ms}.$$

The probability that at least two stations are transmitting simultaneously is

$$P(n \geq 2) = \sum_{n=2}^{4} p(n) = 0.283\,.$$

■

3.2.5 System $M/M/c/K/H$

Let us now consider a combination of the previous models: there are c service channels, the number of customers in the system is limited to K, and the size of the finite customer population is H. We assume that

$$H \geq K > c.$$

Then

$$\lambda(n) = \begin{cases} \lambda(H-n) & \text{for } 0 \leq n \leq K-1, \\ 0 & \text{for } n \geq K\,, \end{cases}$$

$$\mu(n) = \begin{cases} n\mu & \text{for } 0 < n \leq c, \\ c\mu & \text{for } n \geq c\,. \end{cases}$$

Fig. 3.6 shows the states of this system and the transitions between them.

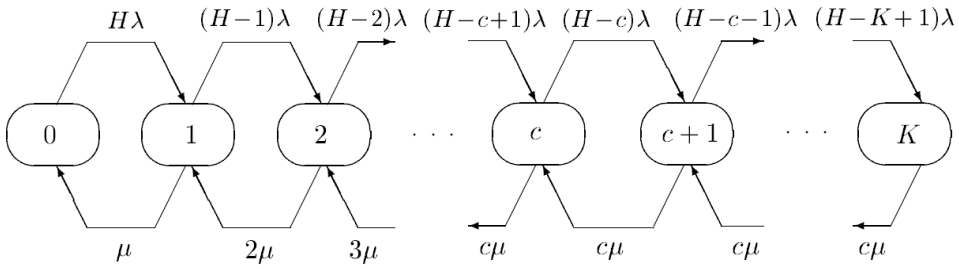


Figure 3.6: Transitions between the states of the $M/M/c/K/H$ system, $H \geq K > c$

Substituting the above expressions for $\lambda(n)$ and $\mu(n)$ into solution (3.12), we obtain

$$p(n) = p(0) \prod_{i=0}^{n-1} \frac{\lambda(H-i)}{(i+1)\mu} = p(0) \left(\frac{\lambda}{\mu}\right)^n \binom{H}{n} \qquad \text{for } 0 \leq n \leq c-1,$$

and

$$\begin{aligned} p(n) &= p(0) \prod_{i=0}^{c-1} \frac{\lambda(H-i)}{(i+1)\mu} \prod_{i=c}^{n-1} \frac{\lambda(H-i)}{c\mu} \\ &= p(0) \left(\frac{\lambda}{\mu}\right)^n \binom{H}{n} \frac{n!}{c!}\, c^{c-n} \qquad \text{for } c \leq n \leq K\,. \end{aligned}$$

As usual, the expression defining $p(0)$ is obtained from the normalisation condition.

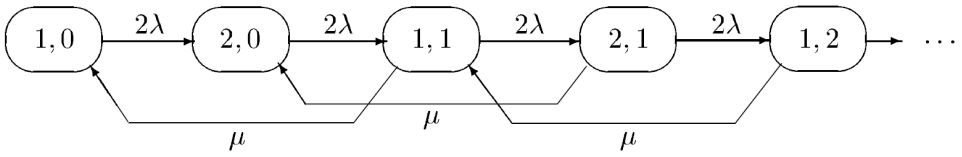
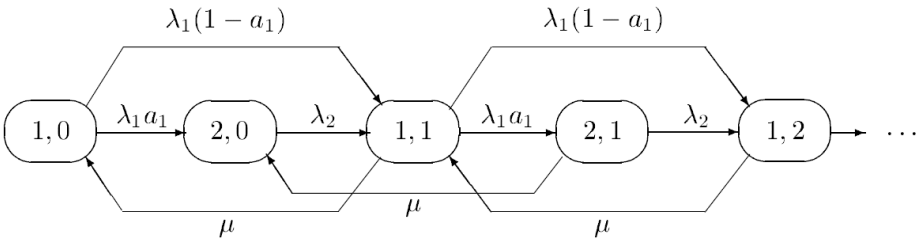


Figure 3.7: Transition graph between the states of the $E_2/M/1$ system



Rysunek 3.8: *Graf przejść między stanami systemu* $C_2/M/1$

Figure 3.8: Transition graph between the states of the $C_2/M/1$ system

3.3 Models deviating from the $M(n)/M(n)/1$ scheme

Figures 3.7 and 3.8 show transitions between the states of a station whose service-time distribution is not exponential, but Erlang or Cox. The state is described by the pair (n, f), where n is the number of tasks in the station and f is the phase of service of the currently executed task.

Figures 3.9 and 3.10 show the transitions between the states of a station for which the interarrival-time distribution is Erlang or Cox. In this case, the state is described by the pair (f, n).

The equations that can be written on the basis of these diagrams do not have, as in the case of the $M(n)/M(n)/1$ station, an analytical solution in the form of a closed formula for $p(n)$. They must therefore be solved numerically.

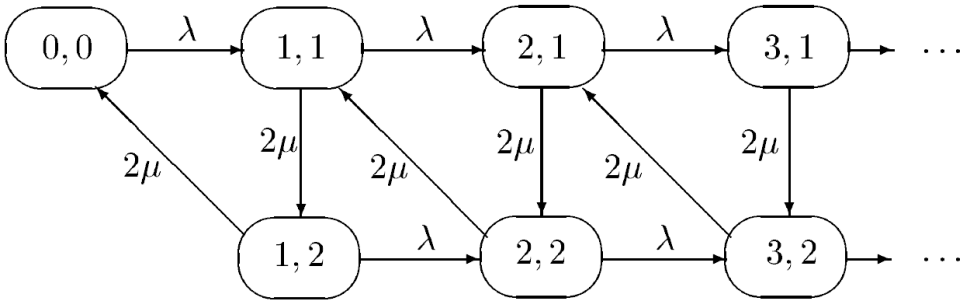


Figure 3.9: Transition graph between the states of the $M/E_2/1$ system

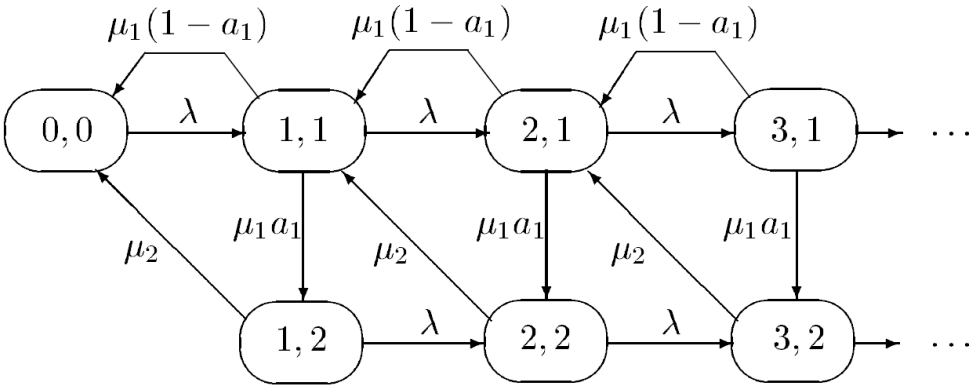


Figure 3.10: Transition graph between the states of the $M/C_2/1$ system

Chapter 4

Markov network models

4.1 Open queues

Let us assume that a queueing network consists of M stations. The service time at station i, $i = 1, \ldots, M$, follows an exponential distribution with parameter $\mu_i(n_i)$, where n_i is the number of customers present at that station. All stations use natural queueing discipline.

The route of customers through the network is determined by the transition probability matrix

$$\boldsymbol{R} = [r_{ij}]_{\substack{i=1,\ldots,M,\\ j=1,\ldots,M}},$$

where r_{ij} denotes the probability that, after being served at station i, a customer proceeds to station j.

The vector

$$\boldsymbol{r}_0 = [r_{01}, \ldots, r_{0M}]$$

defines the probabilities r_{0i} that a customer entering the network will first arrive at station i.

The probability

$$r_{i0} = 1 - \sum_{j=1}^{M} r_{ij}$$

is the probability that, after service at station i, the customer permanently leaves the network.

The external arrival stream is Poisson with parameter λ, which depends on the number

$$N = \sum_{i=1}^{M} n_i$$

of requests already present in the network, i.e.

$$\lambda = \lambda(N).$$

The state of the network is defined by the vector

$$\boldsymbol{n}(t) = [n_1(t), \ldots, n_M(t)].$$

Let us introduce the following notation for states neighbouring state $\boldsymbol{n}$.

State

$$\boldsymbol{n}^{j+} = [n_1, \ldots, n_{j-1}, n_j + 1, n_{j+1}, \ldots, n_M]$$

is the state in which, relative to state $\boldsymbol{n}$, there is one more customer at station j, while all remaining stations contain the same number of customers.

State

$$\boldsymbol{n}^{j-} = [n_1, \ldots, n_{j-1}, n_j - 1, n_{j+1}, \ldots, n_M]$$

is the state in which, relative to state $\boldsymbol{n}$, there is one fewer customer at station j, while all remaining stations contain the same number of customers.

State

$$\boldsymbol{n}^{j+,i-} = [n_1, \ldots, n_{j-1}, n_j + 1, n_{j+1}, \ldots, n_{i-1}, n_i - 1, n_{i+1}, \ldots, n_M]$$

is the state in which, relative to state $\boldsymbol{n}$, there is one more customer at station j, one fewer customer at station i, and the same number of customers at all remaining stations.

As in the case of a single station, we can write the equation describing the probability of state $\boldsymbol{n}$ at time $t + \Delta t$:

$$\begin{aligned} p(\boldsymbol{n}, t+\Delta t) &= p(\boldsymbol{n}, t)[1 - \lambda(N)\Delta t] \prod_{j=1}^{M} \{[1 - \mu_j(n_j)\Delta t](1 - r_{jj})\} \\ &\quad + \sum_{j=1}^{M} p(\boldsymbol{n}^{j+}, t)\mu_j(n_j + 1)r_{j0}\Delta t \\ &\quad + \sum_{j=1}^{M} p(\boldsymbol{n}^{j-}, t)\lambda(N-1)r_{0j}\Delta t \\ &\quad + \sum_{i=1}^{M} \sum_{\substack{j=1 \\ j\neq i}}^{M} p(\boldsymbol{n}^{j+,i-}, t)\mu_j(n_j + 1)r_{ji}\Delta t \; . \end{aligned}$$

This is an infinite system of equations, since the number of possible states $\boldsymbol{n}$ is infinite.

The terms on the right-hand side, written in successive lines, correspond to the following possibilities of reaching state $\boldsymbol{n}$ at time $t + \Delta t$:

- at time t the network was already in state $\boldsymbol{n}$ and nothing changed: no customer arrived, no service was completed at any station, or service was completed and the customer immediately returned to the same station;

- at time t the network was in state $\boldsymbol{n}^{j+}$, $j = 1, \ldots, M$, and during Δt the service of a customer at station j ended and that customer left the network;

- at time t the network was in state $\boldsymbol{n}^{j-}$, $j = 1, \ldots, M$, and during Δt one customer entered the network and proceeded to station j;

- at time t the network was in state $\boldsymbol{n}^{j+,i-}$, $j = 1, \ldots, M$, $i = 1, \ldots, M$, $i \neq j$, and during Δt service at station j ended and the customer moved to station i.

If a steady state exists, then from the above equations we obtain the system

$$\begin{aligned} 0 &= -p(\boldsymbol{n})\left\{\lambda(N)+\sum_{j=1}^{M}\mu_j(n_j)(1-r_{jj})\right\} \\ &\quad +\sum_{j=1}^{M}p(\boldsymbol{n}^{j+})\mu_j(n_j+1)r_{j0}+\sum_{j=1}^{M}p(\boldsymbol{n}^{j-})\lambda(N-1)r_{0j} \\ &\quad +\sum_{i=1}^{M}\sum_{\substack{j=1\\ j\neq i}}^{M}p(\boldsymbol{n}^{j+,i-})\mu_j(n_j+1)r_{ji}. \end{aligned}$$

After a minor rearrangement, this can be written as

$$\begin{aligned} 0 &= -p(\boldsymbol{n})\left[\lambda(N)+\sum_{j=1}^{M}\mu_j(n_j)\right] \\ &\quad +\sum_{j=1}^{M}p(\boldsymbol{n}^{j+})\mu_j(n_j+1)r_{j0}+\sum_{j=1}^{M}p(\boldsymbol{n}^{j-})\lambda(N-1)r_{0j} \\ &\quad +\sum_{i=1}^{M}\sum_{j=1}^{M}p(\boldsymbol{n}^{j+,i-})\mu_j(n_j+1)r_{ji}\ . \end{aligned} \tag{4.1}$$

The transitions between states described by equations (4.1) are illustrated in Fig. 4.1.

System (4.1) has the following solution [52]:

$$p(\boldsymbol{n})=\frac{1}{G}\,\Lambda(N)V(\boldsymbol{n})\ , \tag{4.2}$$

where

$$\begin{aligned} \Lambda(N) &= \prod_{m=0}^{N-1}\lambda(m)\ , \\ V(\boldsymbol{n}) &= \prod_{i=1}^{M}\prod_{m=1}^{n_i}\frac{V_i}{\mu_i(m)}\ . \end{aligned}$$

The elements of the vector

$$\boldsymbol{V}=[V_1,\ldots,V_M]$$

are the solutions of the traffic equations

$$V_i=r_{0i}+\sum_{j=1}^{M}V_jr_{ji}\ ,\qquad i=1,\ldots,M\,. \tag{4.3}$$

Here, V_i is the mean number of visits that a customer, once it has entered the network, makes to station i.

The normalisation constant G is

$$G=\sum_{N=0}^{\infty}H(N)\Lambda(N)\ , \tag{4.4}$$

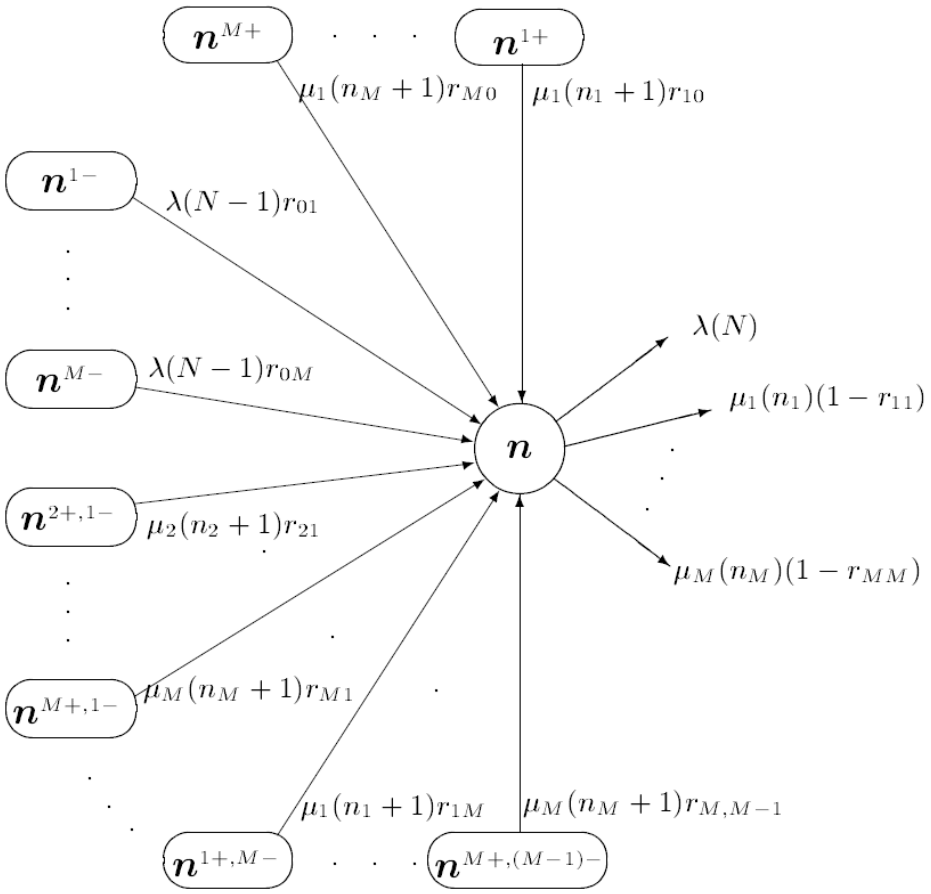


Figure 4.1: Ways of entering and leaving state described by equations (4.1)

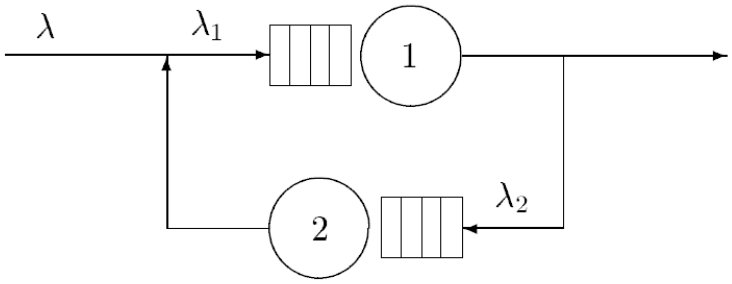


Figure 4.2: Two-node open network in Example 4.1

where

$$H(N) = \sum_{\boldsymbol{n}:\sum_{i=1}^{M} n_i = N} V(\boldsymbol{n}) \ .$$

A condition for the existence of a steady state is that the normalisation constant G be finite, i.e. that the series in equation (4.4) converge.

The correctness of solution (4.2) can be verified by substituting it into equations (4.1) and obtaining an identity.

Example 4.1

Let us use the above formulas to determine the state probabilities of the following system. The network in Fig. 4.2 consists of two service stations ($M = 2$), fed by a Poisson stream of customers with constant parameter $\lambda(N) = \lambda$. All external arrivals go first to station 1, so that $r_{01} = 1$ and $r_{02} = 0$. The parameters of the exponential service-time distributions at both stations are constant: $\mu_1(n_1) = \mu_1$ and $\mu_2(n_2) = \mu_2$. The transition probability matrix has the form:

$$\boldsymbol{R} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} = \begin{bmatrix} 0 & r_{12} \\ 1 & 0 \end{bmatrix} \ .$$

The state of the network is defined by the vector

$$\boldsymbol{n} = [n_1, n_2],$$

and the total number of customers in the network is

$$N = n_1 + n_2.$$

We solve the traffic equations:

$$\begin{aligned} V_1 &= r_{01} + V_1 r_{11} + V_2 r_{21} \ , \\ V_2 &= r_{02} + V_1 r_{12} + V_2 r_{22} \ , \end{aligned}$$

that is, for the given matrix $\boldsymbol{R}$,

$$V_1 = 1 + V_2 \ , \qquad V_2 = V_1 r_{12} \ ,$$

which gives

$$V_1 = \frac{1}{1 - r_{12}} , \qquad V_2 = \frac{r_{12}}{1 - r_{12}} .$$

Since the parameters λ, μ_1, and μ_2 are constants, the throughput of the network and of the individual stations does not depend on the state of the network. The total network throughput is λ, while the throughputs of the two stations are

$$\lambda_1 = V_1 \lambda , \qquad \lambda_2 = V_2 \lambda .$$

We then calculate

$$\begin{aligned} V(\boldsymbol{n}) &= \left(\frac{V_1}{\mu_1}\right)^{n_1} \left(\frac{V_2}{\mu_2}\right)^{n_2} , \\ \Lambda(N) &= \lambda^N = \lambda^{n_1+n_2} , \\ V(\boldsymbol{n})\Lambda(N) &= \left(\frac{V_1\lambda}{\mu_1}\right)^{n_1} \left(\frac{V_2\lambda}{\mu_2}\right)^{n_2} = \rho_1^{n_1}\rho_2^{n_2} , \end{aligned}$$

where

$$\rho_i = \frac{\lambda_i}{\mu_i} , \qquad i = 1, 2.$$

Next,

$$\begin{aligned} H(N) &= \sum_{n_1=0}^{N} V(n_1, N - n_1) \\ &= \sum_{n_1=0}^{N} \left(\frac{V_1}{\mu_1}\right)^{n_1} \left(\frac{V_2}{\mu_2}\right)^{N-n_1} , \end{aligned}$$

and

$$\begin{aligned} G &= \sum_{N=0}^{\infty} \sum_{n_1=0}^{N} \left(\frac{V_1}{\mu_1}\right)^{n_1} \left(\frac{V_2}{\mu_2}\right)^{N-n_1} \lambda^N \\ &= \sum_{N=0}^{\infty} \sum_{n_1=0}^{N} \left(\frac{V_1\lambda}{\mu_1}\right)^{n_1} \left(\frac{V_2\lambda}{\mu_2}\right)^{N-n_1} \\ &= \sum_{N=0}^{\infty} \sum_{n_1=0}^{N} \rho_1^{n_1} \rho_2^{N-n_1} \\ &= \sum_{n_1=0}^{\infty} \rho_1^{n_1} \sum_{n_2=0}^{\infty} \rho_2^{n_2} = \frac{1}{1-\rho_1} \frac{1}{1-\rho_2} . \end{aligned}$$

The last transformation is valid provided that

$$\rho_1 < 1 \qquad \text{and} \qquad \rho_2 < 1.$$

This is the condition for the existence of a steady state in the network under consideration.

The state probabilities are therefore given by

$$p(n_1, n_2) = \rho_1^{n_1}(1 - \rho_1)\, \rho_2^{n_2}(1 - \rho_2) ,$$

where

$$\rho_i = \frac{\lambda_i}{\mu_i}$$

is the utilisation factor of station i, $i = 1, 2$.

Thus, the joint probability that there are n_1 customers at the first station and n_2 customers at the second station can be written as the product of the probabilities for the two stations treated independently, each fed by a Poisson arrival stream with parameters λ_1 and λ_2 obtained from the traffic equations.

■

Similarly, for a network of M stations with arbitrary topology, we obtain

$$p(n_1, \ldots, n_M) = \prod_{i=1}^{M} \rho_i^{n_i}(1 - \rho_i) \ , \tag{4.5}$$

provided that $\lambda(N) = \lambda$ and $\mu_i(n_i) = \mu_i$ for all stations, and also provided that

$$\rho_i < 1 \ , \qquad i = 1, \ldots, M.$$

It should be noted that, in a network where customers may visit stations multiple times, the arrival and departure processes at the stations are generally not Poissonian, even though the form of solution (4.5) might suggest otherwise.

4.2 Closed networks

Let us assume, as before, that there are M stations in the network. The number N of customers is constant: no customer enters the network,

$$\lambda(N) = 0, \qquad r_{0i} = 0,$$

and no customer leaves it,

$$r_{i0} = 0, \qquad i = 1, \ldots, M.$$

The state of the network is determined by the same vector $\boldsymbol{n}$, and equations (4.1) now take the form

$$0 = -p(\boldsymbol{n}) \sum_{j=1}^{M} \mu_j(n_j) + \sum_{i=1}^{M} \sum_{j=1}^{M} p(\boldsymbol{n}^{j+,i-}) \mu_j(n_j + 1) r_{ji} \ . \tag{4.6}$$

The number of these equations, equal to the number of network states, is finite and equals

$$\binom{M + N - 1}{M - 1}.$$

For example, with 10 stations and 10 customers, this gives more than 92,000 states.

The solution of equations (4.6) is given by [41]

$$p(\boldsymbol{n}) = \frac{1}{G} V(\boldsymbol{n}) \ , \tag{4.7}$$

where

$$V(\boldsymbol{n}) = \prod_{i=1}^{M} \prod_{m=1}^{n_i} \frac{V_i}{\mu_i(m)} ,$$

$$G(N) = H(N) = \sum_{\boldsymbol{n}:\, \sum_{i=1}^{M} n_i = N} V(\boldsymbol{n}) . \tag{4.8}$$

Since in the traffic equations (4.3) the free term $r_{0i} = 0$ for all stations i, these equations have infinitely many solutions. The choice of the solution vector $\boldsymbol{V}$ is theoretically irrelevant, because any scaling factor applied to it appears with the same total power in the numerator and denominator of expression (4.7), so the value of $p(\boldsymbol{n})$ does not depend on that factor.

In practice, however, an inappropriate choice of the solution vector $\boldsymbol{V}$ may lead to numerical difficulties. Therefore, algorithms for calculating the normalisation constant take into account the possibility of rescaling the vector $\boldsymbol{V}$ during the computation.

4.2.1 Convolution algorithm for calculating the normalisation constant

Direct calculation of the normalisation constant by summing the terms $V(\boldsymbol{n})$ would be numerically very inefficient. Therefore, Buzen's algorithm [11] is used; it computes the constant G recursively.

Let us write $G = G(M, N)$ to emphasise that the value of the normalisation constant depends on the number of stations and the number of customers in the network. From the product form of the solution (4.7)–(4.8), it follows that the constant $G(M, N)$ can be expressed in terms of the normalisation constant for a network containing one fewer station:

$$G(M, N) = \sum_{n_i=0}^{N} \left[G_{-i}(M-1, N-n_i) \prod_{m=1}^{n_i} \frac{V_i}{\mu_i(m)} \right] ,$$

where G_{-i} denotes the normalisation constant for the network obtained by removing an arbitrarily chosen station i.

Let $\overset{*}{\bigcirc}$ denote the vector convolution operation. If

$$\boldsymbol{a} = [a_0, a_1, \ldots, a_R], \qquad \boldsymbol{b} = [b_0, b_1, \ldots, b_R],$$

and vector $\boldsymbol{c}$ is their convolution,

$$\boldsymbol{c} = \boldsymbol{a} \overset{*}{\bigcirc} \boldsymbol{b},$$

then

$$\boldsymbol{c} = [c_0, c_1, \ldots, c_R],$$

and its components are given by

$$c_i = \sum_{j=0}^{i} a_j b_{i-j}.$$

Let us denote

$$\gamma_i(n_i) = \prod_{m=1}^{n_i} \frac{V_i}{\mu_i(m)}$$

and introduce the $(N+1)$-dimensional vector

$$\mathbf{\Gamma}_i = [\gamma_i(0), \gamma_i(1), \ldots, \gamma_i(N)]. \tag{4.9}$$

We then define the following sequence of $(N+1)$-dimensional vectors:

$$\begin{aligned} \boldsymbol{G}_0 &= [1, 0, \ldots, 0] \,, \\ \boldsymbol{G}_i &= \boldsymbol{G}_{i-1} \circledast \mathbf{\Gamma}_i = \mathbf{\Gamma}_1 \circledast \mathbf{\Gamma}_2 \circledast \cdots \circledast \mathbf{\Gamma}_i \,. \end{aligned} \tag{4.10}$$

Let $G_i(r)$ denote the $(r+1)$st element of the vector $\boldsymbol{G}_i$. Note that this is the normalisation constant for a network consisting of i stations and containing r customers:

$$G_i(r) = G(i, r).$$

The constant $G(M, N)$ we seek is the last element of the last vector computed, namely the Nth element of $\boldsymbol{G}_M$:

$$G(M, N) = G_M(N).$$

The above procedure for computing the normalisation constant is called *the convolution algorithm.*

An additional advantage is that, besides reducing the computational effort required to determine the constant G, it also allows us to determine the marginal queue-length distributions at individual stations without first computing the full state probabilities $p(\boldsymbol{n})$.

For any chosen station M in the network, we have

$$\begin{aligned} p_M(n) &= \sum_{\boldsymbol{n}: n_M = n} p(\boldsymbol{n}) \\ &= \sum_{\boldsymbol{n}: n_M = n} \frac{1}{G(M,N)} \gamma_1(n_1)\gamma_2(n_2)\cdots\gamma_M(n) \\ &= \frac{\gamma_M(n)}{G(M,N)} \sum_{\boldsymbol{n}: \sum_{i=1}^{M-1} n_i = N-n} \gamma_1(n_1)\gamma_2(n_2)\cdots\gamma_{M-1}(n_{M-1}) \\ &= \frac{\gamma_M(n)}{G(M,N)} \, G_{-M}(M-1, N-n) \,, \end{aligned}$$

from which we can calculate the utilisation factor and the mean number of jobs:

$$\begin{aligned} \varrho_M &= 1 - p_M(0) \,, \\ E[n_M] &= \sum_{n_M=0}^{N} n_M \, p_M(n_M) \,, \end{aligned}$$

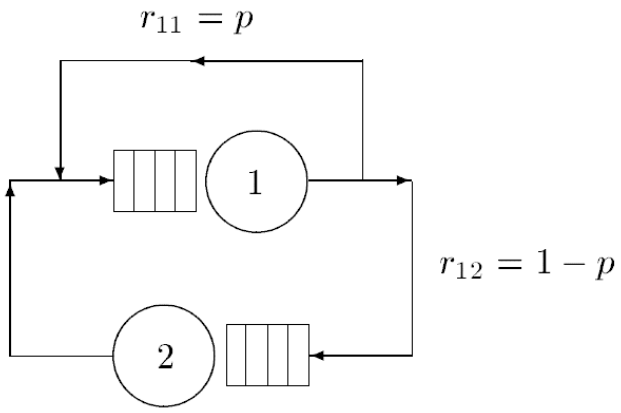


Figure 4.3: Two-node closed network in Example 4.2

as well as the throughput of station M:

$$\begin{aligned}
\lambda_M &= \sum_{n=1}^{N} p_M(n)\mu_M(n) \\
&= \sum_{n=1}^{N} \frac{G_{-M}(M-1,N-n)}{G(M,N)}\,\gamma_M(n)\mu_M(n) \\
&= \sum_{n=1}^{N} \frac{G_{-M}(M-1,N-n)}{G(M,N)}\,\gamma_M(n-1)V_M \\
&= \frac{V_M}{G(M,N)} \sum_{n=1}^{N} G_{-M}(M-1,N-n)\gamma_M(n-1) \\
&= V_M\,\frac{G(M,N-1)}{G(M,N)}\,.
\end{aligned}$$

Let us calculate the marginal distributions in a closed network containing two stations ($M = 2$) *with fixed parameters* $\mu_1(n_1) = \mu_1$ *and* $\mu_2(n_2) = \mu_2$, *in which 3 customers* ($N = 3$) *circulate. The transition probability matrix has the form, cf. Fig. 4.3:*

$$\boldsymbol{R} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} = \begin{bmatrix} p & 1-p \\ 1 & 0 \end{bmatrix}\,.$$

From the infinitely many solutions of the traffic equations

$$\begin{aligned}
V_1 &= V_1 p + V_2\,, \\
V_2 &= V_1(1-p)\,,
\end{aligned}$$

let us choose the solution

$$V_1 = 1\,, \qquad V_2 = 1-p\,.$$

According to (4.7), the probabilities of states in the network are

$$p(\boldsymbol{n}) = \frac{1}{G}V(\boldsymbol{n}) = \frac{1}{G(2,3)}\,\gamma_1(n_1)\gamma_2(n_2) = \frac{1}{G(2,3)}\left(\frac{V_1}{\mu_1}\right)^{n_1}\left(\frac{V_2}{\mu_2}\right)^{n_2}\,. \tag{4.11}$$

We form the vectors

$$\begin{aligned}\boldsymbol{\Gamma}_1 &= [\gamma_1(0), \gamma_1(1), \gamma_1(2), \gamma_1(3)] \\ &= \left[1, \frac{V_1}{\mu_1}, \left(\frac{V_1}{\mu_1}\right)^2, \left(\frac{V_1}{\mu_1}\right)^3\right] , \\ \boldsymbol{\Gamma}_2 &= [\gamma_2(0), \gamma_2(1), \gamma_2(2), \gamma_2(3)] \\ &= \left[1, \frac{V_2}{\mu_2}, \left(\frac{V_2}{\mu_2}\right)^2, \left(\frac{V_2}{\mu_2}\right)^3\right] .\end{aligned}$$

Since $\boldsymbol{G}_1 = \boldsymbol{\Gamma}_1$, we have

$$G_1(0) = \gamma_1(0), \qquad G_1(1) = \gamma_1(1), \qquad G_1(2) = \gamma_1(2), \qquad G_1(3) = \gamma_1(3).$$

Next, we calculate

$$\boldsymbol{G}_2 = \boldsymbol{\Gamma}_1 \circledast \boldsymbol{\Gamma}_2 = [G_2(0), G_2(1), G_2(2), G_2(3)] ,$$

where

$$\begin{aligned}G_2(0) &= 1 , \\ G_2(1) &= \frac{V_1}{\mu_1} + \frac{V_2}{\mu_2} , \\ G_2(2) &= \left(\frac{V_1}{\mu_1}\right)^2 + \frac{V_1}{\mu_1}\frac{V_2}{\mu_2} + \left(\frac{V_2}{\mu_2}\right)^2 , \\ G_2(3) &= \left(\frac{V_1}{\mu_1}\right)^3 + \left(\frac{V_1}{\mu_1}\right)^2 \frac{V_2}{\mu_2} + \frac{V_1}{\mu_1}\left(\frac{V_2}{\mu_2}\right)^2 + \left(\frac{V_2}{\mu_2}\right)^3 .\end{aligned}$$

For the second station, the distribution of the number of customers is

$$p_2(0) = \gamma_2(0)\frac{G_1(3)}{G_2(3)} , \qquad p_2(1) = \gamma_2(1)\frac{G_1(2)}{G_2(3)} ,$$
$$p_2(2) = \gamma_2(2)\frac{G_1(1)}{G_2(3)} , \qquad p_2(3) = \gamma_2(3)\frac{G_1(0)}{G_2(3)} .$$

Its throughput has the value

$$\lambda_2 = V_2 \frac{G_2(2)}{G_2(3)} .$$

To determine $p_1(n_1)$, we change the order of the calculations; now the first station is treated as the last one:

$$\begin{aligned}\boldsymbol{G}_1 &= \boldsymbol{\Gamma}_2 , \\ \boldsymbol{G}_2 &= \boldsymbol{\Gamma}_2 \circledast \boldsymbol{\Gamma}_1 ,\end{aligned}$$

and then obtain

$$p_1(0) = \gamma_1(0)\frac{G_1(3)}{G_2(3)} , \qquad p_1(1) = \gamma_1(1)\frac{G_1(2)}{G_2(3)} ,$$
$$p_1(2) = \gamma_1(2)\frac{G_1(1)}{G_2(3)} , \qquad p_1(3) = \gamma_1(3)\frac{G_1(0)}{G_2(3)} .$$

The throughput of the first station is

$$\lambda_1 = V_1 \frac{G_2(2)}{G_2(3)} .$$

■

Let us discuss, using a simple example, the **technique of network aggregation**: a portion of the network is replaced by a single node with a service rate $\mu(n)$ selected such that, depending on the number of tasks in the queue, its throughput is the same as that of the subnetwork it replaces. We therefore calculate the throughput of the subnetwork $\lambda(n)$ as a function of the number of tasks it contains and assume $\mu(n) = \lambda(n)$. This substitution is imperceptible to the rest of the network. This procedure does not introduce any error in the case of the Markov networks discussed in this chapter.

Fig. 4.4a illustrates the model under consideration. Let us assume that the average service times are $S_1 = 10$ *ms,* $S_2 = 100$ *ms, and* $S_3 = 25$ *ms, and that the service time distributions are exponential with parameters* $\mu_i = 1/S_i$, $i = 1, 2, 3$. *The transition probabilities have values* $r_{12} = 0.2$ *and* $r_{13} = 0.8$. *Using the aggregation technique, we wish to examine the utilisation of the central processing unit* ϱ_1 *as a function of the level of multiprogramming (the number of clients in the network),* N.

We replace stations 2 and 3 with a single service station having the same throughput as the subnetwork they form — Fig. 4.4b. To this end, we first determine the throughput $\lambda(n)$ of the selected subnetwork, treated independently of the whole network (n is the number of clients in that subnetwork). With the configuration shown in the figure, the throughput $\lambda(n)$ is equal to the sum of the throughputs of stations 2 and 3:

$$\lambda(n) = \lambda_2(n) + \lambda_3(n),$$

which we calculate as

$$\lambda_2(n) = \mu_2 \varrho_2(n) = \mu_2 \Big[1 - p(0,n)\Big] \qquad \text{and} \qquad \lambda_3(n) = \mu_3 \varrho_3(n) = \mu_3 \Big[1 - p(n,0)\Big],$$

where $p(0,n)$ and $p(n,0)$ are, respectively, the probabilities that station 2 is empty or station 3 is empty, calculated according to formulas (4.7) and (4.8) for the subnetwork under consideration, viewed as an independent closed network:

$$p(0,n) = \frac{1}{G(n)} \left(\frac{V_2}{\mu_2}\right)^0 \left(\frac{V_3}{\mu_3}\right)^n , \qquad p(n,0) = \frac{1}{G(n)} \left(\frac{V_2}{\mu_2}\right)^n \left(\frac{V_3}{\mu_3}\right)^0 .$$

The visit counts V_2 and V_3, as always in a closed network, are not uniquely determined; we only know that — in accordance with the ratio r_{12}/r_{13} — the proportion $V_2/V_3 = 1/4$. To simplify the calculations, let us choose $V_2 = 10$ and $V_3 = 40$, so that

$$\frac{V_2}{\mu_2} = \frac{V_3}{\mu_3} = 1 .$$

We calculate, for successive values of n — the number of customers in the subnetwork:

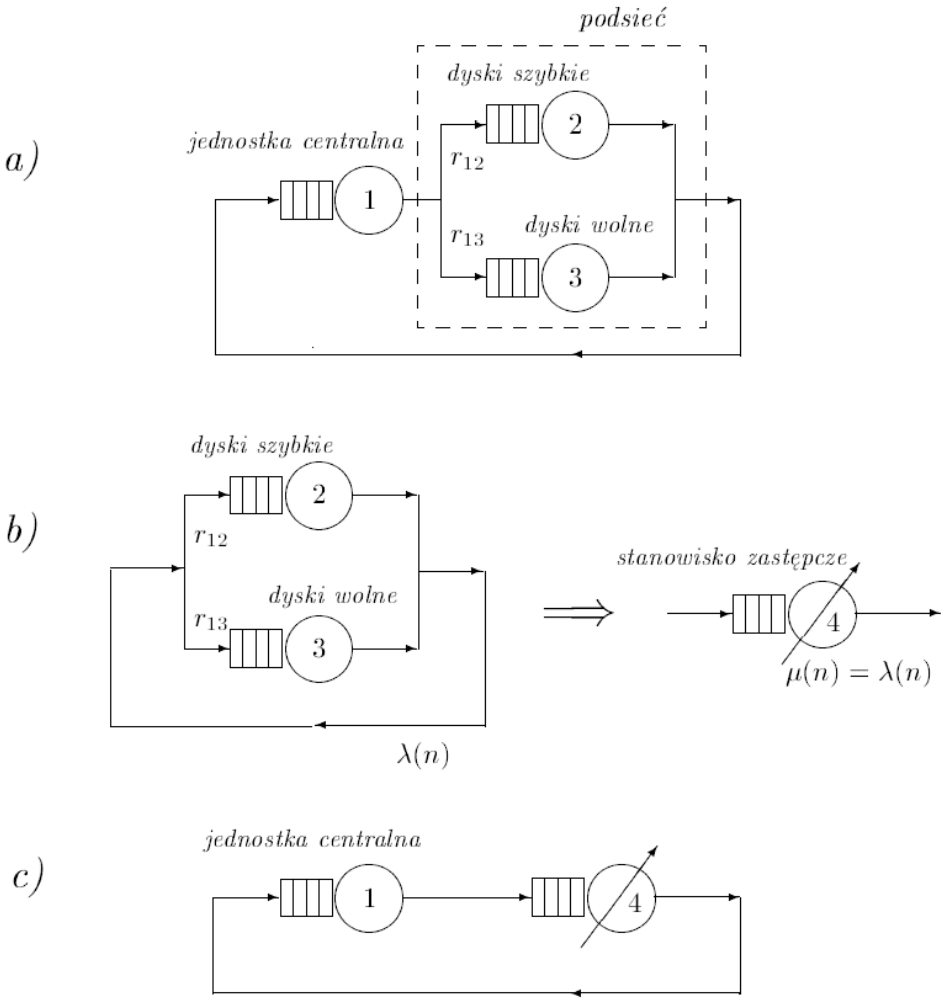


Figure 4.4: Subnetwork aggregation in Example 4.3

- $n = 1$:
$$G(1) = \left(\frac{V_2}{\mu_2}\right)^0 \left(\frac{V_3}{\mu_3}\right)^1 + \left(\frac{V_2}{\mu_2}\right)^1 \left(\frac{V_3}{\mu_3}\right)^0 = 1 + 1 = 2,$$
i.e.
$$p(0,1) = p(1,0) = 0.5.$$

- $n = 2$:
$$G(2) = \left(\frac{V_2}{\mu_2}\right)^0 \left(\frac{V_3}{\mu_3}\right)^2 + \left(\frac{V_2}{\mu_2}\right)^1 \left(\frac{V_3}{\mu_3}\right)^1 + \left(\frac{V_2}{\mu_2}\right)^2 \left(\frac{V_3}{\mu_3}\right)^0 = 1 + 1 + 1 = 3,$$
that is,
$$p(0,2) = p(2,0) = \frac{1}{3}.$$

- Similarly, for any n:
$$G(n) = n + 1,$$
that is,
$$p(0,n) = p(n,0) = \frac{1}{n+1}.$$
The utilisations of the stations are therefore
$$\varrho_2(n) = \varrho_3(n) = 1 - \frac{1}{n+1} = \frac{n}{n+1},$$
and the throughput of the subnetwork is
$$\lambda(n) = \varrho_2(n)\mu_2 + \varrho_3(n)\mu_3 = \frac{n}{n+1}\left(10\frac{1}{s} + 40\frac{1}{s}\right) = 50\frac{n}{n+1}\frac{1}{s}\,.$$

We now return to the basic network, consisting of a central unit and an aggregated station — Fig. 4.4c. Both stations have exponential service time distributions, where
$$\mu_1 = 100\,\frac{1}{s}, \qquad \mu_4(n) = 50\frac{n}{n+1}\,\frac{1}{s}.$$

We calculate the state probabilities in this network according to formula (4.7):
$$p(n_1, n_4) = \frac{1}{G(n_1+n_4)} \left(\frac{V_1}{\mu_1}\right)^{n_1} \frac{V_4}{\mu_4(1)} \cdots \frac{V_4}{\mu_4(n_4)}\,.$$
Let us denote
$$\gamma_4(n) = \frac{V_4}{\mu_4(1)} \cdots \frac{V_4}{\mu_4(n)} = 2^n(n+1),$$
and let us calculate, for various values of $N = n_1 + n_4$, the number of customers in the network:

- $N = 1$:
$$G(1) = \frac{V_1}{\mu_1} + \gamma_4(1) = 1 + 2(1+1) = 5.$$

- $N = 2$:
$$G(2) = G(1) + \gamma_4(2) = 5 + 2^2(2+1) = 17.$$

- $N = 3$:
$$G(3) = G(2) + \gamma_4(3) = 17 + 2^3(3 + 1) = 49.$$
- $N = n$:
$$G(n) = G(n - 1) + \gamma_4(n) = n\, 2^{n+1} + 1.$$

The utilisation of the central unit with N clients present in the network is therefore

$$\varrho_1(N) = 1 - p(0, N) = 1 - \frac{\gamma_4(N)}{G(N)} = \frac{G(N - 1)}{G(N)}\ .$$

The values of the normalisation constant and the CPU utilisation ϱ_1 for $N = 1, 2, 3, 4$ are as follows:

- $N = 1$:
$$G(1) = 5, \qquad \varrho_1(1) = \frac{1}{5} = 0.2.$$
- $N = 2$:
$$G(2) = 17, \qquad \varrho_1(2) = \frac{5}{17} = 0.2941.$$
- $N = 3$:
$$G(3) = 49, \qquad \varrho_1(3) = \frac{17}{49} = 0.3469.$$
- $N = 4$:
$$G(4) = 129, \qquad \varrho_1(4) = \frac{49}{129} = 0.3798.$$

Chapter 5

General Product Form Model

5.1 Product-Form Solutions and Local Equilibrium

Let us recall the description of a single service station with an exponential distribution of interarrival times and an exponential distribution of service times. Equations (3.11) represent the probability balance for state n:

$$\begin{aligned} 0 &= -p(n)[\lambda(n)+\mu(n)] + p(n+1)\mu(n+1) + \\ &\quad p(n-1)\lambda(n-1)\,, \quad n \geq 0, \end{aligned}$$

illustrated in Fig. 3.2. This equation represents the *global balance* of probabilities: it takes into account all possibilities of entering and leaving state n. The above equation can be decomposed into two equations:

$$\begin{aligned} 0 &= -p(n)\mu(n) + p(n-1)\lambda(n-1)\,, \\ 0 &= -p(n)\lambda(n) + p(n+1)\mu(n+1)\,, \quad n \geq 0, \end{aligned}$$

which balance the probability flows between neighbouring states, across the dashed partitions marked in Fig. 5.1.

Similarly, for the networks considered in the previous chapter, both open and closed, one can distinguish between global and local equilibrium. Figure 4.1 illustrates all paths leading into state $\boldsymbol{n}$ and out of that state. All these possibilities are taken into account in equations (4.1), which represent the *global equilibrium balance* and which can be decomposed into a larger number of simpler *local equilibrium* equations. These represent the balance of probabilities for entering and leaving a state associated with the arrival and departure of the same customer at a single station, while the other customers in the network remain stationary.

Let us examine global and local balance equations using the example of the simple network shown in Fig. 5.2.

All states of this network are shown in Fig. 5.3, and the global equilibrium equations are listed below.

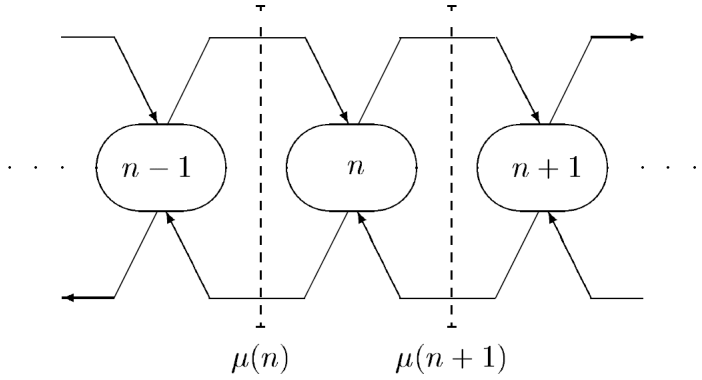


Figure 5.1: Balance of probability flows between neighbouring states of a system with exponential service times and a Poisson arrival stream

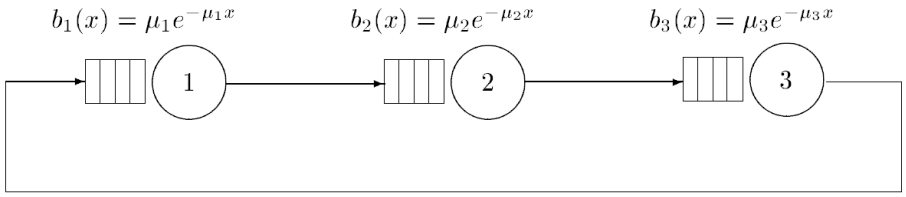


Figure 5.2: A simple closed network in local equilibrium

$$
\begin{aligned}
0 &= -p(2,0,0)\mu_1 + p(1,0,1)\mu_3 \,, && (5.1)\\
0 &= -p(0,2,0)\mu_2 + p(1,1,0)\mu_1 \,, && (5.2)\\
0 &= -p(0,0,2)\mu_3 + p(0,1,1)\mu_2 \,, && (5.3)\\
0 &= -p(1,1,0)(\mu_1+\mu_2) + p(2,0,0)\mu_1 + p(0,1,1)\mu_3 \,, && (5.4)\\
0 &= -p(1,0,1)(\mu_1+\mu_3) + p(0,0,2)\mu_3 + p(1,1,0)\mu_2 \,, && (5.5)\\
0 &= -p(0,1,1)(\mu_2+\mu_3) + p(1,0,1)\mu_1 + p(0,2,0)\mu_2 \,. && (5.6)
\end{aligned}
$$

The above six equations are not independent, because the sum of all probabilities is fixed. One of the equations is therefore linearly dependent on the others. It is replaced by the normalisation condition stating that the sum of the state probabilities is equal to one.

Equations (5.1)–(5.3) are simultaneously local and global equilibrium equations. Equations (5.4)–(5.6) are global equilibrium equations only. For example, equation (5.4) can be decomposed into

$$0 = p(0,1,1)\mu_3 - p(1,1,0)\mu_1 \,,$$

which balances the probability flows associated with the departure of a customer from the first station and the arrival of that customer at the third station, and

$$0 = -p(1,1,0)\mu_2 + p(2,0,0)\mu_1 \,,$$

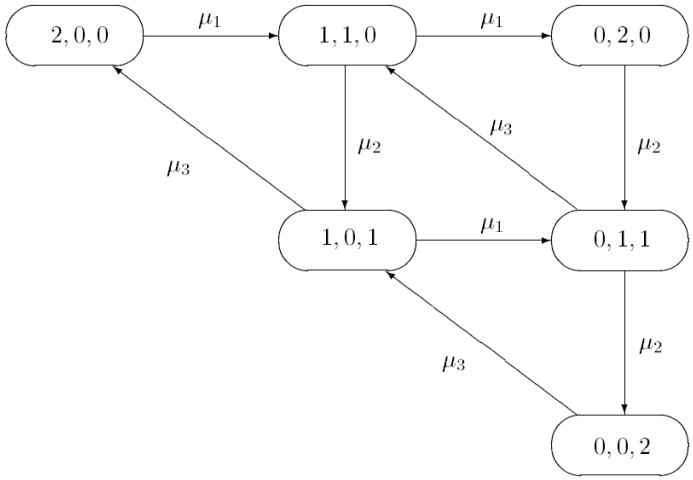


Figure 5.3: State-transition diagram

which balances the probability flows associated with the departure of a customer from the second station and the arrival of that customer at the first station.

From the local equilibrium equations, we can select five independent equations, supplement them with the normalisation condition, and obtain the same solution as from the global balance equations.

However, networks that remain in local equilibrium have a product-form solution, which, according to formulas (4.7) and (4.8), can be written immediately as

$$p(\boldsymbol{n}) = p(n_1, n_2, n_3) = \frac{1}{G} V(n_1, n_2, n_3) = \frac{1}{G} \left(\frac{1}{\mu_1}\right)^{n_1} \left(\frac{1}{\mu_2}\right)^{n_2} \left(\frac{1}{\mu_3}\right)^{n_3} , \tag{5.7}$$

where

$$\begin{aligned} G(N) &= \sum_{\boldsymbol{n}:\sum_{i=1}^{3} n_i = 2} V(n_1, n_2, n_3) \\ &= \left(\frac{1}{\mu_1}\right)^2 + \left(\frac{1}{\mu_2}\right)^2 + \left(\frac{1}{\mu_3}\right)^2 \\ &\quad + \left(\frac{1}{\mu_1}\right)\left(\frac{1}{\mu_2}\right) + \left(\frac{1}{\mu_1}\right)\left(\frac{1}{\mu_3}\right) + \left(\frac{1}{\mu_2}\right)\left(\frac{1}{\mu_3}\right) . \end{aligned}$$

5.2 Some Station Types in Local Equilibrium

Among Markov models represented as networks of states, there is a class of models whose states remain not only in global equilibrium, but also in local equilibrium, which implies the existence of a solution to the balance equations in the form

$$p(\boldsymbol{n}) = \frac{1}{G} \prod_{i=1}^{M} h_i(n_i) , \tag{5.8}$$

where the form of the function $h_i(n_i)$ depends only on the nature of station i.

Networks having a solution of the form (5.8) are called *decomposable (separable) networks* or *networks with a product-form solution*. This class of models includes the queueing systems discussed previously with Poisson arrival streams and exponential service times, as well as networks built from such stations.

Below, we introduce other types of stations that maintain local equilibrium and therefore can be combined into a network with a product-form solution. For the remaining Markov models, which do not satisfy the local equilibrium condition, the global balance equations must be solved numerically.

Four types of stations that remain in local equilibrium, selected from a larger set of theoretically possible stations [89], were incorporated into the BCMP model, the most general classical analytical queueing model with a product-form solution. The model name comes from the initials of its creators: Baskett, Chandy, Muntz, and Palacios [6].

Since the stations described below form part of the network discussed in the following subsection, we now introduce the station index i into the formulas, referring to station i in the network. We assume that all stations serve K classes of customers, and the superscript k in all notation refers to the k-th customer class.

(i) *FIFO* station with a first-in-first-out service discipline. Customers served at this station belong to K classes, but all classes have identical service-time distributions, characterised by the parameter $\mu_i^{(k)} = \mu_i$, $k = 1, \ldots, K$. Customer class membership determines only the subsequent path of the customer through the network.

The state of the station is defined by the vector

$$\boldsymbol{\nu}_i = [k_{i1}, k_{i2}, \ldots, k_{in_i}],$$

where n_i is the number of customers at station i, and k_{ij} is the class of the j-th customer in the queue.

In the remaining types of stations, the service time follows a Cox distribution, which may differ for each customer class. The Cox distribution, discussed in more detail in Appendix **??**, consists of an arbitrary number F of exponential phases, of which the first is always mandatory, while each subsequent phase occurs with a probability determined by the distribution parameters. Phase F is always the final phase. Using the Cox distribution, one can approximate almost any service-time distribution, which is why it is referred to as a quasi-general distribution.

Let us introduce the following notation:

$F_i^{(k)}$ — the maximum number of phases of the Cox distribution at station i when serving customers of class k,

$\mu_{if}^{(k)}$ — the exponential service-rate parameter of phase f when serving a class-k customer at station i,

$a_{if}^{(k)}$ — the probability that, after completion of phase f of service for a class-k customer at station i, service continues to the next phase,

$A_{if}^{(k)} = \prod_{j=1}^{f} a_{ij}^{(k)}$ — the probability that there will be more than f service phases for the customer,

$b_{if}^{(k)} = 1 - a_{if}^{(k)}$ — the probability that phase f is the final phase of service.

Stations with Coxian service-time distributions remain in local equilibrium only if customer service begins immediately upon arrival. Therefore, the remaining queueing disciplines in the BCMP model are:

(ii) *IS*-type station (see p. 37), having a sufficient number of identical parallel service channels to serve all customers simultaneously. Customers of class k are served with service-time distribution $B_i^{(k)}(x)$.

The state of this station is determined by the vector

$$\boldsymbol{\nu}_i = [\boldsymbol{\nu}_i^{(1)}, \boldsymbol{\nu}_i^{(2)}, \ldots, \boldsymbol{\nu}_i^{(K)}],$$

where $\boldsymbol{\nu}_i^{(k)}$ is the vector

$$\boldsymbol{\nu}_i^{(k)} = [\nu_{i1}^{(k)}, \nu_{i2}^{(k)}, \ldots, \nu_{iF_i^{(k)}}^{(k)}].$$

Here, $\nu_{if}^{(k)}$ denotes the number of class-k customers currently in service phase f.

Assuming that customers of class k, $k = 1, 2, \ldots, K$, arrive at the station according to a Poisson process with rate $\lambda_i^{(k)}(n_i^{(k)})$, where $n_i^{(k)}$ is the number of customers of that class currently at the station, the global equilibrium equation for state $\boldsymbol{\nu}_i$ takes the form

$$\begin{aligned} 0 \;=\; & -p(\boldsymbol{\nu}_i)\left[\sum_{k=1}^{K}\lambda_i^{(k)}(n_i^{(k)}) + \sum_{k=1}^{K}\sum_{f=1}^{F_i^{(k)}}\nu_{if}^{(k)}\mu_{if}^{(k)}\right] + \\ & + \sum_{k=1}^{K} p\left(\boldsymbol{\nu}_i^{1(k)-}\right)\lambda_i^{(k)}(n_i^{(k)}-1) + \\ & + \sum_{k=1}^{K}\sum_{f=1}^{F_i^{(k)}-1} p\left(\boldsymbol{\nu}_i^{f(k)+,[f+1](k)-}\right)(\nu_{if}^{(k)}+1)\mu_{if}^{(k)}a_{if}^{(k)} + \\ & + \sum_{k=1}^{K}\sum_{f=1}^{F_i^{(k)}} p\left(\boldsymbol{\nu}_i^{f(k)+}\right)(\nu_{if}^{(k)}+1)\mu_{if}^{(k)}b_{if}^{(k)}\,. \end{aligned} \tag{5.9}$$

We use notation analogous to that introduced in the previous chapter: $\boldsymbol{\nu}_i^{f(k)+}$ denotes the vector $\boldsymbol{\nu}_i$ in which component $\nu_{if}^{(k)}$ has been increased by one, and $\boldsymbol{\nu}_i^{f(k)+,[f+1](k)-}$ denotes the vector $\boldsymbol{\nu}_i$ in which $\nu_{if}^{(k)}$ has been increased by one and $\nu_{i,f+1}^{(k)}$ has been decreased by one.

The first term on the right-hand side of equation (5.9) represents the probability flow out of state $\boldsymbol{\nu}_i$. The state may change because a new customer of any class arrives, which contributes the factor

$$\sum_{k=1}^{K}\lambda_i^{(k)}(n_i^{(k)}),$$

or because some service phase of some customer completes, which contributes the factor

$$\sum_{k=1}^{K}\sum_{f=1}^{F_i^{(k)}}\nu_{if}^{(k)}\mu_{if}^{(k)}\,.$$

The remaining terms represent the probability flow into state $\boldsymbol{\nu}_i$ due to the following events:

(a) there was one fewer customer in the first service phase, and a new customer of the appropriate class arrived,

(b) there was one more customer in phase f and one fewer customer in phase $f+1$ of the same class; service in phase f completed and continued in the next phase,

(c) there was one more customer in phase f; service of a customer of this class completed in phase f, and that phase was the final service phase.

Of course, all components of the state vectors must remain non-negative.

The local equilibrium equation for state $\boldsymbol{\nu}_i$ associated with the departure and arrival of a class-k customer in service phase f takes the form

$$0 = -p(\boldsymbol{\nu}_i)\nu_{if}^{(k)}\mu_{if}^{(k)} + A_{i,f-1}^{(k)}\lambda_i^{(k)}(n_i^{(k)}-1)p(\boldsymbol{\nu}_i^{f(k)-}), \qquad f = 1,\ldots,F_i^{(k)}, \tag{5.10}$$

where, by convention, $A_{i0}^{(k)} = 1$.

The solution of equation (5.9) or (5.10), as well as the analogous equations for the remaining three station types, is given in equations (5.12).

(iii) *PS*-type station, whose processing capacity is shared equally among all customers. The state vector $\boldsymbol{\nu}_i$ is defined in the same way as for the IS station.

(iv) *LIFO* queue with a service discipline giving priority to the most recently arrived customer. When a new customer arrives, the current service is interrupted and later resumed from the point of interruption, after all customers that arrived later have completed service. In this case,

$$\boldsymbol{\nu}_i = [(k_{i1}, f_{i1}), \ldots, (k_{in_i}, f_{in_i})],$$

where the pair (k_{ij}, f_{ij}) describes the class and current service phase of the j-th customer in the $LIFO$ stack.

Example 5.1 Example 5.1

A closed network contains two stations: station 1 is of type IS, and station 2 is of type PS — see Fig. 5.4. Two customer classes, with populations $N^{(1)}$ and $N^{(2)}$, circulate within the network.

The routing probabilities are as follows:

$$r_{12}^{(1)(1)} + r_{11}^{(1)(1)} = 1,$$

$$r_{12}^{(2)(2)} = r_{21}^{(2)(2)} = r_{21}^{(1)(1)} = 1,$$

that is, customers do not change class when moving between stations.

A fraction of class-1 customers, after being served at station 1, immediately return to the same station. Service times are exponentially distributed with parameters

$$\mu_1^{(1)}, \qquad \mu_1^{(2)}, \qquad \mu_2^{(1)}, \qquad \mu_2^{(2)}.$$

Let us write the global and local equilibrium equations.

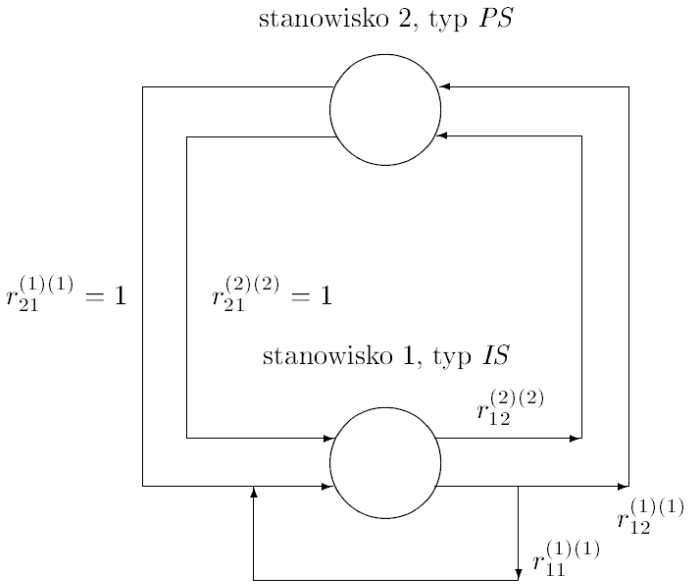


Figure 5.4: Model from Example 5.1

Global equilibrium balance:

$$\begin{aligned}
0 \;=\; & -p(n_1^{(1)}, n_1^{(2)}, n_2^{(1)}, n_2^{(2)}) \left[n_1^{(1)}\mu_1^{(1)} + n_1^{(2)}\mu_1^{(2)} + \frac{n_2^{(1)}}{n_2^{(1)}+n_2^{(2)}}\mu_2^{(1)} + \frac{n_2^{(2)}}{n_2^{(1)}+n_2^{(2)}}\mu_2^{(2)} \right] \\
& + p(n_1^{(1)}-1, n_1^{(2)}, n_2^{(1)}+1, n_2^{(2)}) \frac{n_2^{(1)}+1}{n_2^{(1)}+n_2^{(2)}+1}\mu_2^{(1)} \\
& + p(n_1^{(1)}+1, n_1^{(2)}, n_2^{(1)}-1, n_2^{(2)})(n_1^{(1)}+1)\mu_1^{(1)} r_{12}^{(1)(1)} \\
& + p(n_1^{(1)}, n_1^{(2)}+1, n_2^{(1)}, n_2^{(2)}-1)(n_1^{(2)}+1)\mu_1^{(2)} \\
& + p(n_1^{(1)}, n_1^{(2)}-1, n_2^{(1)}, n_2^{(2)}+1) \frac{n_2^{(2)}+1}{n_2^{(1)}+n_2^{(2)}+1}\mu_2^{(2)} \,.
\end{aligned}$$

The local equilibrium equation associated with the departure and arrival of a class-1 customer at station 1 is

$$\begin{aligned}
0 \;=\; & -p(n_1^{(1)}, n_1^{(2)}, n_2^{(1)}, n_2^{(2)}) n_1^{(1)}\mu_1^{(1)} \\
& + p(n_1^{(1)}-1, n_1^{(2)}, n_2^{(1)}+1, n_2^{(2)}) \frac{n_2^{(1)}+1}{n_2^{(1)}+n_2^{(2)}+1}\mu_2^{(1)} \\
& + p(n_1^{(1)}, n_1^{(2)}, n_2^{(1)}, n_2^{(2)}) n_1^{(1)}\mu_1^{(1)} r_{11}^{(1)(1)} \,.
\end{aligned}$$

The local equilibrium equation associated with the departure and arrival of a class-2 customer at station 1 is

$$\begin{aligned}
0 \;=\; & -p(n_1^{(1)}, n_1^{(2)}, n_2^{(1)}, n_2^{(2)}) n_1^{(2)}\mu_1^{(2)} \\
& + p(n_1^{(1)}, n_1^{(2)}-1, n_2^{(1)}, n_2^{(2)}+1) \frac{n_2^{(2)}+1}{n_2^{(1)}+n_2^{(2)}+1}\mu_2^{(2)} \,.
\end{aligned}$$

The local equilibrium equation associated with the departure and arrival of a class-1 customer at station 2 is

$$\begin{aligned} 0 &= -p(n_1^{(1)}, n_1^{(2)}, n_2^{(1)}, n_2^{(2)}) \frac{n_2^{(1)}}{n_2^{(1)} + n_2^{(2)}} \mu_2^{(1)} \\ &\quad + p(n_1^{(1)} + 1, n_1^{(2)}, n_2^{(1)} - 1, n_2^{(2)})(n_1^{(1)} + 1)\mu_1^{(1)} r_{12}^{(1)(1)} . \end{aligned}$$

The local equilibrium equation associated with the departure and arrival of a class-2 customer at station 2 is

$$\begin{aligned} 0 &= -p(n_1^{(1)}, n_1^{(2)}, n_2^{(1)}, n_2^{(2)}) \frac{n_2^{(2)}}{n_2^{(1)} + n_2^{(2)}} \mu_2^{(2)} \\ &\quad + p(n_1^{(1)}, n_1^{(2)} + 1, n_2^{(1)}, n_2^{(2)} - 1)(n_1^{(2)} + 1)\mu_1^{(2)} . \end{aligned}$$

■

5.3 BCMP Model

5.3.1 Model Definition and Form of the Solution

The BCMP model takes the form of a network of arbitrary topology, comprising a finite number M of stations of the four types described above. Customers are grouped into a finite number K of classes and may change class when moving between stations. The paths of customers through the network are determined by the probability matrix $\boldsymbol{R} = [r_{ij}^{(k)(l)}]$; $r_{ij}^{(k)(l)}$ denotes the probability that a customer of class k, upon leaving station i, moves to station j and becomes a customer of class l.

The pairs (i, k), where i is the station number and k is the customer class, define the states of the Markov chain associated with customer movements through the network. This chain is decomposable into m subchains. Let $E_1, E_2, \ldots, E_m$ denote the sets of states of these subchains, and let

$$N(E_j) = \sum_{(i,k) \in E_j} n_i^{(k)}.$$

The network may be mixed, i.e. closed for certain customer classes and open for the remaining classes. Of course, a network that is closed or open for all classes is a special case of a mixed network.

Two types of external traffic are possible:

(a) a Poisson stream with parameter $\lambda(N)$, where N is the total number of customers in the network. A customer arriving at the network chooses station i and class k with probability $r_i^{(k)}$;

(b) there are m Poisson streams corresponding to the m subchains mentioned above. The parameter of the jth stream is a function of the number of customers in subchain j, namely $\lambda_j(N(E_j))$. Each subchain has its own distribution $r_i^{(k)}$ for the choice of station i and class k, with

$$\sum_{(i,k) \in E_j} r_i^{(k)} = 1.$$

The solution for the network defined in this way is given in [6]:

$$p(\boldsymbol{\nu}) = \frac{1}{G}\, d(\boldsymbol{\nu}) \prod_{i=1}^{M} f_i(\boldsymbol{\nu}_i)\,, \tag{5.11}$$

where G is the normalisation constant, $d(\boldsymbol{\nu})$ is a function of the numbers of customers in the network, and $f_i(\boldsymbol{\nu}_i)$ depends on the type of station i:

$$f_i(\boldsymbol{\nu}_i) = \begin{cases} \left(\dfrac{1}{\mu_i}\right)^{n_i} \displaystyle\prod_{j=1}^{n_i} V_i^{(k_{ij})} & \text{for station } i \text{ of type } FIFO\,, \\ \displaystyle\prod_{k=1}^{K} \prod_{l=1}^{F_i^{(k)}} \left\{ \left[\frac{V_i^{(k)} A_{il}^{(k)}}{\mu_{il}^{(k)}}\right]^{\nu_{il}^{(k)}} \frac{1}{\nu_{il}^{(k)}!} \right\} & \text{for station } i \text{ of type } IS\,, \\ n_i! \displaystyle\prod_{k=1}^{K} \prod_{l=1}^{F_i^{(k)}} \left\{ \left[\frac{V_i^{(k)} A_{il}^{(k)}}{\mu_{il}^{(k)}}\right]^{\nu_{il}^{(k)}} \frac{1}{\nu_{il}^{(k)}!} \right\} & \text{for station } i \text{ of type } PS\,, \\ \displaystyle\prod_{j=1}^{n_i} \left\{ V_i^{(k_j)} A_{i\nu_j}^{(k_j)} \frac{1}{\mu_{i\nu_j}^{(k_j)}} \right\} & \text{for station } i \text{ of type } LIFO\,. \end{cases} \tag{5.12}$$

Here, $V_i^{(k)}$ is the mean number of visits made by a customer of class k to station i.
If the external arrival process is of type (a), i.e. customers arrive at rate $\lambda(N)$, then

$$d(\boldsymbol{\nu}) = \prod_{j=0}^{N-1} \lambda(j).$$

In the case of external traffic of type (b), $d(\boldsymbol{\nu})$ is calculated as

$$d(\boldsymbol{\nu}) = \prod_{j=1}^{m} \prod_{i=0}^{N(E_j)-1} \lambda_j(i).$$

For a closed network,

$$d(\boldsymbol{\nu}) = 1.$$

The correctness of solution (5.12) is proved by direct verification, i.e. by showing that it satisfies the equilibrium equations.

5.3.2 Marginal Distributions

Let us define the state of the network as

$$\boldsymbol{n} = (\boldsymbol{n}_1, \boldsymbol{n}_2, \ldots, \boldsymbol{n}_M),$$

where

$$\boldsymbol{n}_i = [n_i^{(1)}, n_i^{(2)}, \ldots, n_i^{(K)}].$$

Thus, we take into account only the numbers of customers of each class at the stations, disregarding their current service phases.

Let $1/\mu_i^{(k)}$ be the mean service time for a customer of class k at station i. The probability distribution then has the form

$$p(\boldsymbol{n}) = \frac{1}{G}\, d(\boldsymbol{n}) \prod_{i=1}^{M} g_i(\boldsymbol{n}_i), \tag{5.13}$$

where

$$g_i(\boldsymbol{n}_i) = \begin{cases} n_i! \left[\prod_{k=1}^{K} \frac{1}{n_i^{(k)}!} \left(V_i^{(k)} \right)^{n_i^{(k)}} \right] \left(\frac{1}{\mu_i} \right)^{n_i} & \text{for station } i \text{ of type } \textit{FIFO}\,, \\ n_i! \prod_{k=1}^{K} \frac{1}{n_i^{(k)}!} \left(\frac{V_i^{(k)}}{\mu_i^{(k)}} \right)^{n_i^{(k)}} & \text{for station } i \text{ of type } \textit{PS} \text{ or } \textit{LIFO}\,, \\ \prod_{k=1}^{K} \frac{1}{n_i^{(k)}!} \left(\frac{V_i^{(k)}}{\mu_i^{(k)}} \right)^{n_i^{(k)}} & \text{for station } i \text{ of type } \textit{IS}\,. \end{cases} \tag{5.14}$$

As can be seen, although the *PS*, *LIFO*, and *IS* stations may have general service-time distributions, only the mean values of these distributions appear in expression (5.14).

For open networks, in which the external customer arrival process has a constant parameter

$$\lambda(N) = \lambda = const.,$$

independent of the number of customers in the network, the normalisation constant can be written as

$$\begin{aligned} G &= \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \cdots \sum_{n_M=0}^{\infty} \left(\prod_{i=1}^{M} \lambda^{n_i} h_i(n_i) \right) \\ &= \left(\sum_{n_1=0}^{\infty} \lambda^{n_1} h_1(n_1) \right) \left(\sum_{n_2=0}^{\infty} \lambda^{n_2} h_2(n_2) \right) \cdots \left(\sum_{n_M=0}^{\infty} \lambda^{n_M} h_M(n_M) \right)\,, \end{aligned}$$

where

$$\sum_{n_i=0}^{\infty} \lambda^{n_i} h_i(n_i) = \begin{cases} \left(1 - \sum_{k=1}^{K} \lambda \frac{V_i^{(k)}}{\mu_i} \right)^{-1} & \text{for station } i \text{ of type } \textit{FIFO}\,, \\ \left(1 - \sum_{k=1}^{K} \lambda \frac{V_i^{(k)}}{\mu_i^{(k)}} \right)^{-1} & \text{for station } i \text{ of type } \textit{PS} \text{ or } \textit{LIFO}\,, \\ \exp \left[\sum_{k=1}^{K} \lambda \frac{V_i^{(k)}}{\mu_i^{(k)}} \right] & \text{for station } i \text{ of type } \textit{IS}\,. \end{cases}$$

Since the normalisation constant, like the expression $p(\boldsymbol{n})$, has a product form, the numbers of customers at different stations are independent random variables. Let us calculate the marginal distribution at station i:

$$p_i(n_i) = \frac{1}{G}\, \lambda^{n_i} h_i(n_i) \prod_{\substack{j=1 \\ j \neq i}}^{M} \left(\sum_{n_j=0}^{\infty} \lambda^{n_j} h_j(n_j) \right)\,.$$

Using the expression for the normalisation constant, we obtain

$$p_i(n_i) = \frac{\lambda^{n_i} h_i(n_i)}{\sum_{m=0}^{\infty} \lambda^m h_i(m)} .$$

After introducing the quantity ϱ_i,

$$\varrho_i = \begin{cases} \displaystyle\sum_{k=1}^{K} \lambda \left(\frac{V_i^{(k)}}{\mu_i} \right) & \text{if station } i \text{ is of type } \textit{FIFO}\,, \\ \displaystyle\sum_{k=1}^{K} \lambda \left(\frac{V_i^{(k)}}{\mu_i^{(k)}} \right) & \text{if station } i \text{ is of type } \textit{PS}, \textit{IS}, \text{ or } \textit{LIFO}\,, \end{cases}$$

we obtain

$$p_i(n_i) = \begin{cases} (1-\varrho_i)\varrho_i^{n_i} & \text{for a station of type } \textit{FIFO}, \textit{PS}, \text{ or } \textit{LIFO}\,, \\ e^{-\varrho_i} \dfrac{\varrho_i^{n_i}}{n_i!} & \text{for a station of type } \textit{IS}\,. \end{cases}$$

In this model, one may also account for the dependence of service time on the number of customers at the station. Let $x_i(n_i)$ be an arbitrary positive function of n_i, representing the relative service rate at station i when n_i customers are present, relative to the service rate when $n_i = 1$. Thus,

$$x_i(1) = 1.$$

In that case, the expression $f_i(\boldsymbol{\nu}_i)$ in solution (5.12) must be multiplied by the factor

$$\frac{1}{\prod_{l=1}^{n_i} x_i(l)} .$$

If the service rate for a given customer class depends on the number of customers of that class, and the relative service factor is $x_i^{(k)}(n_i^{(k)})$ with

$$x_i^{(k)}(1) = 1,$$

then the expression $f_i(\boldsymbol{\nu}_i)$ in solution (5.12) is replaced by

$$f_i(\boldsymbol{\nu}_i) \prod_{k=1}^{K} \prod_{j=1}^{n_i^{(k)}} \frac{1}{x_i^{(k)}(j)} .$$

One may also formulate more general relationships between the service rate and the numbers of customers present at several stations simultaneously. Let

$$I = \{i_1, i_2, \ldots, i_p\}$$

be a subset of stations, and let

$$n_I = \sum_{i \in I} n_i$$

be the total number of customers at those stations. The service coefficient at station $i \in I$ is then assumed to depend on both n_I and n_i. Let $Z_I(n_I)$ be the relative service-rate factor, with

$$Z_I(1) = 1.$$

In this case, the expression

$$\prod_{i \in I} f_i(\boldsymbol{\nu}_i)$$

is replaced by

$$\left\{ \prod_{i \in I} \left[f_i(\boldsymbol{\nu}_i) \prod_{j=1}^{n_i} \frac{1}{x_i(j)} \right] \right\} \prod_{l=1}^{n_I} \frac{1}{Z_I(l)} \, .$$

5.3.3 Numerical Problems in Solving BCMP Models

Numerical results for BCMP models can be obtained in two ways:

1. *Direct use of the formulas presented in the previous subsection*. The normalisation constant

   $$G = G(M, \boldsymbol{N}),$$

   where

   $$\boldsymbol{N} = [N^{(1)}, \ldots, N^{(K)}],$$

   denotes the numbers of customers in the individual classes in the network. It is calculated using the convolution algorithm presented in the previous chapter and extended to account for K customer classes.

   The algorithm assumes that customers do not change class membership. If class switching is possible, the set of K classes must be divided into $C \leq K$ subsets between which no customer exchange occurs, and the network must be transformed into one with C customer classes and no class switching. A detailed description of the algorithm for closed networks is given in [9], and for mixed networks in [80].

   Certain computational savings for networks with sparse routing matrices $\boldsymbol{R}$, where customers of particular classes visit only parts of the network, can be achieved by selecting an appropriate order of computation [70].

   Recall that the value of the normalisation constant depends on the chosen traffic vector $\boldsymbol{V}$, which is not unique in closed networks:

   $$G(M, \boldsymbol{N}, \boldsymbol{V}) = \prod_{k=1}^{K} \left(V_1^{(k)} \right)^{N^{(k)}} G(M, \boldsymbol{N}, \mathbf{1}), \tag{5.15}$$

   and, depending on the choice of $\boldsymbol{V}$, the value of G may grow or shrink rapidly during the calculations, exceeding the range allowed by floating-point arithmetic.

   Equation (5.15) permits simple rescaling of the normalisation constant:

   $$G(M, \boldsymbol{N}, \boldsymbol{V}') = G(M, \boldsymbol{N}, \boldsymbol{V}) \prod_{k=1}^{K} \left(\frac{V_1^{(k)\prime}}{V_1^{(k)}} \right)^{N^{(k)}} , \tag{5.16}$$

   which makes it possible to keep the value of G within acceptable numerical limits.

2. *Use of the Mean Value Analysis (MVA) algorithm*. The MVA algorithm, discussed previously, yields exact results for BCMP networks. We previously described it for closed networks with multiple customer classes and service times independent of the number of customers at a station, cf. equations (2.7)–(2.12).

   Let us rewrite the algorithm for closed networks, using the notation adopted here and allowing the service time to depend on the number of customers at a station:

   $$E[B_i^{(k)}] = E[B_i^{(k)}(n_i)].$$

   This dependence implies that the queue length observed by an arriving customer does not fully determine that customer's response time, because subsequent arrivals may alter the service times of customers already present.

   Recall the notation: for a given population vector $\boldsymbol{n}$, containing a total of N customers of all K classes, n_i is the total number of customers at station i, $i = 1, \ldots, M$; $n_i^{(k)}$ is the number of class-k customers at station i; and $N^{(k)}$ is the total number of class-k customers in the network.

   The response time for a single visit to station i can be written as

   $$E[R_i^{(k)}(\boldsymbol{n})] = E[B_i^{(k)}]\,, \tag{5.17}$$

   if station i is of type *IS*.

   For all other station types:

   $$E[R_i^{(k)}(\boldsymbol{n})] = E[B_i^{(k)}]\left[1 + E[N_i(\boldsymbol{n} - \mathbf{1}^{(k)})]\right] \tag{5.18}$$

   when the service time is independent of the number of customers at the station, or

   $$E[R_i^{(k)}(\boldsymbol{n})] = \sum_{n_i=1}^{N} E[B_i^{(k)}(n_i)]\, n_i\, p_i(n_i - 1, \boldsymbol{n} - \mathbf{1}^{(k)}) \tag{5.19}$$

   when the service time depends on the number of customers at the station.

   The vector $\mathbf{1}^{(k)}$ is a K-element vector whose k-th component is equal to 1 and whose remaining components are equal to 0.

   The throughput of station i for class k customers is then

   $$\lambda_i^{(k)}(\boldsymbol{n}) = \frac{V_i^{(k)} N^{(k)}}{\sum_{j=1}^{M} V_j^{(k)} E[R_j^{(k)}(\boldsymbol{n})]}\,, \tag{5.20}$$

   and the average number of class-k customers at station i is

   $$E[N_i^{(k)}(\boldsymbol{n})] = \lambda_i^{(k)}(\boldsymbol{n}) E[R_i^{(k)}(\boldsymbol{n})]\,. \tag{5.21}$$

   For equation (5.19), one must recursively compute the marginal queue-length distributions at the stations:

   $$p_i(n_i, \boldsymbol{n}) = \sum_{k=1}^{K} E[B_i^{(k)}(n_i)]\, \lambda_i^{(k)}(\boldsymbol{n})\, p_i(n_i - 1, \boldsymbol{n} - \mathbf{1}^{(k)})\,, \tag{5.22}$$

   $$p_i(0, \boldsymbol{n}) = 1 - \sum_{n_i=1}^{N} p_i(n_i, \boldsymbol{n})\,. \tag{5.23}$$

These formulas are applied recursively for all possible populations between $\mathbf{0}$ and $\boldsymbol{n}$, using the initial conditions

$$E[N_i(\mathbf{0})] = 0,$$

$$p_i(n_i, \mathbf{0}) = 0 \qquad \text{for } n_i > 0,$$

$$p_i(0, \mathbf{0}) = 1.$$

If the utilisation of station i is high, i.e. $p_i(0, \boldsymbol{n})$ is close to zero, then equation (5.23) may become a source of numerical instability and significant errors [13]. Therefore, [98] proposes aggregating the entire network outside station i (as in Example 4.3) into a single substitute station and then computing $p_i(0, \boldsymbol{n})$ as

$$p_i(0, \boldsymbol{n}) = \frac{p_i(0, \boldsymbol{n} - \mathbf{1}^{(k)})\, \lambda^{(k)}(\boldsymbol{n})}{\lambda^{(k)}_{z(i)}(\boldsymbol{n})}\,, \tag{5.24}$$

where $\lambda^{(k)}_{z(i)}$ is the throughput of the reduced network.

A detailed comparison of the computational complexity of the convolution method and the MVA method is given in [120].

When service times are independent of queue length, the computational complexity of both methods is the same, namely proportional to

$$KM \prod_{k=1}^{K} (N^{(k)} + 1)\,.$$

The ratio of memory usage in the convolution method to that in the MVA method is

$$\frac{2 \max_k (N^{(k)} + 1)}{KM + K + M}\,.$$

The relative efficiency of these methods therefore depends on the parameters of the network being analysed. Ease of implementation, conceptual clarity, and the absence of any need to rescale the normalisation constant all favour the mean-value method.

Example 5.2

Model of a multiprogramming system with virtual memory.

The system configuration is shown in Fig. 5.5, and its operational parameters relevant to the model under consideration are as follows:

v_{CPU}	=	1.2×10^6	—	average number of instructions executed by the central processing unit per second,
ω_1	=	4500 revolutions/min	—	rotational speed of drives 2 and 3,
ω_2	=	3600 revolutions/min	—	rotational speed of drive 4,
t_g	=	8 ms	—	average head movement time for any disk,
t_{s1}	=	2 μs	—	average transfer time for a single word to or from drives 2 and 3,
t_{s2}	=	4 μs	—	average transfer time for a single word to or from drive 4.

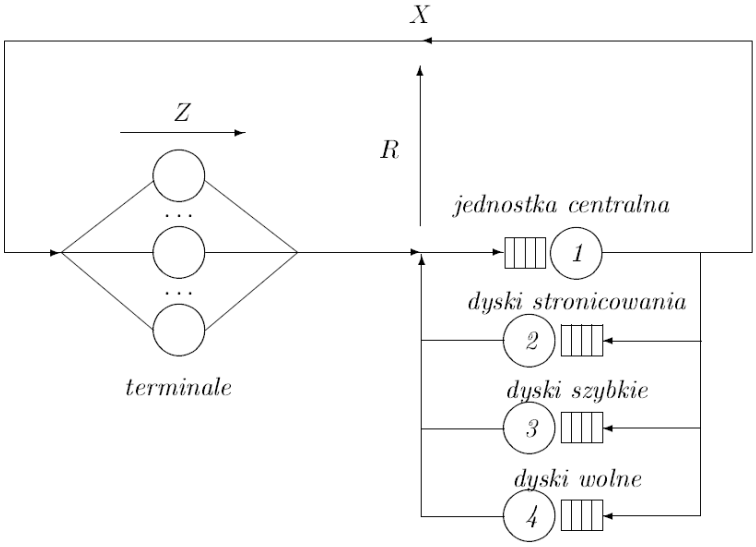


Figure 5.5: System model in Example 5.2

Access to external memory is performed in blocks of size 1024 bytes.

The system has virtual memory; the experimentally determined program lifetime curve (common to tasks of both classes) is as follows:

$N = N^{(1)} + N^{(2)}$	1	2	3	4	5	10	15	20	25	30	
$E[B_{str}]$	50	46	43	39	35	24	17	12	8	5	[ms]

Here, $E[B_{str}]$ denotes the average time between successive page faults.

The system supports two classes of programs: interactive tasks run from terminals and batch tasks executed without terminal involvement.

The parameters for the first class are as follows:

$L^{(1)} = 2.1 \times 10^6$ — average number of instructions in the program,
$V_3^{(1)} = 80$ — average number of accesses to files on drive 3,
$V_4^{(1)} = 60$ — average number of accesses to files on drive 4.

For the second class of tasks, these parameters are:

$$L^{(2)} = 1.5 \times 10^6, \qquad V_3^{(2)} = 50, \qquad V_4^{(2)} = 30.$$

The average terminal think time for first-class tasks is

$$E[Z] = 20\ s.$$

We wish to investigate the effect of the numbers of tasks in both classes on the system performance parameters, specifically the response time for first-class tasks, CPU utilisation, and disk utilisation.

We will use a BCMP model, assuming that the terminal is represented by an IS station with a finite number of service channels, and the central processing unit by a PS station. The service-time distributions at these stations are irrelevant: the model allows arbitrary distributions (represented by appropriately chosen Cox distributions); these may differ between task classes, and only their means affect the results.

In the case of queues modelled as FIFO stations, we must assume that the service-time distributions are exponential and identical for both customer classes (although in practice only the latter assumption corresponds to reality). The disk as a service station is discussed in more detail in Example 6.2; here we simply calculate the average service time associated with the transfer of a 1024-byte block.

For the fast fixed-head drives (stations 2 and 3), the service time consists of the rotational latency (on average, half the disk rotation period) required for the read/write head to reach the appropriate sector, plus the read/write time itself. In the case of the slower moving-head drive (station 4), one must also take into account the head movement time, since the head must be positioned over the relevant track.

We calculate the times

$$E[B_2^{(1)}] = E[B_2^{(2)}] = E[B_3^{(1)}] = E[B_3^{(2)}] :$$

average rotational latency + average transfer time

$$= \frac{1}{2} \times \frac{1}{4500} \times 60 \times 10^3 + 1024 \times 2 \times 10^{-3} = 6.66(6) + 2.048 = 8.715 \text{ ms} ,$$

and the time

$$E[B_4^{(1)}] = E[B_4^{(2)}] :$$

average rotational latency + average head movement time + average transfer time

$$= \frac{1}{2} \times \frac{1}{3600} \times 60 \times 10^3 + 8 + 1024 \times 4 \times 10^{-3} = 8.33(3) + 8 + 4.096 = 20.429 \text{ ms}.$$

The average service time in the central processing unit, $E[B_1^{(i)}]$, can be obtained as follows:

- We calculate the total average service time of a task at the central processing unit:

$$E[D_1^{(i)}] = \frac{L^{(i)}}{v_{CPU}} ,$$

so for the given data

$$\begin{aligned} E[D_1^{(1)}] &= \frac{2.1 \times 10^6}{1.2 \times 10^6 \text{ 1/s}} = 1.75 \text{ s}, \\ E[D_1^{(2)}] &= \frac{1.5 \times 10^6}{1.2 \times 10^6 \text{ 1/s}} = 1.25 \text{ s}; \end{aligned}$$

- We calculate the number of interruptions of the total CPU service time caused by page faults (and thus by the customer's transition to station 2, the paging disk):

$$V_2^{(i)}(N) = \frac{E[D_1^{(i)}]}{E[B_{str}(N)]} ;$$

- We calculate the total number of interruptions of the CPU service time caused by transitions to any disk (excluding the final CPU visit, after which the task completes):

$$V_1^{(i)}(N) = V_2^{(i)}(N) + V_3^{(i)} + V_4^{(i)} + 1 ;$$

- We calculate the average CPU service time between successive interruptions:

$$E[B_1^{(i)}(N)] = \frac{E[D_1^{(i)}]}{V_1^{(i)}(N)} .$$

For example, for $N^{(1)} = 10$, $N^{(2)} = 5$, i.e. for

$$E[B_{str}(15)] = 17\,\text{ms},$$

we obtain for the first class of customers

$$\begin{aligned} V_2^{(1)}(15) &= \frac{1.75\,\text{s}}{17 \times 10^{-3}\,\text{s}} = 102.94\,, \\ V_1^{(1)}(15) &= V_2^{(1)}(15) + V_3^{(1)} + V_4^{(1)} + 1 = 102.94 + 80 + 60 + 1 = 243.94\,, \\ E[B_1^{(1)}(15)] &= \frac{E[D_1^{(1)}]}{V_1^{(1)}(15)} = \frac{1.75\,\text{s}}{243.94} = 7.17 \times 10^{-3}\,\text{s}\,, \end{aligned}$$

and for the second class of customers

$$\begin{aligned} V_2^{(2)}(15) &= \frac{1.25\,\text{s}}{17 \times 10^{-3}\,\text{s}} = 73.53\,, \\ V_1^{(2)}(15) &= V_2^{(2)}(15) + V_3^{(2)} + V_4^{(2)} + 1 = 73.53 + 50 + 30 + 1 = 154.53\,, \\ E[B_1^{(2)}(15)] &= \frac{E[D_1^{(2)}]}{V_1^{(2)}(15)} = \frac{1.25\,\text{s}}{154.53} = 8.09 \times 10^{-3}\,\text{s}\,. \end{aligned}$$

We determine the transition probabilities for both task classes as

$$r_{10}^{(i)}(N) = \frac{1}{V_1^{(i)}(N)}, \qquad r_{12}^{(i)}(N) = \frac{V_2^{(i)}(N)}{V_1^{(i)}(N)}, \qquad r_{13}^{(i)}(N) = \frac{V_3^{(i)}}{V_1^{(i)}(N)}, \qquad r_{14}^{(i)}(N) = \frac{V_4^{(i)}}{V_1^{(i)}(N)}.$$

In the same case, $N^{(1)} = 10$, $N^{(2)} = 5$, for the first class of customers we obtain:

$$\begin{aligned} r_{10}^{(1)}(15) &= \frac{1}{V_1^{(1)}(15)} = \frac{1}{243.94} = 0.00410\,, \\ r_{12}^{(1)}(15) &= \frac{V_2^{(1)}(15)}{V_1^{(1)}(15)} = \frac{102.94}{243.94} = 0.42199\,, \\ r_{13}^{(1)}(15) &= \frac{V_3^{(1)}}{V_1^{(1)}(15)} = \frac{80}{243.94} = 0.32795\,, \\ r_{14}^{(1)}(15) &= \frac{V_4^{(1)}}{V_1^{(1)}(15)} = \frac{60}{243.94} = 0.24596\,, \end{aligned}$$

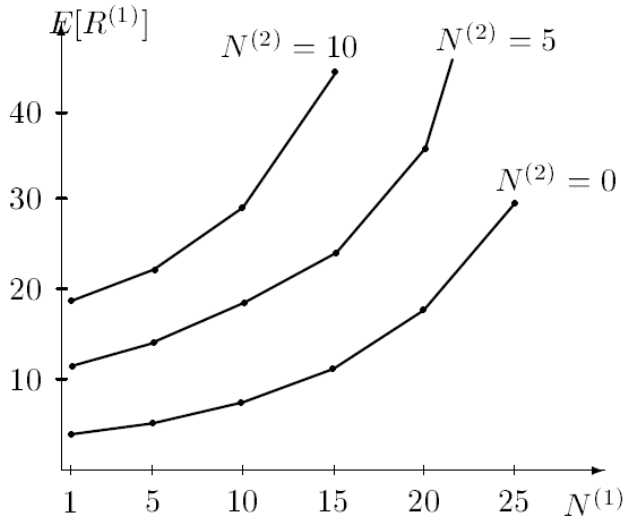


Figure 5.6: System response time for terminal tasks as a function of the number of tasks in both classes

and for the second class of customers:

$$
\begin{aligned}
r_{10}^{(2)}(15) &= \frac{1}{V_1^{(2)}(15)} = \frac{1}{154.53} = 0.00647\,, \\
r_{12}^{(2)}(15) &= \frac{V_2^{(2)}(15)}{V_1^{(2)}(15)} = \frac{73.53}{154.53} = 0.47582\,, \\
r_{13}^{(2)}(15) &= \frac{V_3^{(2)}}{V_1^{(2)}(15)} = \frac{50}{154.53} = 0.32356\,, \\
r_{14}^{(2)}(15) &= \frac{V_4^{(2)}}{V_1^{(2)}(15)} = \frac{30}{154.53} = 0.19415\,.
\end{aligned}
$$

The results are shown in Figs. 5.6 and 5.7. As can be seen, the presence of batch jobs increases the system response time observed by terminal users. It is also evident that, for a small number of tasks, the system bottleneck is the central processing unit. Later, due to increasingly frequent page faults (as less and less RAM is available per user), the number of accesses to the paging disk increases, tasks queue up for this disk, and CPU utilisation consequently drops.

■

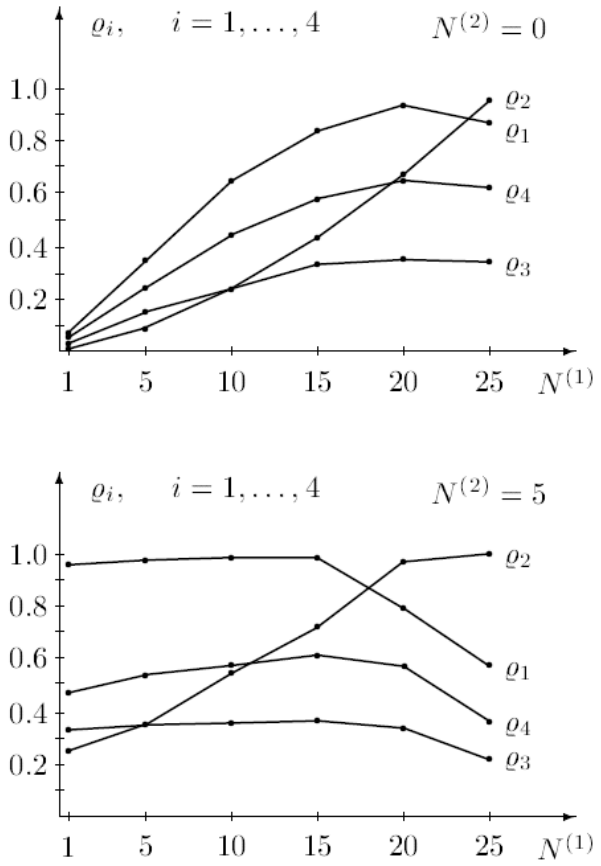


Figure 5.7: CPU and disk utilisation as a function of the number of active tasks

Chapter 6

Non-Markovian models

A significant limitation of the BCMP model is that, in the case of a natural queueing service policy, it only allows for a discrete distribution of service times. In a real computer system, the times of uninterrupted CPU operation, and consequently the transmission times, do not follow a distribution to the exponential distribution, and by making this assumption we may commit a serious error.

Below we will briefly demonstrate that it is possible to perform an accurate analysis of a single service point with a , in which one of the distributions $A(x)$, $B(x)$ differs from the exponential, and thus systems of the type $M/G/1$ and $G/M/1$, whilst the description of the system $G/G/1$, and thus the network of systems $G/G/1$ is, in practice, only in an approximate form.

A natural way of describing the operation of systems $M/G/1$ and $G/M/1$ is via Markov chains: we observe the process $N(t)$, i.e. the number of customers at these stations not at all times, but at selected instants. For the $M/G/1$ system, these are the moments of service completion ; for the $G/M/1$ system, these are the moments when a new task arrives. Viewed only at these specific moments in time, the process is a Markov chain — we free ourselves from dependence on the system's history, which introduces the distribution G.

6.1 System $M/G/1$

The distribution $p(n)$ of the number of customers present at the station is described by the Pollaczek–Khinchine formula, which gives the generating function of this distribution [61]:

$$G(z) = \bar{b}(\lambda - \lambda z)\, \frac{(1-\varrho)(1-z)}{\bar{b}(\lambda - \lambda z) - z}\,, \tag{6.1}$$

from which the expected value follows:

$$E[N] = \left.\frac{dG(z)}{dz}\right|_{z=1} = \varrho + \frac{\lambda^2 E[B^2]}{2(1-\varrho)}\,. \tag{6.2}$$

The utilisation factor is

$$\varrho = \frac{E[B]}{E[A]} = \lambda E[B]\,.$$

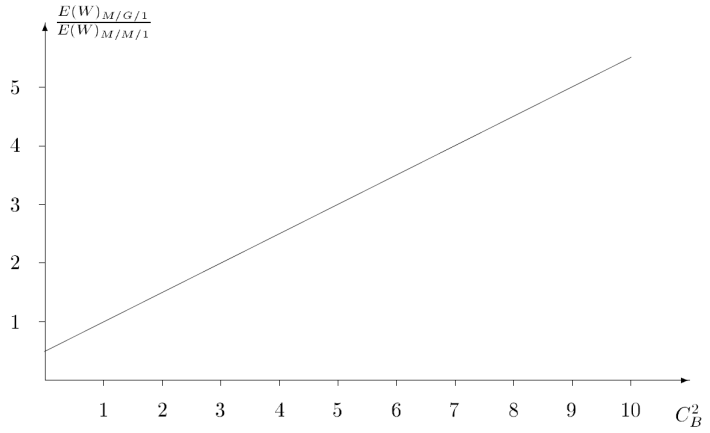


Figure 6.1: Average waiting time $E[W]_{M/G/1}$ in the $M/G/1$ system compared with the waiting time $E[W]_{M/M/1}$ in the $M/M/1$ system for the same values of λ and $E[B]$

The Laplace transform $\bar{w}(s)$ of the waiting-time distribution $w(x)$ is given by another form of the Pollaczek–Khinchine formula [61]:

$$\bar{w}(s) = \frac{s(1-\varrho)}{s - \lambda[1 - \bar{b}(s)]} \,, \tag{6.3}$$

from which we calculate the average waiting time:

$$E[W] = (-1)\left.\frac{d\bar{w}(s)}{ds}\right|_{s=0} = \frac{\lambda E[B^2]}{2(1-\varrho)} \,. \tag{6.4}$$

Rewriting equation (6.4) gives

$$E[W]_{M/G/1} = \frac{\varrho E[B]}{2(1-\varrho)}(1 + C_B^2) \,,$$

and consequently

$$\frac{E[W]_{M/G/1}}{E[W]_{M/M/1}} = \frac{1 + C_B^2}{2} \,.$$

Thus, the average waiting time in the $M/G/1$ system depends linearly on the squared coefficient of variation C_B^2 of the service-time distribution. For example, in the $M/D/1$ system it is approximately half as large as in the $M/M/1$ system, while maintaining the same mean service time; see Fig. 6.1.

Example 6.1

Let the distribution densities $A(x)$ and $B(x)$ in the $M/G/1$ system be given by:

$$a(x) = \lambda e^{-\lambda x} \qquad \textit{and} \qquad b(x) = \frac{1}{4}\mu' e^{-\mu' x} + \frac{3}{4}(2\mu')e^{-2\mu' x} \,,$$

that is, the service time follows a hyperexponential distribution. We wish to determine the distribution $p(n)$.

We calculate:

$$\begin{aligned} \bar{b}(s) &= \frac{1}{4}\frac{\mu'}{\mu'+s} + \frac{3}{4}\frac{2\mu'}{2\mu'+s}\,, \\ G(z) &= \bar{b}(\lambda-\lambda z)\,\frac{(1-\varrho)(1-z)}{\bar{b}(\lambda-\lambda z)-z}\,. \end{aligned}$$

The values of $E[N]$ and $p(n)$ can be obtained directly from the generating function $G(z)$ in this form, but it is more convenient to transform it to

$$G(z) = (1-\varrho)\left[\frac{1/4}{1-(2/5)z} + \frac{3/4}{1-(2/3)z}\right]\,.$$

Hence, after expanding $G(z)$ into a power series, we obtain

$$p(n) = \frac{3}{32}\left(\frac{2}{5}\right)^n + \frac{9}{32}\left(\frac{2}{3}\right)^n\,.$$

■

Example 6.2

Let us use the $M/G/1$ model to describe the operation of a disk drive, taking into account, as components of the service time, the head movement time, rotational latency, and transfer time.

Let us introduce the following notation:

- T_p — head movement time, required to position the read/write head over the appropriate track on the disk,
- T_r — rotational latency, i.e. the time during which the head, already positioned over the selected track, waits for disk rotation to bring the desired sector under the head,
- T_s — the time needed to transfer a block of data from the disk to main memory, or in the opposite direction.

The total access time is the sum

$$B = T_p + T_r + T_s.$$

We assume that its components are independent random variables, so

$$\begin{aligned} E[B] &= E[T_p] + E[T_r] + E[T_s], \\ \mathrm{var}[B] &= \mathrm{var}[T_p] + \mathrm{var}[T_r] + \mathrm{var}[T_s]. \end{aligned}$$

The head movement time is a function of the distance, measured in number of tracks, that the head must travel from the track on which the previous read or write operation took place to the track required for the next operation. This is a nonlinear function, taking into account the time required for acceleration and deceleration of the arm. We assume the form

$$t(i) = a + b\sqrt{i},$$

where $i \geq 1$ is the distance in tracks. The parameters a and b are characteristic of a given disk type, and

$$t(0) = 0.$$

Let the disk contain C tracks, and let the distribution of the target track be uniform. Since successive accesses often address neighbouring blocks, we distinguish this case by assuming that with probability p the head movement time is $T_p = 0$, while with probability $1 - p$ head movement is required.

If the head was previously positioned over the first track (the one closest to the centre), it may move by $1, 2, \ldots, C-1$ tracks outward.

If the head was previously over track 2, it may move one track inward or by $1, 2, \ldots, C-2$ tracks outward.

By listing all possible previous positions and movements in this way, we see that there are $C(C-1)$ possible moves, including $2(C-i)$ moves of length i tracks.

The probability that, in the next access, the head must travel a distance of i tracks, for $i \geq 1$, is therefore

$$p(i) = \begin{cases} p & \text{for } i = 0\,, \\ (1-p)\,\dfrac{2(C-i)}{C(C-1)} & \text{for } i = 1, \ldots, C-1\,, \end{cases}$$

which allows us to calculate

$$\begin{aligned} E[T_p] &= \sum_{i=1}^{C-1} p(i)\, t(i)\,, \\ E[T_p^2] &= \sum_{i=1}^{C-1} p(i)\, [t(i)]^2\,, \\ \mathrm{var}[T_p] &= E[T_p^2] - E[T_p]^2\,. \end{aligned}$$

We assume the rotational latency to be uniformly distributed over the interval $[0, R]$, where R is the time of one full rotation. Under this assumption,

$$E[T_r] = \frac{R}{2}, \qquad \mathrm{var}[T_r] = \frac{R^2}{12}\,.$$

The transfer time T_s, associated with the transmission of a block of fixed size, is assumed to be constant:

$$E[T_s] = LD,$$

where L is the block length in words, and D is the transfer time of a single word. Hence,

$$\mathrm{var}[T_s] = 0.$$

In this way, we have determined the mean and variance of the total service time, so we can use formulas (6.2) and (6.4) concerning the queue length and waiting time in the $M/G/1$ system to analyse disk performance.

■

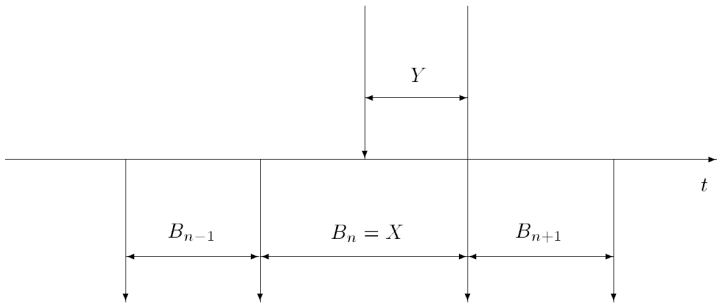


Figure 6.2: Service time X and residual service time Y

Example 6.3

A new customer arrives at a service station and finds n customers there, of whom $n-1$ are waiting in the queue, while one is in service. The time required to complete service for the customer currently being served is called the **residual service time**. *So far, we have assumed (in the MVA algorithm and in the derivation of the response-time distribution for the $M/M/1$ system) that the residual service time is equal to the service time itself. This is true in the case of an exponential service-time distribution, which has the memoryless property — cf. equation (3.8) — so that, regardless of the observation instant, the residual service time follows the same exponential distribution. However, this is not true for a general service-time distribution.*

Figure 6.2 shows a time axis with marked service intervals for successive customers. Let us assume that at time X during the service of the nth customer, a new customer arrives. The residual service time is the interval Y marked in the figure. What is its distribution?

Service times follow distribution $B(x)$ with density $b(x)$, but the interval X, singled out by the fact that a new customer arrives during it, has a different distribution. Since the probability that a new customer arrives during a service interval is proportional to the length of that interval (arrival intensity λ is constant), the density of the service interval during which the customer arrives has the form

$$f_X(x) = \gamma x b(x) \ .$$

We determine the coefficient γ from the normalisation condition

$$\int_0^\infty f_X(x)\,dx = \gamma \int_0^\infty x b(x)\,dx = 1 \ ,$$

hence

$$\gamma = \frac{1}{E[B]} \ ,$$

that is,

$$f_X(x) = \frac{x\,b(x)}{E[B]} \ .$$

Since the arrival of a customer at any instant is equally likely, then for a fixed interval length $X = x$, the conditional distribution of the random variable Y is uniform:

$$P[Y \le y \mid X = x] = \frac{y}{x} \ , \qquad 0 \le y \le x.$$

Hence the joint density of the variables X and Y is

$$P[y < Y \leq y + dy,\ x < X \leq x + dx] = \frac{dy}{x} \cdot \frac{x\, b(x)\, dx}{E[B]} = \frac{b(x)}{E[B]}\, dx\, dy\ ,$$

from which we obtain the marginal density $f_Y(y)$ of the variable Y:

$$f_Y(y)\, dy = \int_{x=y}^{\infty} \frac{b(x)}{E[B]}\, dx\, dy = \frac{1 - B(y)}{E[B]}\, dy\ .$$

It is convenient to express this result in terms of the Laplace transform:

$$\bar{f}_Y(s) = \frac{1/s - \bar{B}(s)}{E[B]} = \frac{1 - \bar{b}(s)}{sE[B]}\ ,$$

and then compute the successive moments of the residual service-time distribution:

$$E[Y^m] = (-1)^m \frac{d^m}{ds^m} \bar{f}_Y(s)\bigg|_{s=0} = \frac{E[B^{m+1}]}{(m+1)E[B]}\ , \tag{6.5}$$

and, in particular, the mean value

$$E[Y] = -\frac{d}{ds} \bar{f}_Y(s)\bigg|_{s=0} = \frac{E[B^2]}{2E[B]}\ . \tag{6.6}$$

Since

$$E[B^2] = E[B]^2 + \sigma_B^2,$$

result (6.6) can be written as

$$E[Y] = \frac{E[B]}{2} + \frac{\sigma_B^2}{2E[B]}\ .$$

Thus, the greater the variance of the service-time distribution, the longer the mean residual service time.

In the case of deterministic service ($\sigma_B^2 = 0$), the mean residual service time is equal to half the service time. For exponential service, where

$$E[B] = \frac{1}{\mu}\ , \qquad \sigma_B^2 = \frac{1}{\mu^2},$$

we obtain

$$E[Y] = \frac{1}{\mu}\ ,$$

that is, the mean residual service time is equal to the mean service time.

■

Example 6.4

Let us consider a local token-ring network with token passing. A station holding the token may send one packet of fixed size to any other station, after which it passes the token, and thus the right to transmit, to the neighbouring station; see Fig. **??**.

Assume that the packet transmission time is constant and equal to x, and that the token-passing time is also constant and equal to y. There are n stations in the network. Packets are generated by each station with the same intensity λ, and the intervals between packet arrivals are exponentially distributed.

We wish to determine the mean queue length at a station and the waiting time until a packet is transmitted, [42].

From the perspective of a packet waiting in the queue at any station, the service time is the interval that elapses between the moment this packet begins transmission and the moment when the token returns to the station and enables the transmission of the next packet. This time can be written as

$$B = nx + Ky,$$

where K is a random variable determined by the number of stations that, during a single token circulation cycle, have a non-empty packet queue when the token arrives. Let us assume that the probability that the token finds a station with a non-empty queue is ϱ. This is an approximation, because the token arrival process is not Poissonian, and therefore the probability of observing the system in a given state at the moment of token arrival is not equal to the steady-state probability of that state.

The variable K follows a binomial distribution (cf. Appendix **??**): it represents the number of successes in $(n-1)$ trials, where the probability of success is ϱ. We therefore calculate the mean service time:

$$E[B] = nx + E[K]y = nx + (n-1)\varrho y$$

and the variance

$$\mathrm{var}[B] = \mathrm{var}[nx] + \mathrm{var}[Ky] = 0 + y^2\mathrm{var}[K] = y^2(n-1)\varrho(1-\varrho).$$

Since

$$\varrho = \lambda E[B] = \lambda\big[nx + (n-1)\varrho y\big],$$

we can determine ϱ:

$$\varrho = \frac{\lambda nx}{1 - \lambda(n-1)y}\,.$$

We substitute the values of $E[B]$ and

$$E[B^2] = \mathrm{var}[B] + E[B]^2$$

into formulas (6.2) and (6.4) to obtain the desired values of $E[N]$ and $E[W]$.

■

6.2 System $M/G/1$/PQ

We shall consider two priority schemes:
— **preemptive discipline** (*preemptive*), in which the arrival of a task with a higher priority than the task currently being executed interrupts the execution of the latter. Execution is resumed from the point of interruption once there are no more higher-priority tasks in the system. Within each task class, the *FIFO* policy applies, i.e. service in order of arrival;

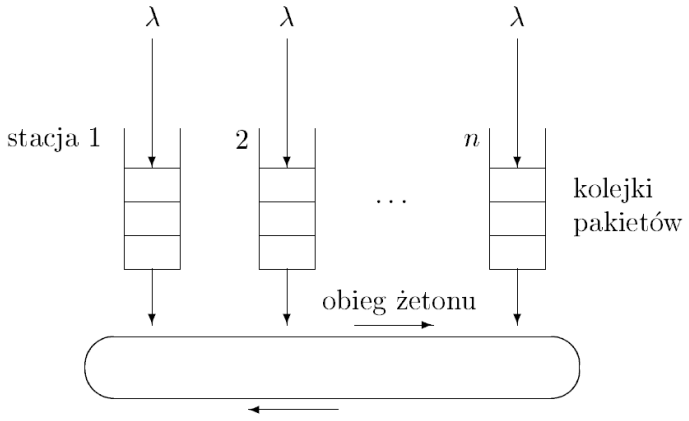


Figure 6.3: Token-passing network mode

— **non-preemptive discipline** (*non-preemptive, head of the line*), in which task processing is not interrupted. Only after the current task has completed is the highest-priority task selected from among those currently waiting. Within each priority class, the *FIFO* policy applies.

In the single-station model considered below, we assume that there are K classes of customers. The arrival stream of class-k customers is a Poisson stream with parameter $\lambda^{(k)}$, and their service time follows an arbitrary distribution $B^{(k)}(x)$ with mean $E[B^{(k)}]$ and variance $\sigma_B^{(k)\,2}$. The load imposed on the station by customers of class k is

$$\varrho^{(k)} = \lambda^{(k)} E[B^{(k)}],$$

and the total load is

$$\varrho = \sum_{k=1}^{K} \varrho^{(k)}.$$

The condition for system stability is, of course,

$$\varrho < 1.$$

We assume that priority decreases as k increases: class 1 has the highest priority, while class K has the lowest. We denote non-preemptive priority disciplines by the symbol *NPQ*, and preemptive priority disciplines by *PPQ*.

We shall determine the mean waiting time in the queue for each customer class; the mean number of waiting customers then follows from Little's formula.

6.2.1 System $M/G/1/NPQ$

The waiting time $W^{(k)}$ of a class-k customer consists of three components:

$W_1^{(k)}$ — the time required to complete service for the customer being served at the moment the customer under consideration arrives,

$W_2^{(k)}$ — the total service time of all customers of equal or higher priority found at the station by the arriving customer,

$W_3^{(k)}$ — the total service time of customers arriving later but having higher priority.

When calculating the residual service time according to formula (6.6), we must take into account all customer classes. The overall service-time distribution is

$$B(x) = \sum_{k=1}^{K} \frac{\lambda^{(k)}}{\Lambda} B^{(k)}(x) \,,$$

where

$$\Lambda = \sum_{k=1}^{K} \lambda^{(k)} \,.$$

The mean of this distribution is

$$E[B] = \sum_{k=1}^{K} \frac{\lambda^{(k)}}{\Lambda} E[B^{(k)}] = \frac{\varrho}{\Lambda},$$

and its second moment is

$$E[B^2] = \frac{1}{\Lambda} \sum_{k=1}^{K} \lambda^{(k)} E\Big[(B^{(k)})^2\Big] \,.$$

Substituting these into formula (6.6), we obtain the expected value of $W_1^{(k)}$, which is the same for all customer classes:

$$E[W_1^{(k)}] = \frac{E[B^2]}{2E[B]} = \frac{1}{2\varrho} \sum_{j=1}^{K} \lambda^{(j)} E\Big[(B^{(j)})^2\Big] \,. \tag{6.7}$$

If $E[L^{(j)}]$ denotes the mean queue length of class-j customers (excluding the customer currently in service), then

$$E[W_2^{(k)}] = \sum_{j=1}^{k} E[L^{(j)}]\, E[B^{(j)}] \,,$$

and since, by Little's theorem,

$$E[L^{(j)}] = \lambda^{(j)} E[W^{(j)}] \,,$$

it follows that

$$E[W_2^{(k)}] = \sum_{j=1}^{k} \lambda^{(j)} E[W^{(j)}] E[B^{(j)}] = \sum_{j=1}^{k} \varrho^{(j)} E[W^{(j)}] \,. \tag{6.8}$$

During the waiting time $W^{(k)}$ of the class-k customer under consideration, new customers arrive. Let $L'^{(j)}$ denote the number of class-j customers arriving during this time. Since the arrival processes are Poisson,

$$E[L'^{(j)}] = \lambda^{(j)} E[W^{(k)}] \,.$$

Customers with higher priorities, i.e. with $j < k$, will be served before the customer under consideration. Therefore,

$$E[W_3^{(k)}] = \sum_{j=1}^{k-1} E[L'^{(j)}]E[B^{(j)}] = E[W^{(k)}] \sum_{j=1}^{k-1} \varrho^{(j)} \; . \tag{6.9}$$

Hence the total waiting time of a class-k customer is

$$E[W^{(k)}] = E[W_1^{(k)}] + E[W_2^{(k)}] + E[W_3^{(k)}] \, .$$

After substituting (6.7)–(6.9) and solving the resulting system of equations, we obtain

$$E[W^{(k)}] = \frac{\frac{1}{2} \sum_{j=1}^{K} \lambda^{(j)} E\left[(B^{(j)})^2\right]}{\left(1 - \sum_{j=1}^{k-1} \varrho^{(j)}\right) \left(1 - \sum_{j=1}^{k} \varrho^{(j)}\right)} \; . \tag{6.10}$$

Example 6.5

Input and output messages are transmitted over a communication line. We assume that the mean length of an input message is 3000 characters, while the mean length of an output message is 9000 characters. The number of characters in an input message follows an Erlang distribution of order 2, whereas the number of characters in an output message follows an Erlang distribution of order 3. The message arrival rates are

$$\lambda^{(1)} = \lambda^{(2)} = 3 \; 1/s,$$

and we assume that both message streams can be regarded as Poisson processes.

Access to the line is organised according to a non-preemptive priority scheme, with priority given to outgoing messages. We wish to calculate the response time of the line for both types of messages.

The mean transmission time of output messages (the first class of tasks) is

$$E[B^{(1)}] = \frac{9000}{60000} = 0.15 \, \mathrm{s} \, ,$$

and the mean transmission time of input messages (the second class of tasks) is

$$E[B^{(2)}] = \frac{3000}{60000} = 0.05 \, \mathrm{s} \, .$$

If the mean of an Erlang distribution of order k is $E[B]$, then its second moment is

$$E[B^2] = \frac{k+1}{k} E[B]^2$$

(cf. Appendix A). Hence

$$\begin{aligned} E[(B^{(1)})^2] &= \frac{3+1}{3} E[B^{(1)}]^2 = \frac{4}{3}(0.15)^2 = 3 \times 10^{-2} \, \mathrm{s}^2 \, , \\ E[(B^{(2)})^2] &= \frac{2+1}{2} E[B^{(2)}]^2 = \frac{3}{2}(0.05)^2 = 3.75 \times 10^{-3} \, \mathrm{s}^2 \, . \end{aligned}$$

The loads generated by the two message classes are

$$\varrho^{(1)} = \lambda^{(1)} E[B^{(1)}] = 3 \times 0.15 = 0.45, \qquad \varrho^{(2)} = \lambda^{(2)} E[B^{(2)}] = 3 \times 0.05 = 0.15.$$

The mean waiting time for access to the line for outgoing messages is

$$E[W^{(1)}] = \frac{\lambda^{(1)} E[(B^{(1)})^2] + \lambda^{(2)} E[(B^{(2)})^2]}{2(1-\varrho^{(1)})} = \frac{3 \times 3 \times 10^{-2} + 3 \times 3.75 \times 10^{-3}}{2(1-0.45)} = 0.092\,\mathrm{s}\,,$$

and for input messages

$$E[W^{(2)}] = \frac{\lambda^{(1)} E[(B^{(1)})^2] + \lambda^{(2)} E[(B^{(2)})^2]}{2(1-\varrho^{(1)})(1-\varrho^{(1)}-\varrho^{(2)})} = \frac{3 \times 3 \times 10^{-2} + 3 \times 3.75 \times 10^{-3}}{2(1-0.45)(1-0.45-0.15)} = 0.154\,\mathrm{s}\,.$$

The response time is the waiting time plus the transmission time:

$$\begin{aligned} E[R^{(1)}] &= E[W^{(1)}] + E[B^{(1)}] = 0.092 + 0.15 = 0.242\,\mathrm{s}\,, \\ E[R^{(2)}] &= E[W^{(2)}] + E[B^{(2)}] = 0.154 + 0.05 = 0.204\,\mathrm{s}\,. \end{aligned}$$

■

6.2.2 System $M/G/1/PPQ$

In the case of the preemptive-resume discipline, when calculating the residual service time we take into account only tasks whose priority is equal to or higher than that of the task under consideration:

$$E[W_1^{(k)}] = \frac{1}{2} \sum_{j=1}^{k} \lambda^{(j)} E\left[(B^{(j)})^2\right]. \tag{6.11}$$

The remaining components of the waiting time remain unchanged, and the final result is

$$E[W^{(k)}] = \frac{\displaystyle\sum_{j=1}^{k} \lambda^{(j)} E\left[(B^{(j)})^2\right]}{2\left(1-\sum_{j=1}^{k-1}\varrho^{(j)}\right)\left(1-\sum_{j=1}^{k}\varrho^{(j)}\right)}\,. \tag{6.12}$$

However, it must be remembered that this is the waiting time until the start of service, which may then be interrupted several times by tasks of higher priority.

The average *task execution time*, i.e. the time between the start and the completion of service, denoted by $E[C^{(k)}]$, is calculated by noting that the service station can devote only the fraction

$$1 - \sum_{j=1}^{k-1} \varrho^{(j)}$$

of its capacity to serving tasks of priority k. Hence

$$E[C^{(k)}] = \frac{E[B^{(k)}]}{1-\sum_{j=1}^{k-1}\varrho^{(j)}}\,. \tag{6.13}$$

Therefore, the response time of the station for customers of class k is

$$E[R^{(k)}] = E[W^{(k)}] + E[C^{(k)}]\,. \tag{6.14}$$

Example 6.6

Access to the central processing unit is organised according to an absolute priority scheme. The highest priority ($k = 1$) is assigned to terminal transmission handling, the next ($k = 2$) to communication with external memory, and the lowest ($k = 3$) to message processing.

The parameters for the individual task classes are as follows. The arrival streams are Poisson, with

$$\lambda^{(1)} = 10\ 1/s, \qquad \lambda^{(2)} = 60\ 1/s, \qquad \lambda^{(3)} = 10\ 1/s.$$

The mean service times are

$$E[B^{(1)}] = 0.015\ s, \qquad E[B^{(2)}] = 0.003\ s, \qquad E[B^{(3)}] = 0.03\ s.$$

We assume that the service time of first-class tasks follows a second-order Erlang distribution, the service time of second-class tasks is constant, and the service time of the lowest-priority tasks is exponentially distributed. We wish to calculate the characteristic response time for each class.

We calculate:

$$\begin{aligned}
E\Big[(B^{(1)})^2\Big] &= \frac{2+1}{2}\,E[B^{(1)}]^2 = \frac{3}{2}(1.5\times 10^{-2}\,\mathrm{s})^2 = 3.375\times 10^{-4}\,\mathrm{s}^2\,,\\
E\Big[(B^{(2)})^2\Big] &= E[B^{(2)}]^2 = (3\times 10^{-3}\,\mathrm{s})^2 = 9\times 10^{-6}\,\mathrm{s}^2\,,\\
E\Big[(B^{(3)})^2\Big] &= 2\,E[B^{(3)}]^2 = 2(3\times 10^{-2}\,\mathrm{s})^2 = 1.8\times 10^{-3}\,\mathrm{s}^2\,.
\end{aligned}$$

The utilisations of the central processing unit by the individual task classes are

$$\begin{aligned}
\varrho^{(1)} &= \lambda^{(1)}E[B^{(1)}] = 10\,\frac{1}{\mathrm{s}}\times 0.015\,\mathrm{s} = 0.15\,,\\
\varrho^{(2)} &= \lambda^{(2)}E[B^{(2)}] = 60\,\frac{1}{\mathrm{s}}\times 0.003\,\mathrm{s} = 0.18\,,\\
\varrho^{(3)} &= \lambda^{(3)}E[B^{(3)}] = 10\,\frac{1}{\mathrm{s}}\times 0.03\,\mathrm{s} = 0.30\,,
\end{aligned}$$

and the total utilisation is

$$\varrho = \varrho^{(1)} + \varrho^{(2)} + \varrho^{(3)} = 0.63\,.$$

The mean waiting times until the start of service are:

$$\begin{aligned}
E[W^{(1)}] &= \frac{\lambda^{(1)}E\big[(B^{(1)})^2\big]}{2(1-\varrho^{(1)})} = \frac{10\times 3.375\times 10^{-4}}{2(1-0.15)} \approx 0.0020\,\mathrm{s}\,,\\
E[W^{(2)}] &= \frac{\lambda^{(1)}E\big[(B^{(1)})^2\big] + \lambda^{(2)}E\big[(B^{(2)})^2\big]}{2(1-\varrho^{(1)})(1-\varrho^{(1)}-\varrho^{(2)})}\\
&= \frac{10\times 3.375\times 10^{-4} + 60\times 9\times 10^{-6}}{2(1-0.15)(1-0.15-0.18)} \approx 0.0031\,\mathrm{s}\,,\\
E[W^{(3)}] &= \frac{\lambda^{(1)}E\big[(B^{(1)})^2\big] + \lambda^{(2)}E\big[(B^{(2)})^2\big] + \lambda^{(3)}E\big[(B^{(3)})^2\big]}{2(1-\varrho^{(1)}-\varrho^{(2)})(1-\varrho^{(1)}-\varrho^{(2)}-\varrho^{(3)})}\\
&= \frac{10\times 3.375\times 10^{-4} + 60\times 9\times 10^{-6} + 10\times 1.8\times 10^{-3}}{2(1-0.15-0.18)(1-0.15-0.18-0.30)} \approx 0.0261\,\mathrm{s}\,.
\end{aligned}$$

6.3 System $G/M/1$

The analysis of this system requires solving the equation

$$\vartheta = \bar{a}(\mu - \mu\vartheta)\,, \tag{6.15}$$

where $\bar{a}(s)$ is the Laplace transform of the density $a(x)$. If the system has a steady state, i.e. if the utilisation factor $\varrho < 1$, then the interval $(0,1)$ contains exactly one root ϑ of equation (6.15) [114].

This root is the parameter of the waiting-time distribution for the station. The cumulative distribution function $W(x)$ has the form

$$W(x) = 1 - \vartheta e^{-\mu(1-\vartheta)x}\,, \tag{6.16}$$

from which we obtain

$$E[W] = \frac{\vartheta}{\mu(1-\vartheta)}\,. \tag{6.17}$$

The distribution of the number of customers in the system is geometric:

$$p(n) = (1-\vartheta)\vartheta^n\,.$$

Thus, the average waiting time in the queue of a $G/M/1$ system depends on the form of the input distribution, and not only on its first two moments.

Example 6.7

Let us assume that the service-time distribution is exponential, while the interarrival-time distribution $A(x)$ is Erlang when $C_A^2 < 1$, or second-order hyperexponential when $C_A^2 > 1$, with density

$$a(x) = 2\alpha^2\lambda e^{-2\alpha\lambda x} + 2(1-\alpha)^2\lambda e^{-2(1-\alpha)\lambda x}\,. \tag{6.18}$$

By varying the parameter $\alpha \in (0, 0.5)$, we obtain different values of the squared coefficient of variation C_A^2. The mean of the distribution is

$$E[A] = \frac{1}{\lambda}\,.$$

We wish to calculate the mean waiting time using formula (6.17) and compare it with the mean waiting time in an $M/M/1$ system

$$(C_A^2 = 1),$$

for which the mean values of $A(x)$ and $B(x)$ are the same as in the system under consideration.

For an Erlang interarrival-time distribution of order k, equation (6.15) takes the form

$$\vartheta = \left(\frac{k\lambda}{\mu - \mu\vartheta + k\lambda}\right)^k.$$

For the hyperexponential distribution (6.18), it becomes

$$\vartheta = \frac{2\alpha^2\lambda}{\mu - \mu\vartheta + 2\alpha\lambda} + \frac{2(1-\alpha)^2\lambda}{\mu - \mu\vartheta + 2(1-\alpha)\lambda}\,.$$

After solving these equations numerically, we obtain the results shown in Table 6.1. ■

The above results indicate that, for a fixed value of λ, the waiting time increases almost linearly with increasing C_A^2.

ϱ \ C_A^2	0.00	0.25	0.50	1.00	2.00	3.00	4.00	5.00	10.00
0.01	0.00	0.00	0.04	1.00	1.34	1.51	1.61	1.67	1.83
0.25	0.06	0.23	0.46	1.00	1.39	1.65	1.83	1.93	2.35
0.50	0.26	0.43	0.62	1.00	1.45	1.83	2.16	2.46	3.68
0.75	0.40	0.55	0.70	1.00	1.49	1.97	2.42	2.88	5.17
0.99	0.50	0.62	0.75	1.00	1.50	2.01	2.49	2.99	5.47

Table 6.1: $\frac{E[W]_{G/M/1}}{E[W]_{M/M/1}}$ as a function of C_A^2 and ϱ; $\lambda = 1$, and in both queueing systems the service-time distribution $b(x)$ is the same

6.4 System $G/G/1$

The determination of the waiting time in the queue of a $G/G/1$ system involves solving the Lindley equation [74], which is an integral equation of Wiener–Hopf type:

$$w(x) = \begin{cases} \displaystyle\int_{-\infty}^{x} w(x-\xi)\, dF(\xi) & \text{for } x \geq 0\,, \\ 0 & \text{for } x < 0\,, \end{cases} \tag{6.19}$$

where

$$F(\xi) = \int_0^{\infty} b(\xi + y) a(y)\, dy\,. \tag{6.20}$$

The spectral factorisation method [94], [95] indicates a possible route to solving equation (6.19), but imposes a number of restrictions on the functions $a(x)$ and $F(x)$ and does not lead to final results in the form of an explicit dependence of the waiting time or queue length on the parameters of the distributions $A(x)$ and $B(x)$. Consequently, it is of limited practical value.

Specifically, it is required that

$$\lim_{x\to\infty} \frac{a(x)}{e^{-Dx}} < \infty\,,$$

for any positive real constant D, and that the Laplace transform of the function $F(x)$ can be written in the form

$$\bar{F}(s) = 1 + \frac{\Psi_+(s)}{\Psi_-(s)}\,,$$

where, for $\Re(s) < 0$, the function $\Psi_+(s)$ is analytic and has no zeros in the right half-plane of the complex variable s, while $\Psi_-(s)$ is analytic and has no zeros for $\Re(s) > 0$.

The functions $\Psi_+(s)$ and $\Psi_-(s)$ must satisfy the relations

$$\lim_{|s|\to\infty} \frac{\Psi_+(s)}{s} = 1 \qquad \text{for } \Re(s) < 0\,, \qquad \lim_{|s|\to\infty} \frac{\Psi_-(s)}{s} = 1 \qquad \text{for } \Re(s) > 0\,. \tag{6.21}$$

The transform of the waiting-time distribution in the queue then has the form

$$\bar{w}(s) = \frac{s\bar{w}(0)}{\Psi_+(s)}\,,$$

where

$$\bar{w}(0) = \lim_{s\to 0} \frac{\Psi_+(s)}{s} = (1-\varrho) E[A] \Psi_-(0)\,.$$

Example 6.8

Let $a(x)$ and $b(x)$ be hyperexponential distributions of the same form as in the previous example. Assume that $\lambda = 1$, $\mu = 2$, and

$$C_A^2 = C_B^2 = 5\,,$$

so that for both functions $a(x)$ and $b(x)$ the coefficient α in formula (6.18) has the value 0.092.

The transform of the function $F(x)$ is then

$$\bar{F}(s) = \bar{a}(-s)\bar{b}(s) = \left(\frac{0.02}{0.18 - s} + \frac{0.17}{1.82 - s}\right)\left(\frac{0.03}{0.37 + s} + \frac{0.33}{3.63 + s}\right)\,,$$

from which we obtain

$$\Psi_+(s) = \frac{s(s - 0.49)}{(s - 0.18)(s - 1.82)}\,, \qquad \Psi_-(s) = -\frac{(s + 0.37)(s + 3.63)}{s + 0.20}\,.$$

The function

$$\frac{\Psi_+(s)}{s}$$

converges to zero as $|s| \to \infty$, rather than to unity, as required by condition (6.21). Therefore, it is not possible to calculate the transform $\bar{w}(s)$ of the waiting-time distribution in this case. ■

However, the distribution $\bar{w}(s)$ can be calculated using this method for the $G/E_k/1$ system [96]. Tabulated, numerically computed values of $E[W]$ for the system $E_k/E_k/1$ are given by Page [93]. Herzog, Woo and Chandy [45] propose a numerical solution of various single-station models through the recursive solution of a system of simultaneous equations for the state probabilities $p(n)$.

6.4.1 Some approximations related to the $G/G/1$ system

Bounds on the waiting-time distribution

Let the random variable A_k denote the interarrival time between customers z_k and z_{k+1}, and let B_k denote the service time of customer z_k.

Define the random variable

$$U_k = B_k - A_k\,,$$

and note that $-U_k$ is the amount of time for which the station would remain idle after completing the service of customer z_k, assuming that service began immediately upon arrival.

The limiting random variable

$$U = \lim_{k \to \infty} U_k$$

in steady state has density

$$u(x) = \int_{-\infty}^{\infty} b(x + \tau)a(\tau)\,d\tau$$

and mean

$$E[U] = \frac{1}{\mu} - \frac{1}{\lambda}\,, \qquad E[U] < 0 \qquad \text{for } \varrho < 1\,.$$

As shown by Kingman [59], [60], the waiting-time distribution may be bounded by exponential distributions with appropriately chosen parameters. If Θ satisfies the equation

$$E[e^{\Theta U}] = 1 \,,$$

then the cumulative distribution function

$$W(x) = P(W \leq x)$$

satisfies

$$1 - e^{-\Theta x} \leq W(x) \leq 1 - \eta e^{-\Theta x} \,, \tag{6.22}$$

where

$$\eta = \inf_{r \geq 0} \frac{\displaystyle\int_r^\infty u(y)\, dy}{\displaystyle\int_r^\infty e^{\Theta(y-r)} u(y)\, dy} \,.$$

The lower bound in inequality (6.22) can be sharpened [106]:

$$1 - \xi e^{-\Theta x} \leq W(x) \leq 1 - \eta e^{-\Theta x} \,, \tag{6.23}$$

where

$$\xi = \sup_{r \geq 0} \frac{\displaystyle\int_r^\infty u(y)\, dy}{\displaystyle\int_r^\infty e^{\Theta(y-r)} u(y)\, dy} \,.$$

There also exists a weaker but more convenient inequality, because it does not require knowledge of the distribution $u(x)$:

$$1 - \xi' e^{-\Theta x} \leq W(x) \leq 1 - \eta' e^{-\Theta x} \,, \tag{6.24}$$

where

$$\begin{aligned} \xi' &= \sup_{r \geq 0} \frac{\displaystyle\int_r^\infty b(y)\, dy}{\displaystyle\int_r^\infty e^{\Theta(y-r)} b(y)\, dy} \,, \\ \eta' &= \inf_{r \geq 0} \frac{\displaystyle\int_r^\infty b(y)\, dy}{\displaystyle\int_r^\infty e^{\Theta(y-r)} b(y)\, dy} \,. \end{aligned}$$

The parameter Θ is defined in the same way as before.

Bounds on the mean waiting time

(a) Kingman's upper bound for $E[W]$.

A considerably simpler expression than inequalities (6.22) and (6.23) is the upper bound for the mean waiting time given by Kingman [58].

According to the definition of U_k, the waiting time in queue for customer z_{k+1} is

$$W_{k+1} = \max[0, W_k + U_k] \,. \tag{6.25}$$

In steady state,

$$W = \max[0, U + W] \,. \tag{6.26}$$

Introduce the notation

$$X^+ = \max[0, X] \,, \qquad X^- = \min[0, X] \,.$$

It can be shown that for any random variable X,

$$\sigma_X^2 = \sigma_{X^+}^2 + \sigma_{X^-}^2 + 2E[X^+]E[X^-] \,. \tag{6.27}$$

Substituting $X = U + W$ and recalling that W and $(U + W)^+$ have the same distribution, hence also the same variance,

$$\sigma_W^2 = \sigma_{(U+W)^+}^2 \,,$$

we obtain from equation (6.27)

$$E[W] = \frac{\sigma_W^2}{2E[-U]} - \frac{\sigma_{(U+W)^-}^2}{2E[-U]} \,.$$

Since U is the difference of two independent random variables,

$$\sigma_U^2 = \sigma_A^2 + \sigma_B^2 \,.$$

By omitting the second term, which is non-negative for $\varrho < 1$, we obtain the upper bound

$$E[W] \le w_G = \frac{\lambda(\sigma_A^2 + \sigma_B^2)}{2(1 - \varrho)} \,.$$

As $\varrho \to 1$, the inequality becomes exact.

(b) Marshall's lower bound for $E[W]$.

According to (6.25), when calculating the conditional expectation

$$E[W_{k+1} \mid W_k = v]$$

we consider only the case $U_k \ge -v$. For $U_k \ge -v$, we have

$$W_{k+1} = v + U_k \,,$$

hence

$$\begin{aligned} E[W_{k+1} \mid W_k = v] &= \int_{x=-v}^{\infty} (v + x) u(x) \, dx \\ &= \int_{x=-v}^{\infty} [1 - U(x)] \, dx \\ &\equiv g(v) \,, \end{aligned}$$

where $u(x)$ is the probability density function and $U(x)$ is the cumulative distribution function of the random variable U.

Taking expectations with respect to the distribution of W_k gives

$$\begin{aligned} E[W_{k+1}] &= \int_0^\infty E[W_{k+1} \mid W_k = v]\, w_k(v)\, dv \\ &= \int_0^\infty g(v) w_k(v)\, dv\,, \end{aligned}$$

that is,

$$E[W_{k+1}] = E[g(W_k)]\,.$$

In steady state,

$$E[W] = E[g(W)]\,. \tag{6.28}$$

It can be shown [79] that the function $g(v)$ is convex and therefore satisfies Jensen's inequality

$$E[g(W)] \geq g(E[W])\,. \tag{6.29}$$

Furthermore, there exists a non-negative root w_D of the equation

$$v = g(v)\,,$$

which, by (6.28) and (6.29), is not greater than $E[W]$ [78]:

$$E[W] \geq w_D\,. \tag{6.30}$$

(c) Török upper and lower bounds for $E[W]$.

Using known relationships between the duration of the idle period and the number of customers served during the busy period [17], [58], together with his own estimates of idle-period duration, Török [116] derived the bounds

$$\frac{E[U^2]}{-2E[U]} - \frac{K_2}{2K_0K_1} \leq E[W] \leq \frac{E[U^2]}{-2E[U]} - \frac{K_2K_0}{2K_1}\,, \tag{6.31}$$

where

$$K_i = \left| \int_{-\infty}^0 y^i u(y)\, dy \right|, \qquad i = 0, 1, 2\,.$$

Approximations of the average waiting time

(a) The Kramers–Langenbach–Belz (KLB) approximation [67]

$$E[W] \approx \frac{\varrho(C_A^2 + C_B^2)}{2\mu(1-\varrho)} \begin{cases} \exp\left[\dfrac{2(1-\varrho)}{3\varrho} \dfrac{(1-C_A^2)^2}{C_A^2 + C_B^2}\right] & \text{for } C_A^2 \leq 1\,, \\ \exp\left[-(1-\varrho) \dfrac{C_A^2 - 1}{C_A^2 + 4C_B^2}\right] & \text{for } C_A^2 \geq 1\,. \end{cases} \tag{6.32}$$

$\frac{E[W]_{G/G/1}}{E[W]_{M/M/1}}$	$C_A^2 = 5,\ C_B^2 = 5$	$C_A^2 = 5,\ C_B^2 = 10$	$C_A^2 = 10,\ C_B^2 = 5$	$C_A^2 = 10,\ C_B^2 = 10$
$\varrho = 0.01$	4.09	7.60	5.41	10.21
$\varrho = 0.25$	4.24	7.53	5.73	10.06
$\varrho = 0.50$	4.52	7.47	6.44	9.89
$\varrho = 0.75$	4.91	7.26	7.09	9.75

Table 6.2: Ratio of the average waiting time in the $G/G/1$ system to the average waiting time in the $M/M/1$ system with the same values of $E[A]$ and $E[B]$

(b) Page's approximation [97]

$$E[W] \approx \frac{1}{\lambda}\left[C_A^2 C_B^2 E[N]_{M/M/1} + C_A^2(1-C_B^2)E[N]_{M/D/1} + (1-C_A^2)C_B^2 E[N]_{D/M/1}\right] . \quad (6.33)$$

(c) Diffusion approximation — discussed in the next chapter:

$$E[W] \approx E[B]\frac{C_A^2\varrho + C_B^2}{2(1-\varrho)} . \quad (6.34)$$

(d) Approximation by identification of simulation results.

Most approximate models of the $G/G/1$ system are second-order approximations, taking into account the influence of the first two moments of the distributions $A(x)$ and $B(x)$.

We have already seen that the influence of the variance of one of these distributions on the behaviour of the systems $M/G/1$ and $G/M/1$ is significant. In the $M/G/1$ system, the dependence of the average waiting time on the variance of $B(x)$ is linear, while in the $G/M/1$ system the dependence on the variance of $A(x)$ is nearly linear.

Table 6.2 illustrates the influence of the variances of the distributions $A(x)$ and $B(x)$ — assuming the hyperexponential form (6.18) — on the average waiting time in the $G/G/1$ system.

The table contains values of $E[W]$ in the $G/G/1$ system obtained by simulation and expressed relative to the average waiting time in the $M/M/1$ system with the same values of $E[A]$ and $E[B]$. Thus, the function of interest is

$$\varphi(\varrho, C_A^2, C_B^2) = \frac{E[W]_{G/G/1}}{E[W]_{M/M/1}} .$$

The form of this function can be estimated on the basis of simulation data.

Reference [19] proposes the approximation

$$\varphi(\varrho, C_A^2, C_B^2) = a(\varrho)C_A^2 + b(\varrho)C_B^2 + d(\varrho)C_A^2C_B^2 + e(\varrho)\,, \tag{6.35}$$

where $a(\varrho)$–$e(\varrho)$ are third-degree polynomials. The coefficients of these polynomials were determined by least-squares fitting:

$$\begin{aligned} a(\varrho) &= -1.349\varrho^3 + 1.987\varrho^2 - 0.186\varrho + 0.050\,, \\ b(\varrho) &= -0.134\varrho^3 + 0.199\varrho^2 - 0.019\varrho + 0.455\,, \\ e(\varrho) &= -b(\varrho)\,, \\ d(\varrho) &= 1.439\varrho^3 - 1.987\varrho^2 + 0.186\varrho + 0.450\,. \end{aligned} \tag{6.36}$$

Figure 6.3 compares exact results (shown with a thick line) with several approximations [19].

The values of $E[W]_{G/G/1}$ are shown relative to $E[W]_{M/M/1}$ for systems having the same values of $E[A]$ and $E[B]$.

The exact results naturally depend on the specific forms of $A(x)$ and $B(x)$. In the example shown in the figure, both distributions are second-order hyperexponential distributions of the form (6.18).

For this case, the approximation based on simulation-result identification is practically indistinguishable from the exact results at the scale of the figure.

The accuracy of the various approximations depends strongly on the server utilisation factor ϱ. When ϱ is close to one, almost all approximations (with the exception of Marshall's lower bound) provide results very close to the exact value.

When $\varrho = 0.25$, only the diffusion approximation, the KLB approximation, and the Kingman–Ross upper bound based on (6.24) remain reasonably accurate.

The approximation error increases approximately linearly with the coefficients of variation C_A^2 and C_B^2.

Recall that all estimates of the average waiting time can easily be translated into estimates of the average number of customers at the station using Little's law:

$$E[N] = \frac{E[R]}{E[A]} = \frac{E[W] + E[B]}{E[A]} = \frac{E[W]}{E[A]} + \varrho\,.$$

The most important conclusion from these results is that there is a strong, approximately linear relationship between the average waiting time, the average number of customers at the station, and the second moments of the distributions $A(x)$ and $B(x)$.

For example, if in a real system $C_A^2 = 10$ and $C_B^2 = 10$, but only the mean values of the distributions are measured and the system is approximated by an $M/M/1$ model (that is, by assuming $C_A^2 = C_B^2 = 1$), then the resulting error in estimating $E[N]$ and $E[W]$ may reach approximately 1000%.

Substantially smaller errors can be achieved by using one of the above approximations together with carefully prepared input data.

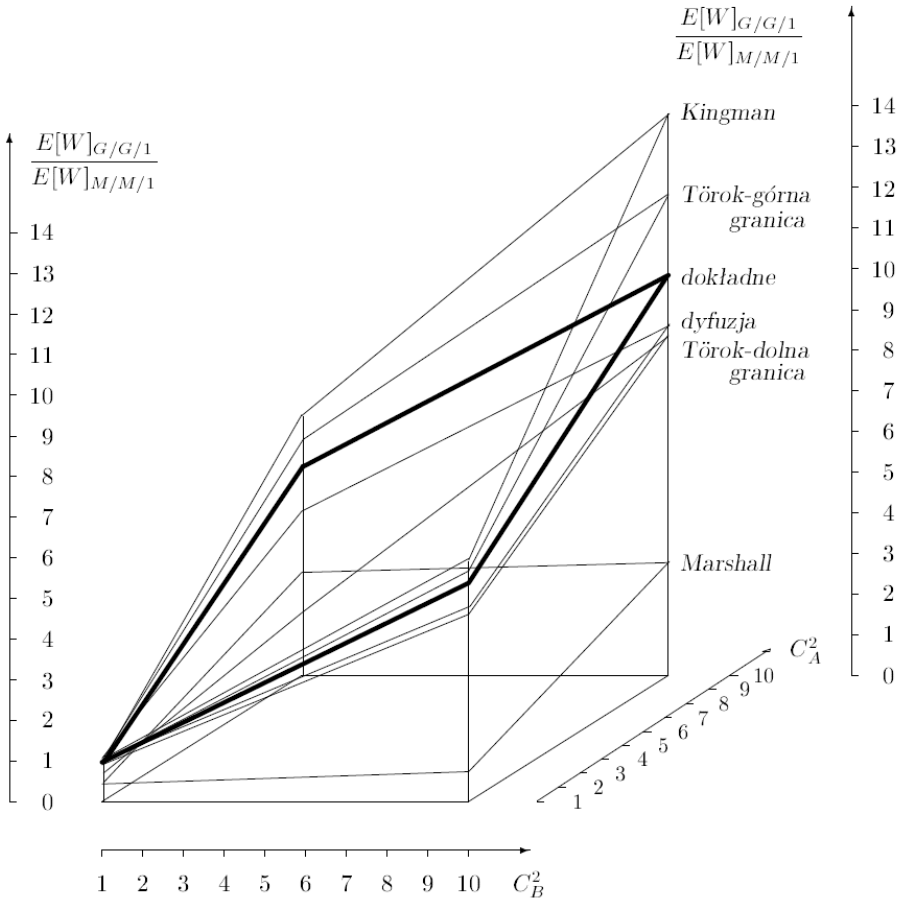


Figure 6.4: Comparison of the values of $E[W]$ given by various approximations for the $G/G/1$ system, $\varrho = 0.75$

Chapter 7

Diffusion approximation, basic models

As we already know, there is no practical exact model for a system of type $G/G/1$. Nor is there an exact model for a system with a natural service discipline and multiple customer classes with different service times, even if the service-time distributions are exponential for all classes and differ only in the value of the parameter μ. This is a major drawback, since queues with natural service disciplines are among the most commonly used models.

In the previous chapter we examined several approximate descriptions of the $G/G/1$ system; this chapter is devoted to a more detailed discussion of the most useful and widely used approach — the diffusion approximation.

The basis of this approximate method is the replacement of the discrete process $N(t)$, representing the number of customers at the service station, by a continuous diffusion process $X(t)$ with similar properties. By solving the diffusion equation with appropriately chosen parameters and boundary conditions, we obtain a function describing the diffusion process that approximates the probability distribution of the queue length.

This is a second-order approximation, i.e. the method does not require information about the exact form of the service-time distribution $B(x)$ or the interarrival-time distribution $A(x)$, but only about the first two moments of these distributions. This means that the results of the approximation are identical for different distributions, provided they have the same mean and variance, for example the distributions $A(x)$, $B(x)$ and $A'(x)$, $B'(x)$.

Note that the data required by the model — the first two moments of the distributions — are easier to obtain while monitoring the computer system under investigation than, for example, complete histograms of these distributions.

We will discuss approximations for the single-server systems $G/G/1$ and $G/G/1/N$ (a finite-capacity queue), with one or several classes of customers, in both steady-state and transient conditions, as well as open and closed networks of $G/G/1$ systems.

7.1 Diffusion approximation of the $G/G/1$ queueing system

Customers arrive at the service station at independent intervals $a_1, a_2, \ldots, a_k, \ldots$, whose distribution $A(x)$ has mean $1/\lambda$ and variance σ_A^2. Let us define

$$\tau_k = \sum_{i=1}^{k} a_i \, .$$

Since τ_k is the sum of independent random variables having the same distribution, the central limit theorem implies that the distribution of the variable

$$\frac{\tau_k - k/\lambda}{\sigma_A \sqrt{k}}$$

approaches the normal distribution as k increases:

$$\lim_{k\to\infty} P\left[\frac{\tau_k - k/\lambda}{\sigma_A\sqrt{k}} \le x\right] = \Phi(x) \, ,$$

where

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\xi^2/2} \, d\xi \, .$$

Thus, for large values of k,

$$P\left[\tau_k \le x\sigma_A\sqrt{k} + \frac{k}{\lambda}\right] \approx \Phi(x) \, .$$

Let

$$t = x\sigma_A\sqrt{k} + \frac{k}{\lambda} \, ,$$

that is,

$$k = t\lambda - x\sigma_A\sqrt{k}\,\lambda \, .$$

For large values of k, we may approximate

$$k \approx t\lambda, \qquad \sqrt{k} \approx \sqrt{t\lambda} \, .$$

Let $H(t)$ denote the number of customers who have arrived at the station during a time interval of length t. Observe that

$$P[H(t) \ge k] = P[\tau_k \le t] \, .$$

Therefore,

$$\begin{aligned} \Phi(x) &\approx P\left[\tau_k \le x\sigma_A\sqrt{k} + \frac{k}{\lambda}\right] = P[H(t) \ge k] \\ &= P\left[H(t) \ge t\lambda - x\sigma_A\sqrt{t\lambda}\,\lambda\right] \\ &= P\left[\frac{H(t) - t\lambda}{\sigma_A\sqrt{t\lambda}\,\lambda} \ge -x\right] \, . \end{aligned}$$

Since for the normal distribution

$$\Phi(x) = 1 - \Phi(-x),$$

that is,

$$P[\xi \geq -x] = P[\xi \leq x]\,,$$

we obtain

$$P\left[\frac{H(t) - t\lambda}{\sigma_A \sqrt{t\lambda}\,\lambda} \leq x\right] \approx \Phi(x)\,.$$

Thus, the number of customers arriving at the station during an interval of length t (provided t is sufficiently large) is approximately normally distributed with mean

$$\lambda t$$

and variance

$$\sigma_A^2 \lambda^3 t\,.$$

If the station operates continuously, customers depart from it at time intervals $b_1, b_2, \ldots, b_k, \ldots$ equal to their service times, which are independent random variables with distribution $B(x)$.

Therefore, the number of customers departing from the station during a time interval of length t is approximately normally distributed with mean

$$\mu t$$

and variance

$$\sigma_B^2 \mu^3 t\,,$$

provided that the interval is sufficiently long and that the station remains continuously busy during that period.

The difference of two independent normally distributed random variables is also normally distributed, with mean equal to the difference of the means and variance equal to the sum of the variances.

Therefore, the change in the number of customers in the system during a time interval of length t,

$$N(t) - N(0),$$

is approximately normally distributed with mean

$$(\lambda - \mu)t$$

and variance

$$(\sigma_A^2 \lambda^3 + \sigma_B^2 \mu^3)t\,.$$

The diffusion process $X(t)$, whose probability density function $f(x, t; x_0)$ satisfies

$$f(x, t; x_0)\, dx = P\left[x \leq X(t) < x + dx \mid X(0) = x_0\right],$$

is defined by the diffusion equation

$$\frac{\partial f(x, t; x_0)}{\partial t} = \frac{\alpha}{2} \frac{\partial^2 f(x, t; x_0)}{\partial x^2} - \beta \frac{\partial f(x, t; x_0)}{\partial x}\,, \tag{7.1}$$

and has the property that its infinitesimal changes

$$dX(t) = X(t + dt) - X(t)$$

follow a normal distribution with mean

$$\beta\, dt$$

and variance

$$\alpha\, dt\,.$$

By choosing the coefficients of the diffusion equation (8) as

$$\begin{aligned} \beta &= \lambda - \mu\,, \\ \alpha &= \sigma_A^2\lambda^3 + \sigma_B^2\mu^3 = C_A^2\lambda + C_B^2\mu\,, \end{aligned} \tag{7.2}$$

we obtain a process $X(t)$ whose variations over time have mean and variance that depend on the observation interval in the same way as in the process $N(t)$.

The processes $N(t)$ and $X(t)$ are obviously not identical: $N(t)$ is discrete, whereas $X(t)$ is continuous; for $N(t)$, the interval over which the changes become approximately normal must be sufficiently long, while for $X(t)$ the normal approximation applies to infinitesimal intervals.

Nevertheless, there is a clear similarity between the two processes, and replacing $N(t)$ by the process $X(t)$, as proposed by Newell [87, 88], provides a reasonable approximation for the $G/G/1$ system.

A more formal justification for the diffusion approximation is provided by limit theorems for the $G/G/1$ system due to Iglehart and Whitt [51, 50]. Namely, if $\hat{N}_n$ is a sequence of random variables derived from $N(t)$:

$$\hat{N}_n = \frac{N(nt) - (\lambda - \mu)nt}{(\sigma_A^2\lambda^3 + \sigma_B^2\mu^3)\sqrt{n}}\,,$$

then this sequence converges weakly (i.e. converges in distribution) to ξ, where $\xi(t)$ is a standard Wiener process (i.e. a diffusion process with $\beta = 0$ and $\alpha = 1$), provided that $\varrho > 1$, that is, provided that the system is overloaded and has no equilibrium state.

When $\varrho = 1$, the sequence $\hat{N}_n$ converges to ξ_R. The process $\xi_R(t)$ is the reflected Wiener process obtained from $\xi(t)$ by restricting it to the interval $x \geq 0$:

$$\xi_R(t) = \xi(t) - \inf\{\xi(u) \,:\, 0 \leq u \leq t\}\,.$$

A system with $\varrho \geq 1$ has no steady state. The number of customers grows over time with fluctuations around a linear trend, just as the value of a diffusion process with positive drift coefficient β increases.

There are no corresponding limit theorems for equilibrium states with $\varrho < 1$; in this case we must rely on heuristic arguments confirming the usefulness of the approximation.

Apart from the problem of selecting appropriate coefficients, we must also consider the boundary conditions of the diffusion equation. The process $N(t)$ never takes negative values, so the diffusion process $X(t)$ must also be restricted to the region $x \geq 0$.

The simplest approach is to introduce a *reflecting barrier* at the point $x = 0$. Since the process is restricted to the positive half-line, we must have

$$\int_0^\infty f(x, t; x_0)\, dx = 1\,,$$

that is,

$$\frac{\partial}{\partial t}\int_0^\infty f(x,t;x_0)\,dx = \int_0^\infty \frac{\partial f(x,t;x_0)}{\partial t}\,dx = 0\,.$$

Substituting the right-hand side of the diffusion equation, we obtain the boundary condition corresponding to a reflecting barrier at the origin:

$$\lim_{x\to 0}\left[\frac{\alpha}{2}\frac{\partial f(x,t;x_0)}{\partial x} - \beta f(x,t;x_0)\right] = 0\,. \tag{7.3}$$

The solution of equation (8) with boundary condition (7.3) has the form [64]

$$f(x,t;x_0) = \frac{\partial}{\partial x}\left[\Phi\left(\frac{x-x_0-\beta t}{\sqrt{\alpha t}}\right) - e^{\frac{2\beta x}{\alpha}}\Phi\left(\frac{-x-x_0-\beta t}{\sqrt{\alpha t}}\right)\right]\,. \tag{7.4}$$

If the system is stable, $\varrho < 1$, i.e. $\beta < 0$, then the system has a steady state. The dependence of the density function on time vanishes:

$$\lim_{t\to\infty} f(x,t;x_0) = f(x)\,,$$

and the partial differential equation (8) becomes the ordinary differential equation

$$0 = \frac{\alpha}{2}\frac{d^2 f(x)}{dx^2} - \beta\frac{df(x)}{dx}\,, \tag{7.5}$$

whose solution is

$$f(x) = -\frac{2\beta}{\alpha}e^{\frac{2\beta x}{\alpha}}\,. \tag{7.6}$$

The obtained solution approximates the queue-length distribution in the $G/G/1$ system:

$$p(n,t;n_0) \approx f(n,t;n_0),$$

and in the steady state

$$p(n) \approx f(n)\,.$$

One may also average the values of the function $f(x)$, for example by taking, [64]

$$p(0) \approx \int_0^{0.5} f(x)\,dx,$$

and

$$p(n) \approx \int_{n-0.5}^{n+0.5} f(x)\,dx, \qquad n = 1,2,\ldots\,.$$

A reflecting barrier does not allow the process to remain at the point $x = 0$; the process is reflected immediately. Therefore, the solutions (7.4) and (7.6) are valid only for $x > 0$; at $x = 0$ the density is zero, i.e.

$$f(0) = 0\,.$$

As a result, this approximation gives good results when the system is heavily loaded, that is, when the utilisation factor ϱ is close to unity and the probability $p(0)$ of an empty system is negligibly small.

The largest discrepancies from exact results occur near the origin, where the behaviour of the reflecting barrier differs from the actual behaviour of the queueing system.

The results can be improved by renormalisation [64] or by shifting the barrier to $x = -C_B^2$ (for $C_B^2 \leq 1$) [32].

The above drawback can be removed by introducing a different boundary condition for the diffusion equation — a *barrier with jumps* [36].

When the diffusion process reaches $x = 0$, it remains there for a random time having an exponential distribution with parameter λ_0. After that, the process jumps to a point x drawn from a probability distribution with density $h(x)$.

The time during which $X(t) = 0$ corresponds to the idle period, during which there are no customers in the station, while the jump back to the interval $x > 0$ corresponds to the start of a new busy period.

Since this event is associated with the arrival of a single customer, we assume that jumps from $x = 0$ always lead to the point $x = 1$:

$$h(x) = \delta(x - 1) \, .$$

We further assume, approximately, that

$$\lambda_0 = \lambda \, .$$

With this boundary condition, the diffusion equation takes the form [36]

$$\begin{aligned} \frac{\partial f(x,t;x_0)}{\partial t} &= \frac{\alpha}{2}\frac{\partial^2 f(x,t;x_0)}{\partial x^2} - \beta\frac{\partial f(x,t;x_0)}{\partial x} + \lambda p_0(t)\delta(x-1) \, , \\ \frac{dp_0(t)}{dt} &= \lim_{x\to 0}\left[\frac{\alpha}{2}\frac{\partial f(x,t;x_0)}{\partial x} - \beta f(x,t;x_0)\right] - \lambda p_0(t) \, , \end{aligned} \tag{7.7}$$

where

$$p_0(t) = P[X(t) = 0] \, .$$

The term

$$\lambda p_0(t)\delta(x-1)$$

in the first equation describes the probability-density flow of the process to the point $x = 1$ at time t as a result of a jump from $x = 0$.

The second equation describes the evolution over time of the probability that the process remains at the barrier. The term

$$\lim_{x\to 0}\left[\frac{\alpha}{2}\frac{\partial f(x,t;x_0)}{\partial x} - \beta f(x,t;x_0)\right]$$

gives the probability density that the diffusion process reaches the barrier at time t, whereas the term

$$\lambda p_0(t)$$

represents the probability density that the process leaves the barrier at time t.

7.1.1 Steady state of the $G/G/1$ system

In the steady state, when

$$\lim_{t\to\infty} p_0(t) = p_0, \qquad \lim_{t\to\infty} f(x,t;x_0) = f(x)\,,$$

equation (7.7) takes the form

$$\begin{aligned} 0 &= \frac{\alpha}{2}\frac{d^2 f(x)}{dx^2} - \beta\frac{df(x)}{dx} + \lambda_0 p_0 \delta(x-1)\,, \\ 0 &= \lim_{x\to 0}\left[\frac{\alpha}{2}\frac{df(x)}{dx} - \beta f(x)\right] - \lambda p_0\,, \end{aligned} \tag{7.8}$$

and its solution is the function

$$f(x) = \begin{cases} \dfrac{\lambda p_0}{-\beta}\left(1 - e^{zx}\right), & \text{for} \quad 0 < x \le 1, \\ \dfrac{\lambda p_0}{-\beta}\left(e^{-z} - 1\right) e^{zx}, & \text{for} \quad x \ge 1, \end{cases} \qquad z = \frac{2\beta}{\alpha}\,. \tag{7.9}$$

After applying the normalisation condition, we obtain

$$p_0 = 1 - \varrho,$$

which is the exact result.

The values $f(x)$, for $x = 1, 2, \ldots$, given by the solution (7.9), are approximate values of $p(n)$, $n = 1, 2, \ldots$, for the $G/G/1$ system.

The example data in Table 7.1 illustrate the accuracy of this approximation: it is better when the distributions $A(x)$ and $B(x)$ do not deviate too far from the reference distributions, and it deteriorates when the coefficients of variation of these distributions differ substantially from one.

The mean number of tasks present in the system can be approximated by

$$\begin{aligned} E[N] &\approx \int_0^\infty x f(x)\,dx \\ &= \frac{\lambda p_0}{-\beta}\left(\int_0^1 x\left(1 - e^{zx}\right)\,dx + \int_1^\infty x\left(e^{-z} - 1\right) e^{zx}\,dx\right) \\ &= \frac{\lambda p_0}{-\beta}\left(\frac{1}{2} - \frac{1}{z}\right) = \left[\frac{1}{2} + \frac{C_A^2\varrho + C_B^2}{2(1-\varrho)}\right]\varrho\,. \end{aligned} \tag{7.10}$$

This is not the only possible approximation. One may also compute the mean as

$$E[N] = \sum_{n=0}^{\infty} n\, p(n),$$

where $p(n) = f(n)$, $n = 1, 2, \ldots$, or by using

$$p(1) = \int_0^{1.5} f(x)\,dx, \qquad p(n) = \int_{n-0.5}^{n+0.5} f(x)\,dx, \qquad n = 2, 3, \ldots,$$

	System $M/M/1$		System $M/G/1$	
n	$p(n)$	$f(n)$	$p(n)$	$f(n)$
0	0.2500	0.2500	0.2500	0.2500
1	0.1875	0.1875	0.1283	0.0747
2	0.1406	0.1406	0.0783	0.0675
3	0.1056	0.1056	0.0544	0.0607
4	0.0791	0.0793	0.0457	0.0547
5	0.0593	0.0596	0.0395	0.0492
6	0.0445	0.0448	0.0352	0.0443
7	0.0334	0.0337	0.0318	0.0399
8	0.0250	0.0253	0.0289	0.0359
9	0.0188	0.0190	0.0264	0.0323
10	0.0141	0.0143	0.0241	0.0291
	$E[N] = 3.00$	$E[N] \approx 3.02$	$E[N] = 7.50$	$E[N] \approx 7.51$

Table 7.1: Exact values $p(n)$ and their approximation $f(n)$, as well as exact and approximate mean queue lengths for the systems $M/M/1$ and $M/G/1$ ($C_B^2 = 5$); $\lambda = 1$, $\mu = 4/3$

or

$$p(n) = \int_{n-1}^{n} f(x)\,dx, \qquad n = 1, 2, 3, \ldots\ .$$

In the latter case,

$$E[N] = \left[1 + \frac{C_A^2 \varrho + C_B^2}{2(1-\varrho)}\right] \varrho\,. \tag{7.11}$$

The approximation (7.11) gives better results than (7.10) for small values of C_A^2 and C_B^2 combined with a low load ϱ.

Let us generalise the above results to the case of **multiple classes of customers.**

Assume that the station serves K classes of customers. Each class is described by its own arrival stream, in which interarrival times follow the distribution $A^{(k)}(x)$, and by its own service-time distribution $B^{(k)}(x)$.

The arrival stream of class k, $k = 1, \ldots, K$, is described by the distribution $A^{(k)}(x)$, with mean $1/\lambda^{(k)}$ and variance $(\sigma_A^{(k)})^2$, while the service time of customers of this class follows the distribution $B^{(k)}(x)$ with mean $1/\mu^{(k)}$ and variance $(\sigma_B^{(k)})^2$.

Let the corresponding densities be denoted by $a^{(k)}(x)$ and $b^{(k)}(x)$, respectively. The queueing discipline is natural; class membership does not affect the order of service.

We assume that the arrival streams of the individual classes are independent. The number of class-k customers arriving in a time interval of length t is then approximately normally distributed with mean

$$\lambda^{(k)} t$$

and variance

$$(\lambda^{(k)})^3 (\sigma_A^{(k)})^2 t = \lambda^{(k)} (C_A^{(k)})^2 t.$$

Hence, the number of arrivals of all classes combined during this interval is also approxi-

mately normally distributed, with mean

$$\lambda t = \sum_{k=1}^{K} \lambda^{(k)} t$$

and variance

$$\lambda C_A^2 t = \sum_{k=1}^{K} \lambda^{(k)} (C_A^{(k)})^2 t.$$

Thus, the parameters of the aggregate arrival process are

$$\lambda = \sum_{k=1}^{K} \lambda^{(k)}, \qquad C_A^2 = \sum_{k=1}^{K} \frac{\lambda^{(k)}}{\lambda} (C_A^{(k)})^2. \tag{7.12}$$

For the combined stream consisting of all customer classes, the service-time density without distinguishing customer class has the form

$$b(x) = \sum_{k=1}^{K} \frac{\lambda^{(k)}}{\lambda} b^{(k)}(x)\,,$$

where $\lambda^{(k)}/\lambda$ is the probability that a customer belongs to class k. This allows us to determine the aggregate service parameters [40]:

$$\frac{1}{\mu} = \sum_{k=1}^{K} \frac{\lambda^{(k)}}{\lambda} \frac{1}{\mu^{(k)}}, \qquad C_B^2 = \mu^2 \sum_{k=1}^{K} \left[\frac{\lambda^{(k)}}{\lambda} \frac{1}{(\mu^{(k)})^2} \big((C_B^{(k)})^2 + 1 \big) \right] - 1\,. \tag{7.13}$$

From these we determine the coefficients α and β of the diffusion equation.

The solution — again given by the function (7.9) — determines the distribution $p(n) \approx f(n)$ of the total number of customers of all classes present at the station.

The probability that there are ν customers of class k in the queue is then

$$p^{(k)}(\nu) = \sum_{n=\nu}^{\infty} \left[p(n) \binom{n}{\nu} \left(\frac{\lambda^{(k)}}{\lambda} \right)^{\nu} \left(1 - \frac{\lambda^{(k)}}{\lambda} \right)^{n-\nu} \right], \qquad k = 1, \ldots, K\,. \tag{7.14}$$

7.1.2 Transient state

The solution of equation (7.7), in the case where the density function of the diffusion process depends on time, may be obtained in the following way [22, 23].

First, we solve the diffusion equation with a boundary condition in the form of an absorbing barrier at the point $x = 0$: the process terminates when the barrier is reached. To distinguish this process from the one with returns, let us denote its density function by $\phi(x, t; x_0)$. The above boundary condition has the form

$$\lim_{x \to 0} \phi(x, t; x_0) = 0.$$

We are interested in the process starting at $x_0 = 1$. In this case, the process $X(t)$ describes changes in the number of customers during a single busy period of the station — the busy

period begins with the arrival of a single customer to an empty station and ends when the station becomes empty again.

The solution, obtained for example by the reflection principle, cf. [18], has the form

$$\phi(x,t;x_0) = \frac{e^{\frac{\beta}{\alpha}(x-x_0)-\frac{\beta^2}{2\alpha}t}}{\sqrt{2\pi\alpha t}}\left[e^{-\frac{(x-x_0)^2}{2\alpha t}} - e^{-\frac{(x+x_0)^2}{2\alpha t}}\right] . \tag{7.15}$$

We now calculate the probability density that the process, initiated at time $t = 0$ at the point $x = x_0$, reaches the barrier at time t, i.e. the density of the first-passage-time distribution from $x = x_0$ to $x = 0$:

$$\begin{aligned} \gamma_{x_0,0}(t) &= \lim_{x\to 0}\left[\frac{\alpha}{2}\frac{\partial}{\partial x}\phi(x,t;x_0) - \beta\phi(x,t;x_0)\right] \\ &= \frac{x_0}{\sqrt{2\pi\alpha t^3}} e^{-\frac{(\beta t+x_0)^2}{2\alpha t}} . \end{aligned} \tag{7.16}$$

The function $\gamma_{1,0}(t)$ is an approximation of the distribution of the busy-period duration in the $G/G/1$ system.

The operation of the station consists of alternating busy and idle periods; similarly, the diffusion process with returns consists of alternating intervals of diffusion starting at $x = 1$ and ending at $x = 0$, and stays at $x = 0$. The density $f(x,t;x_0)$ of the diffusion process with returns can be expressed as a superposition of the densities $\phi(x,t-\tau;1)$ of one-time processes with an absorbing barrier, started earlier, at times $\tau < t$ with an as yet unknown intensity $g_1(\tau)$:

$$f(x,t;x_0) = \phi(x,t;x_0) + \int_0^t g_1(\tau)\phi(x,t-\tau;1)\,d\tau . \tag{7.17}$$

The first term on the right-hand side, the function $\phi(x,t;x_0)$, represents the fulfilment of the initial condition — the first process, initiated at x_0, which has not yet ended by time t.

If the process does not start from a single point x_0, but instead the initial condition is given by a density $\psi(x)$, then

$$\phi(x,t;\psi) = \int_0^\infty \phi(x,t;\xi)\psi(\xi)\,d\xi, \qquad \gamma_{\psi,0}(t) = \int_0^\infty \gamma_{\xi,0}(t)\psi(\xi)\,d\xi.$$

The probability density of the process reaching the barrier at time t is

$$\gamma_0(t) = p_0(0)\delta(t) + \big(1-p_0(0)\big)\gamma_{x_0,0}(t) + \int_0^t g_1(\tau)\gamma_{1,0}(t-\tau)\,d\tau . \tag{7.18}$$

The probability mass flow entering the barrier reappears, after an idle period described by the density $l_0(x)$, at the point $x = 1$ as the intensity $g_1(t)$:

$$g_1(\tau) = \int_0^\tau \gamma_0(t)l_0(\tau-t)\,dt . \tag{7.19}$$

Equations (7.18) and (7.19) make it possible to determine the intensity $g_1(\tau)$. Computationally, this is more convenient if we apply the Laplace transform:

$$\begin{aligned} \bar\gamma_0(s) &= p_0(0) + \big(1-p_0(0)\big)\bar\gamma_{x_0,0}(s) + \bar g_1(s)\bar\gamma_{1,0}(s) , \\ \bar g_1(s) &= \bar\gamma_0(s)\bar l_0(s) , \end{aligned} \tag{7.20}$$

where

$$\bar{\gamma}_{x_0,0}(s) = e^{-x_0 \frac{\beta+\sqrt{\beta^2+2\alpha s}}{\alpha}} ,$$

from which we obtain

$$\bar{g}_1(s) = \left[p_0(0) + \left(1 - p_0(0)\right)\bar{\gamma}_{x_0,0}(s)\right] \frac{\bar{l}_0(s)}{1 - \bar{l}_0(s)\bar{\gamma}_{1,0}(s)} . \tag{7.21}$$

Note that the distribution of the time spent at the barrier, described here by the density $l_0(x)$, is arbitrary. The proposed solution therefore describes a more general case than follows from equations (7.7), which assume an exponential distribution of the time spent at the barrier.

Equation (7.17), after transformation, takes the form

$$\bar{f}(x, s; x_0) = \bar{\phi}(x, s; x_0) + \bar{g}_1(s)\bar{\phi}(x, s; 1) ,$$

where

$$\bar{\phi}(x, s; 1) = \frac{e^{\frac{\beta(x-1)}{\alpha}}}{A(s)} \left[e^{-|x-1|\frac{A(s)}{\alpha}} - e^{-|x+1|\frac{A(s)}{\alpha}}\right] , \qquad A(s) = \sqrt{\beta^2 + 2\alpha s} .$$

The inverse transform of $\bar{f}(x, s; x_0)$, i.e. the solution to the problem of determining the density of a diffusion process with returns, is found numerically in the examples given below, as well as in the next chapter, using the Stehfest algorithm [113]; see Appendix B. The accuracy of the approximation may be improved by using more advanced methods for the numerical inversion of Laplace transforms, discussed for example in [2, 16].

Figure **??** shows example results for the queue-length distribution in the $M/M/1$ system. The system parameters are $\lambda = 0.75$, $\mu = 1$; the results shown were calculated for $t = 75$, with the queue initially empty. These results are compared with those of the exact analytical model, in which the distribution values are expressed through Bessel functions. Simulation results are also shown in the same figure.

Figs. **??** and **??** show changes in the average queue length for step changes in the intensity of the input stream to an $M/D/1$ station, i.e. a system with constant service time (equal here to one unit of time).

In Fig. **??**, at the initial moment the system is in steady state, and the arrival rate changes from $\lambda = 0.5$ to $\lambda = 0.8$. At time $t = 200$, it returns to $\lambda = 0.5$. The average queue length was calculated as

$$E[N(t)] = \sum_{n=0}^{100} p(n, t)\, n,$$

where either $p(n, t) = f(n, t)$ (results labelled "diffusion1") or

$$p(n, t) = \int_{n-1}^{n} f(x, t)\, dx$$

(results labelled "diffusion2").

Figure **??** shows analogous results for more complex changes in the input flow, including a period during which the queue becomes unstable.

The simulation results were obtained by averaging 60,000 independent realisations of the process $N(t)$ for these cases.

Figs. 7.4 and 7.5 illustrate the dynamics of changes in the average queue length.

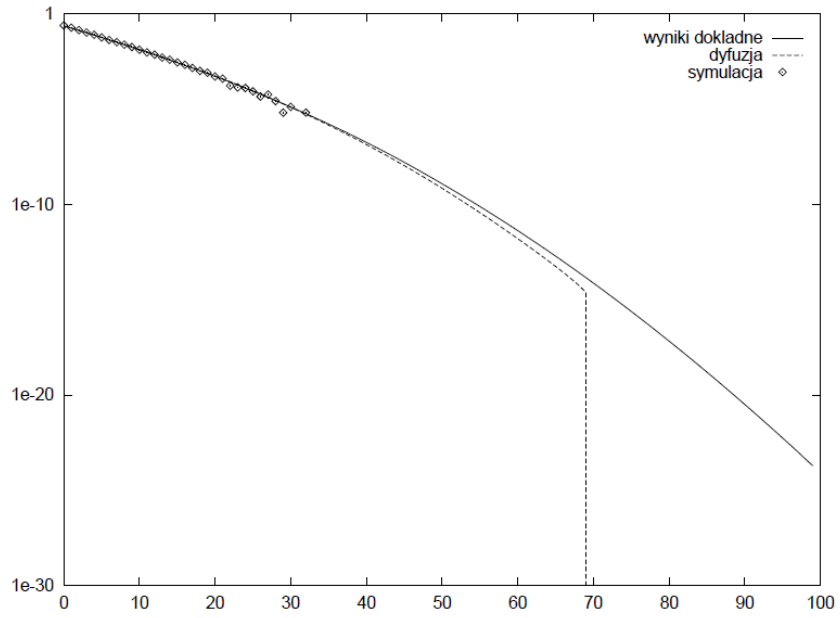


Figure 7.1: The $M/M/1$ system ($\lambda = 0.75$, $\mu = 1$): queue-length distribution, with the queue empty at the initial time, for $t = 75$; comparison of results from the diffusion model, analytical calculations, and simulation (150,000 runs)

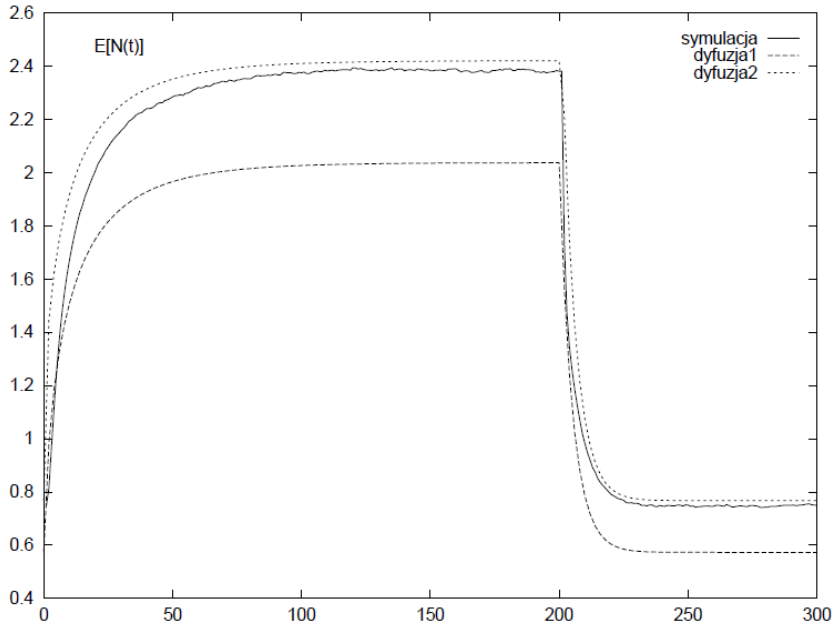


Figure 7.2: Average queue length as a function of changes in the intensity of the Poisson arrival stream λ — comparison of diffusion-approximation results and simulation; the service time is constant and equal to one unit of time

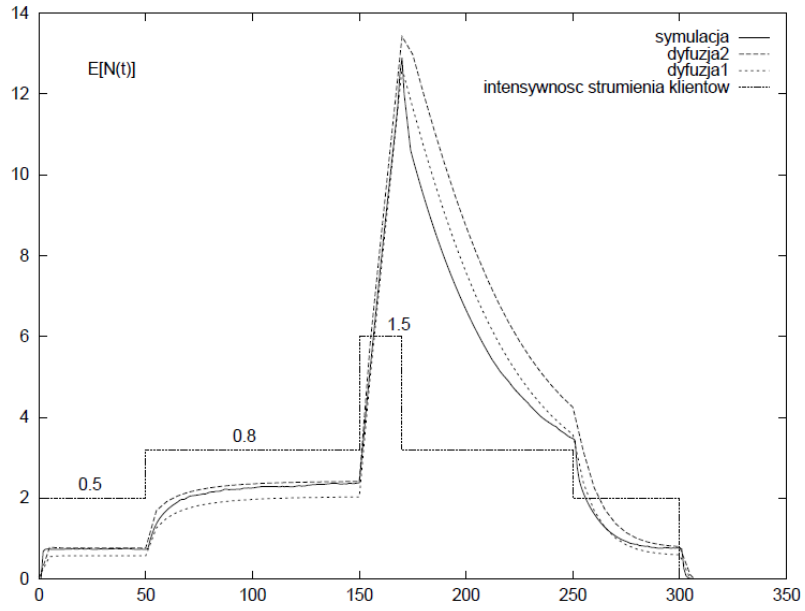


Figure 7.3: Average queue length as a function of changes in the intensity of the Poisson arrival stream λ — comparison of diffusion-approximation results and simulation; the service time is constant and equal to one unit of time

In Fig. 7.4, at time $t = 0$, customers begin to arrive at a constant rate λ to an initially empty queue. The average queue length $E[N(t)]$ increases and eventually reaches the steady-state value $E[N(\infty)]$.

Figure **??** illustrates the opposite case: at time $t = 0$ the $G/G/1$ queue is in steady state and has the value $E[N(0)]$. For $t > 0$, no new customers enter the queue, and its average length decreases.

The shapes of the curves are very similar to the functions

$$1 - e^{-t/T} \qquad \text{and} \qquad e^{-t/T}.$$

The time constant T determines the duration of the transient period.

It can be seen that the rate of change depends on the value $C^2 = C_A^2 = C_B^2$. Figure **??** shows that this is approximately a linear relationship.

Similarly, the utilisation factor ϱ of the system strongly influences the duration of the transient period, and this dependence is nonlinear — see Fig. **??**.

The values of the time constant for a decreasing queue are different, but the dependence on ϱ and C^2 is of the same nature.

7.2 Approximation of an open $G/G/1$ network

Let the network contain M nodes. Let us initially assume a single class of customers. The state of the network can be approximated by an M-dimensional diffusion process,

$$\boldsymbol{X}(t) = [X_1(t), X_2(t), \ldots, X_M(t)]\,.$$

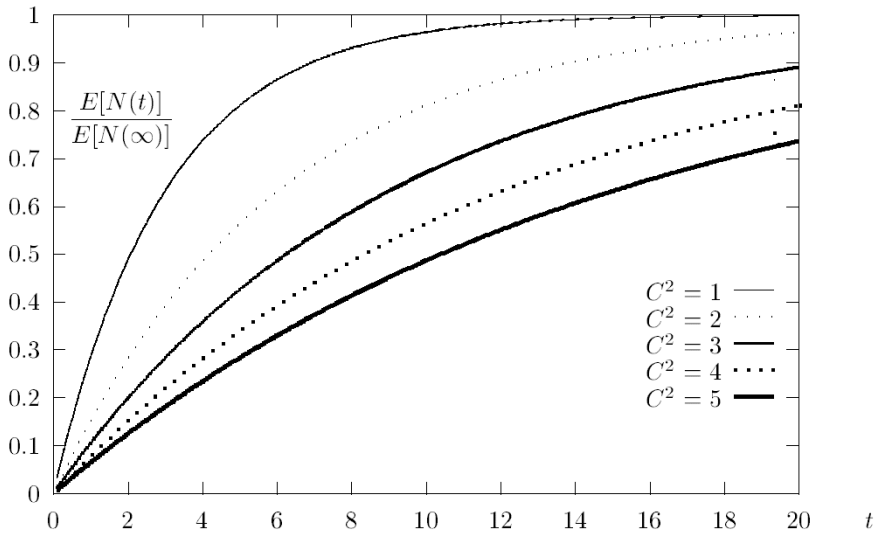


Figure 7.4: Average queue length during its rise from zero to the steady-state value

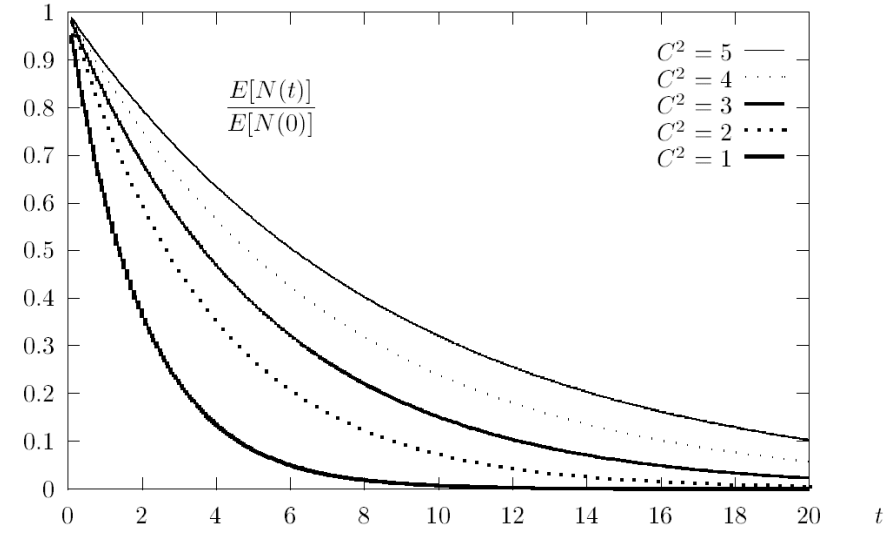


Figure 7.5: Average queue length during its decline from the steady-state value to zero

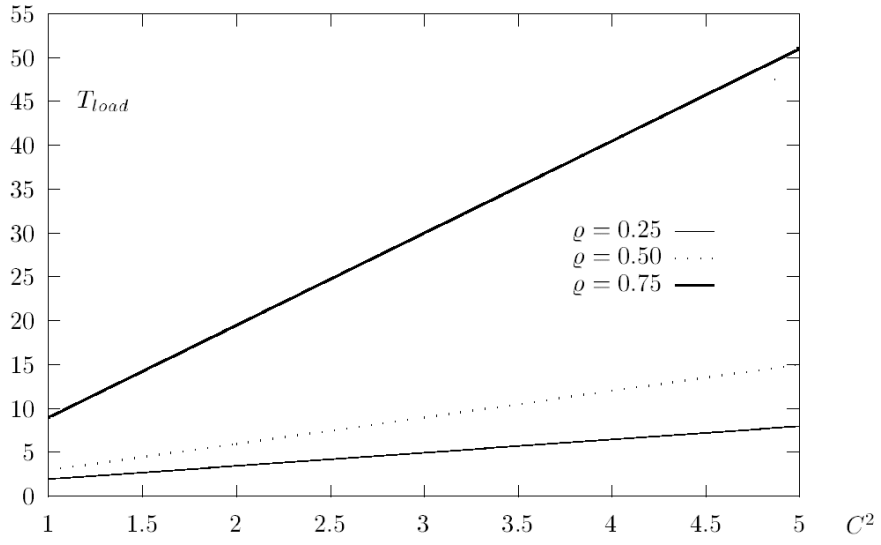


Figure 7.6: Time constant T_{load} during queue build-up as a function of $C^2 = C_A^2 = C_B^2$

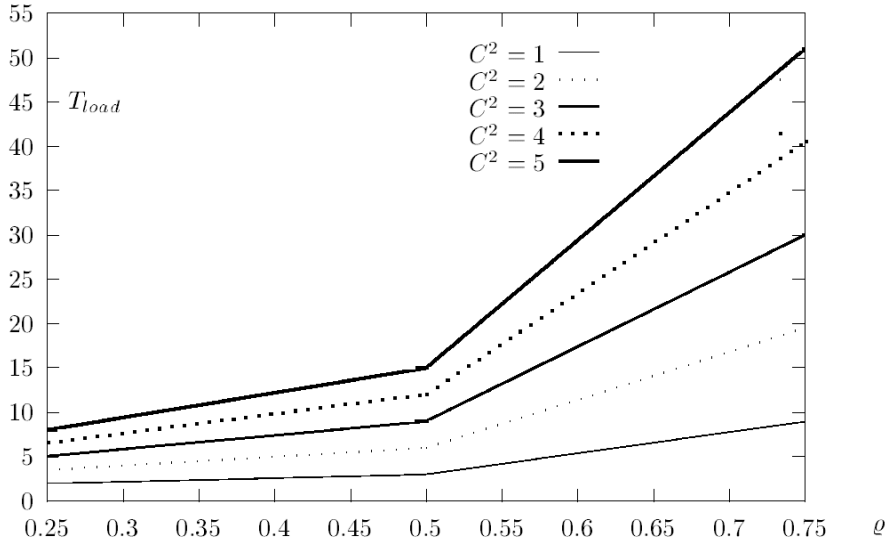


Figure 7.7: Time constant T_{load} during queue build-up as a function of ϱ

The component $X_i(t)$ corresponds to the number of customers at station i.

The probability density function of this process, $f(\boldsymbol{x}, t; \boldsymbol{x}_0)$, satisfies the equation

$$\frac{\partial f(\boldsymbol{x}, t; \boldsymbol{x}_0)}{\partial t} = \frac{1}{2} \sum_{i=1}^{M} \sum_{j=1}^{M} \alpha_{ij} \frac{\partial^2 f(\boldsymbol{x}, t; \boldsymbol{x}_0)}{\partial x_i \partial x_j} - \sum_{i=1}^{M} \beta_i \frac{\partial f(\boldsymbol{x}, t; \boldsymbol{x}_0)}{\partial x_i} .$$

The coefficients α_{ij} correspond to the covariance of the process changes $N_i(t)$ and $N_j(t)$, while the coefficients β_i correspond to the mean changes of $N_i(t)$ per unit time.

Such a network model was proposed in [64]. Assuming that none of the nodes is empty, the values of α_{ij} and β_i were derived there, together with a solution for the steady state under boundary conditions restricting the process to positive values of $x_1, \ldots, x_M$. The solution takes the form

$$f(\boldsymbol{x}) = \prod_{i=1}^{M} |z_i| \, e^{z_i x_i}, \tag{7.22}$$

provided that all elements of the vector

$$\boldsymbol{z} = 2\boldsymbol{\alpha}^{-1}\boldsymbol{\beta}$$

are negative.

The simplest way to apply the diffusion approximation to a network of stations is through decomposition of the model [99]: the arrival rates to individual stations are first calculated, after which each station is treated independently and described using the single-station model. Below we outline this approach.

The value of the parameter λ_i of the arrival stream to station i in the network is determined by the traffic equations

$$\lambda_i = \lambda_{0i} + \sum_{j=1}^{M} \lambda_j r_{ji} \, , \tag{7.23}$$

where, as before, r_{ji} is the routing probability from node j to node i, and λ_{0i} is the intensity of the external flow arriving at node i.

The second moment of the distribution of interarrival times at a given station is calculated using two relationships. The first relates the departure process of a given station to its arrival process, while the second relates the arrival process of a station to the departure processes of all stations in the network [40].

1. The density $d_i(x)$ of the distribution $D_i(x)$ of the interdeparture time between successive customers leaving station i is given by the formula (exact for stations with Poisson arrivals, approximate for $G/G/1$ systems) [10]

   $$d_i(x) = \varrho_i b_i(x) + (1 - \varrho_i)\, a_i(x) * b_i(x), \qquad i = 1, \ldots, M, \tag{7.24}$$

   where $*$ denotes convolution.

   This formula may be interpreted as follows: if the station is busy, customers depart at intervals determined by the service time distribution, whereas after the departure of the last customer during a busy period, one must first wait for a new arrival and then for the completion of its service before the next departure occurs.

The waiting time for a new arrival has the distribution $A(x)$ in the case of a Poisson arrival process; in other cases it is unknown and is approximated by $A(x)$.

From (7.49) we calculate the variance, or coefficient of variation, of this distribution. The mean value is, of course,

$$E[D_i] = \frac{1}{\lambda_i}$$

for station i in equilibrium. The coefficient of variation is

$$C_{Di}^2 = C_{Ai}^2(1-\varrho_i) + \varrho_i^2 C_{Bi}^2 + \varrho_i(1-\varrho_i)\,. \tag{7.25}$$

2. It is assumed that the variance of the total number of customers arriving at station j per unit time from other stations and from outside the network is equal to the sum of the variances of the individual components of this flow.

   Thus, we assume that the flows reaching station j from different stations are mutually independent.

Customers depart from station i according to the distribution $D_i(x)$ and choose the route to station j with probability r_{ij}. Therefore, the time intervals between successive customers travelling along the route $i \to j$ follow a distribution with density $d_{ij}(x)$:

$$d_{ij}(x) = d_i(x)r_{ij} + d_i(x) * d_i(x)(1-r_{ij})r_{ij} + d_i(x) * d_i(x) * d_i(x)(1-r_{ij})^2 r_{ij} + \cdots$$

or, after applying the Laplace transform,

$$\bar{d}_{ij}(s) = \bar{d}_i(s)r_{ij} + \bar{d}_i(s)^2(1-r_{ij})r_{ij} + \bar{d}_i(s)^3(1-r_{ij})^2 r_{ij} + \cdots = \frac{r_{ij}\bar{d}_i(s)}{1-(1-r_{ij})\bar{d}_i(s)}\,.$$

From this we calculate the mean and the coefficient of variation of this distribution:

$$\begin{aligned} E[D_{ij}] &= \frac{1}{\lambda_i r_{ij}}\,, \\ C_{D_{ij}}^2 &= r_{ij}(C_{D_i}^2 - 1) + 1\,. \end{aligned} \tag{7.26}$$

The above quantities refer to the time intervals between customers travelling from station i to station j, whereas the number of customers travelling along this route during a time interval of length t is approximately normally distributed with mean

$$\lambda_i r_{ij} t$$

and variance

$$\big[r_{ij}(C_{D_i}^2 - 1) + 1\big]\lambda_i r_{ij} t\,.$$

The flows from individual stations in the network merge at the input to station j. Assuming independence of these flows (which is an approximation), we conclude that the number of customers arriving at station j during a time interval t is approximately normally distributed with mean

$$\lambda_j t = \left[\sum_{i=1}^{M} \lambda_i r_{ij} + \lambda_{0j}\right] t$$

and variance

$$\sigma_{A_j}^2 t = \left\{ \sum_{i=1}^{M} \left[r_{ij}(C_{D_i}^2 - 1) + 1 \right] \lambda_i r_{ij} + C_{0j}^2 \lambda_{0j} \right\} t \,.$$

Here, the parameters λ_{0j} and C_{0j}^2 refer to the external flow.

The variance $\sigma_{A_j}^2$ determines the coefficient of variation $C_{A_j}^2$ of the distribution of interarrival times to station j:

$$C_{A_j}^2 = \frac{1}{\lambda_j} \sum_{i=1}^{M} r_{ij} \lambda_i \left[(C_{D_i}^2 - 1) r_{ij} + 1 \right] + \frac{C_{0j}^2 \lambda_{0j}}{\lambda_j} \,. \tag{7.27}$$

For K customer classes, the traffic equations (7.23) take the form

$$\lambda_i^{(k)} = \lambda_{0i}^{(k)} + \sum_{j=1}^{M} \lambda_j^{(k)} r_{ji}^{(k)}, \qquad i = 1, \ldots, M, \;\; k = 1, \ldots, K, \tag{7.28}$$

where $r_{ji}^{(k)}$ denotes the probability of transition from station j to station i for a customer of class k.

Equation (7.50) then takes the form

$$C_{D_i}^2 = \lambda_i \sum_{k=1}^{K} \frac{\lambda_i^{(k)}}{(\mu_i^{(k)})^2} \left[(C_{B_i}^{(k)})^2 + 1 \right] + 2\varrho_i(1 - \varrho_i) + (C_{A_i}^2 + 1)(1 - \varrho_i) - 1 \,. \tag{7.29}$$

In the stream of customers leaving station i, a customer of class k appears with probability

$$\frac{\lambda_i^{(k)}}{\lambda_i}$$

and proceeds to station j with probability

$$r_{ij}^{(k)}.$$

The coefficient of variation $(C_{D_{ij}}^{(k)})^2$ for the distribution of time intervals between class-k customers moving from station i to station j is calculated analogously to $C_{D_{ij}}^2$ in equation (7.52), replacing r_{ij} with

$$r_{ij}^{(k)} \frac{\lambda_i^{(k)}}{\lambda_i} \,.$$

Thus equation (7.52) takes the following form for K customer classes:

$$C_{A_j}^2 = \frac{1}{\lambda_j} \sum_{i=1}^{M} \sum_{k=1}^{K} r_{ij}^{(k)} \lambda_i^{(k)} \left[(C_{D_i}^2 - 1) r_{ij}^{(k)} \frac{\lambda_i^{(k)}}{\lambda_i} + 1 \right] + \sum_{k=1}^{K} \frac{(C_{0j}^{(k)})^2 \lambda_{0j}^{(k)}}{\lambda_j} \,. \tag{7.30}$$

Equations (7.50) and (7.52), or in the multiclass case (7.29) and (7.30), form a system that allows the calculation of $C_{A_i}^2$. Together with the traffic intensities λ_i obtained from the traffic equations, these values allow the determination of β_i and α_i for each station.

In the case of multiple customer classes, the distribution of the number of customers in a selected class is calculated according to equation (7.14).

Discussions of the approximation errors, which increase with $C_{A_i}^2$ and $C_{B_i}^2$, can be found in [99].

At this point, an illustrative example should be added, for instance: a network with two stations in series, or a three-station network in which customers move from station (1) to either station (2) or station (3), considering:

(a) one class of customers,

(b) two classes of customers.

7.3 Approximation of the $G/G/1/N$ system

In the system denoted, according to Kendall's notation, by $G/G/1/N$, the number of tasks present at a station is limited to N. When the queue is full, subsequent incoming requests are lost.

The Markov model of this system was discussed in Section 3.2.3. In the diffusion model of the $G/G/1/N$ system, the diffusion process is bounded by two barriers: the first is located at $x = 0$, as in the $G/G/1$ model, and the second is located at $x = N$. When the process reaches the right-hand barrier, it remains there for a time determined by an exponential distribution with parameter λ_N, after which it restarts at the point $x = N - 1$. The time spent at this barrier corresponds to the period during which the queue is full and incoming customers are rejected; the jump to $x = N - 1$ corresponds to the departure of the currently served customer, so the station can once again accept requests. We choose the value of λ_N as

$$\lambda_N = \mu \, .$$

The diffusion equations with this additional boundary condition take the form

$$\begin{aligned} \frac{\partial f(x,t;x_0)}{\partial t} &= \frac{\alpha}{2}\frac{\partial^2 f(x,t;x_0)}{\partial x^2} - \beta\frac{\partial f(x,t;x_0)}{\partial x} \\ &\qquad +\lambda_0 p_0(t)\delta(x-1) + \lambda_N p_N(t)\delta(x-N+1) \, , \\ \frac{dp_0(t)}{dt} &= \lim_{x\to 0}\left[\frac{\alpha}{2}\frac{\partial f(x,t;x_0)}{\partial x} - \beta f(x,t;x_0)\right] - \lambda_0 p_0(t) \, , \\ \frac{dp_N(t)}{dt} &= -\lim_{x\to N}\left[\frac{\alpha}{2}\frac{\partial f(x,t;x_0)}{\partial x} - \beta f(x,t;x_0)\right] - \lambda_N p_N(t) \, , \end{aligned} \tag{7.31}$$

where $p_N(t) = P[X(t) = N]$.

The minus sign before the first term on the right-hand side of the equation for $\frac{dp_N(t)}{dt}$ appears because the increase in the probability $p_N(t)$ is caused by the process moving in the opposite direction to that in the balance equation for $\frac{dp_0(t)}{dt}$.

In the steady state, equations (7.31) become ordinary differential equations, and their solution, for $\varrho = \lambda/\mu \neq 1$, is [36]:

$$f(x) = \begin{cases} \dfrac{\lambda p_0}{-\beta}(1 - e^{zx}) & \text{for} \quad 0 < x \leq 1 \, , \\ \dfrac{\lambda p_0}{-\beta}(e^{-z} - 1)e^{zx} & \text{for} \quad 1 \leq x \leq N-1 \, , \\ \dfrac{\mu p_N}{-\beta}(e^{z(x-N)} - 1) & \text{for} \quad N-1 \leq x < N \, , \end{cases} \tag{7.32}$$

where

$$z = \frac{2\beta}{\alpha}\,,$$

and p_0, p_N are determined from the normalisation condition.

Note that the steady-state solution does not depend on the full distribution of the residence times within the barriers, but only on the mean values of these distributions.

As before, by using equations (7.12)–(7.14), we can also describe the case in which the input stream consists of K streams corresponding to different customer classes.

In the transient state, we proceed similarly to the case of the $G/G/1$ system: we first consider the diffusion process starting at time $t = 0$ at point $x = x_0$ and bounded by two absorbing barriers at $x = 0$ and $x = N$. The density function of this process, $\phi(x, t; x_0)$, takes the form [18]

$$\phi(x,t;x_0) = \begin{cases} \delta(x - x_0) & \text{for} \quad t = 0\,, \\ \dfrac{1}{\sqrt{2\pi\alpha t}} \displaystyle\sum_{n=-\infty}^{\infty} \left\{ \exp\left[\frac{\beta x'_n}{\alpha} - \frac{(x - x_0 - x'_n - \beta t)^2}{2\alpha t} \right] \right. & \\ \qquad \left. - \exp\left[\frac{\beta x''_n}{\alpha} - \frac{(x - x_0 - x''_n - \beta t)^2}{2\alpha t} \right] \right\} & \text{for} \quad t > 0\,, \end{cases} \tag{7.33}$$

where

$$x'_n = 2nN, \qquad x''_n = -2x_0 - x'_n\,.$$

If the initial condition is given by a function $\psi(x)$, $x \in (0, N)$, with

$$\lim_{x\to 0} \psi(x) = \lim_{x\to N} \psi(x) = 0\,,$$

then the density function takes the form

$$\phi(x,t;\psi) = \begin{cases} \psi(x) & \text{for} \quad t = 0\,, \\ \displaystyle\int_0^N \phi(x,t;\xi)\psi(\xi)\,d\xi & \text{for} \quad t > 0\,. \end{cases} \tag{7.34}$$

Let

$$A(s) = \sqrt{\beta^2 + 2\alpha s}\,.$$

The Laplace transform of the function $\phi(x, t; x_0)$ is then expressed as

$$\begin{aligned} \bar{\phi}(x,s;x_0) &= \frac{\exp\left[\frac{\beta(x-x_0)}{\alpha}\right]}{A(s)} \sum_{n=-\infty}^{\infty} \left\{ \exp\left[-\frac{|x - x_0 - x'_n|}{\alpha} A(s) \right] \right. \\ &\qquad \left. - \exp\left[-\frac{|x - x_0 - x''_n|}{\alpha} A(s) \right] \right\}\,, \end{aligned} \tag{7.35}$$

$$\bar{\phi}(x,s;\psi) = \int_0^N \bar{\phi}(x,s;\xi)\psi(\xi)\,d\xi\,. \tag{7.36}$$

The density function $f(x, t; \psi)$ of the diffusion process with elementary returns contains the function $\phi(x, t; \psi)$, which represents the influence of the initial condition, and compositions of the functions $\phi(x, t - \tau; 1)$ and $\phi(x, t - \tau; N - 1)$, which are diffusion process functions with

absorbing barriers at the points $x = 0$ and $x = N$, initiated earlier, at time τ, at the points $x = 1$ and $x = N - 1$, with intensities $g_1(\tau)$ and $g_{N-1}(\tau)$:

$$f(x,t;\psi) = \phi(x,t;\psi) + \int_0^t g_1(\tau)\phi(x,t-\tau;1)\,d\tau + \int_0^t g_{N-1}(\tau)\phi(x,t-\tau;N-1)\,d\tau\,. \quad (7.37)$$

The functions $\gamma_0(t)$ and $\gamma_N(t)$ represent the probabilities that at time t the process reaches the barriers at $x = 0$ and $x = N$, respectively:

$$\begin{aligned}
\gamma_0(t) &= p_0(0)\delta(t) + \Big[1 - p_0(0) - p_N(0)\Big]\gamma_{\psi,0}(t) \\
&\quad + \int_0^t g_1(\tau)\gamma_{1,0}(t-\tau)\,d\tau + \int_0^t g_{N-1}(\tau)\gamma_{N-1,0}(t-\tau)\,d\tau\,, \\
\gamma_N(t) &= p_N(0)\delta(t) + \Big[1 - p_0(0) - p_N(0)\Big]\gamma_{\psi,N}(t) \\
&\quad + \int_0^t g_1(\tau)\gamma_{1,N}(t-\tau)\,d\tau + \int_0^t g_{N-1}(\tau)\gamma_{N-1,N}(t-\tau)\,d\tau\,. \qquad (7.38)
\end{aligned}$$

Here, $\gamma_{1,0}(t)$, $\gamma_{1,N}(t)$, $\gamma_{N-1,0}(t)$, and $\gamma_{N-1,N}(t)$ are functions describing the first-passage time between the corresponding points, i.e.

$$\gamma_{1,0}(t) = \lim_{x\to 0}\left[\frac{\alpha}{2}\frac{\partial\phi(x,t;1)}{\partial x} - \beta\phi(x,t;1)\right]\,.$$

Since

$$\lim_{x\to 0}\phi(x,t;x_0) = \lim_{x\to N}\phi(x,t;x_0) = 0\,,$$

it follows that

$$\begin{aligned}
\gamma_{1,0}(t) &= \lim_{x\to 0}\frac{\alpha}{2}\frac{\partial\phi(x,t;1)}{\partial x}\,, \\
\gamma_{N-1,0}(t) &= \lim_{x\to 0}\frac{\alpha}{2}\frac{\partial\phi(x,t;N-1)}{\partial x}\,, \\
\gamma_{1,N}(t) &= -\lim_{x\to N}\frac{\alpha}{2}\frac{\partial\phi(x,t;1)}{\partial x}\,, \\
\gamma_{N-1,N}(t) &= -\lim_{x\to N}\frac{\alpha}{2}\frac{\partial\phi(x,t;N-1)}{\partial x}\,. \qquad (7.39)
\end{aligned}$$

The functions $\gamma_{\psi,0}(t)$ and $\gamma_{\psi,N}(t)$ denote the probabilities that processes initiated at $t = 0$ at point ξ with intensity $\psi(\xi)$ terminate at time t by reaching the barrier at $x = 0$ or $x = N$, respectively.

The intensities $g_1(t)$ and $g_{N-1}(t)$ can be expressed in terms of the functions $\gamma_0(t)$ and $\gamma_N(t)$:

$$\begin{aligned}
g_1(\tau) &= \int_0^\tau \gamma_0(t)\,l_0(\tau-t)\,dt\,, \\
g_{N-1}(\tau) &= \int_0^\tau \gamma_N(t)\,l_N(\tau-t)\,dt\,. \qquad (7.40)
\end{aligned}$$

The Laplace transforms of (7.38) and (7.40) give $\bar{g}_1(s)$ and $\bar{g}_{N-1}(s)$:

$$\begin{aligned}
\bar{g}_1(s) &= \Bigg\{ p_0(0) + \Big[1 - p_0(0) - p_N(0)\Big]\bar{\gamma}_{\psi,0}(s) \\
&\quad + \Big[p_N(0) + \Big(1 - p_0(0) - p_N(0)\Big)\bar{\gamma}_{\psi,N}(s)\Big] \frac{\bar{l}_N(s)\bar{\gamma}_{N-1,0}(s)}{1 - \bar{l}_N(s)\bar{\gamma}_{N-1,N}(s)} \Bigg\} \\
&\quad \times \frac{\bar{l}_0(s)}{1 - \bar{l}_0(s)\bar{\gamma}_{1,0}(s)} \left[1 - \frac{\bar{l}_0(s)\bar{\gamma}_{1,N}(s)}{1 - \bar{l}_0(s)\bar{\gamma}_{1,0}(s)} \frac{\bar{l}_N(s)\bar{\gamma}_{N-1,0}(s)}{1 - \bar{l}_N(s)\bar{\gamma}_{N-1,N}(s)}\right]^{-1}, \\
\bar{g}_{N-1}(s) &= \frac{\bar{l}_N(s)}{1 - \bar{l}_N(s)\bar{\gamma}_{N-1,N}(s)} \Bigg[p_N(0) + \Big(1 - p_0(0) - p_N(0)\Big)\bar{\gamma}_{\psi,N}(s) \\
&\quad + \bar{g}_1(s)\bar{\gamma}_{1,N}(s) \Bigg]. \qquad (7.41)
\end{aligned}$$

This allows us to determine the Laplace transform of the desired density function $f(x,t;\psi)$ for the diffusion process with returns:

$$\bar{f}(x,s;\psi) = \bar{\phi}(x,s;\psi) + \bar{g}_1(s)\bar{\phi}(x,s;1) + \bar{g}_{N-1}(s)\bar{\phi}(x,s;N-1)\,. \qquad (7.42)$$

The probabilities that at time t the process is at $x = 0$ or $x = N$ are

$$\begin{aligned}
\bar{p}_0(s) &= \frac{1}{s}\Big[\bar{\gamma}_0(s) - \bar{g}_1(s)\Big], \\
\bar{p}_N(s) &= \frac{1}{s}\Big[\bar{\gamma}_N(s) - \bar{g}_{N-1}(s)\Big]. \qquad (7.43)
\end{aligned}$$

As in the case of the $G/G/1$ system, the inverse Laplace transforms are obtained numerically.

Figures **??**, **??**, and **??** illustrate an example solution of the diffusion equation, showing density functions for various times under the initial condition $x_0 = 5$ and barriers restricting the process to the interval $x \in [0, 10]$.

Estimation of the residence time at the barrier (i.e. the period during which incoming cells are lost).

The transmission time of a cell is constant and equal to $1/D$. Assume that transmission starts at time t and that $X(t) = N - 1$. At some time $t + Z$, before $t + 1/D$, the next cell arrives, causing $X(t + Z) = N$. The time H spent at the upper barrier, assuming a Poisson arrival process with intensity λ, has the distribution

$$\Pr[H \le v] = \Pr\left[\frac{1}{D} - Z \le v \;\middle|\; Z \le \frac{1}{D}\right] = \frac{\Pr\left[\frac{1}{D} - v \le Z \le \frac{1}{D}\right]}{\Pr\left[Z \le \frac{1}{D}\right]} = \frac{e^{-\lambda/D}\left(e^{\lambda v} - 1\right)}{1 - e^{-\lambda/D}}. \qquad (7.44)$$

Hence,

$$f_H(v) = \frac{\lambda e^{\lambda v}}{e^{\lambda/D} - 1}, \qquad (7.45)$$

and the mean residence time is

$$E[H] = \int_0^{1/D} v f_H(v)\, dv = \frac{1/D}{1 - e^{-\lambda/D}} - \frac{1}{\lambda}. \qquad (7.46)$$

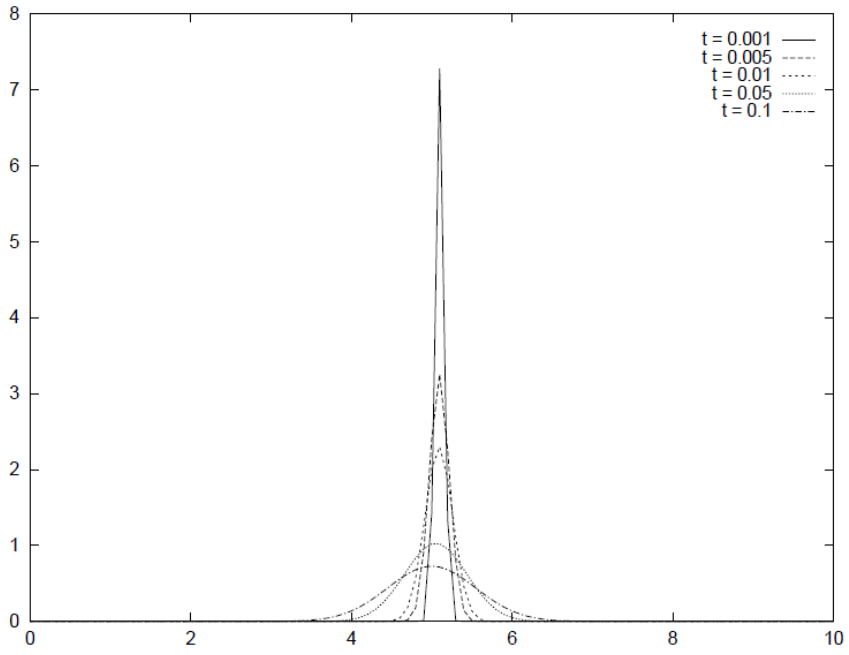


Figure 7.8: Solution of the diffusion equation as a function of time, $t = 0.001$, $t = 0.005$, $t = 0.01$, $t = 0.05$, $t = 0.1$; $N = 10$, $\lambda = 1$, $\mu = 2$

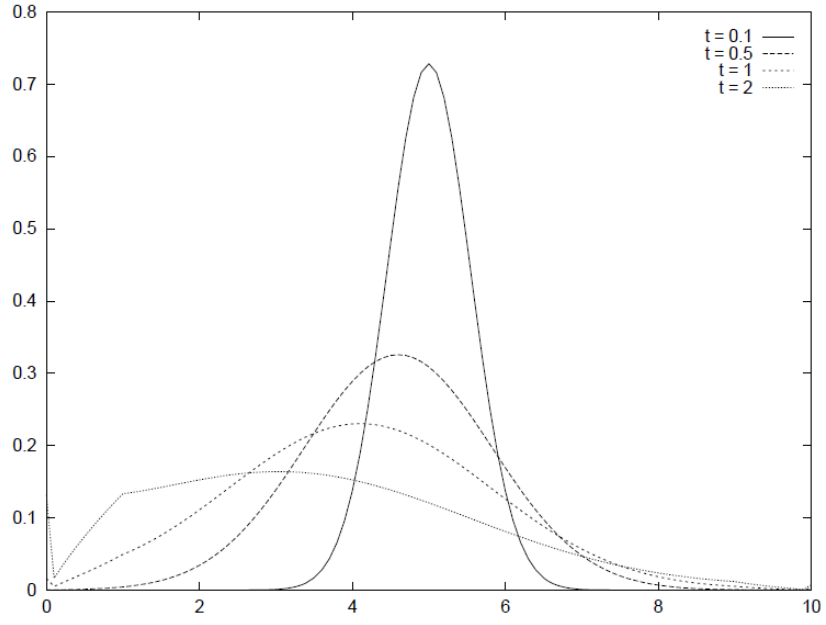


Figure 7.9: Solution of the diffusion equation as a function of time, $t = 0.1$, $t = 0.5$, $t = 1$, $t = 2$; $\lambda = 1$, $\mu = 2$

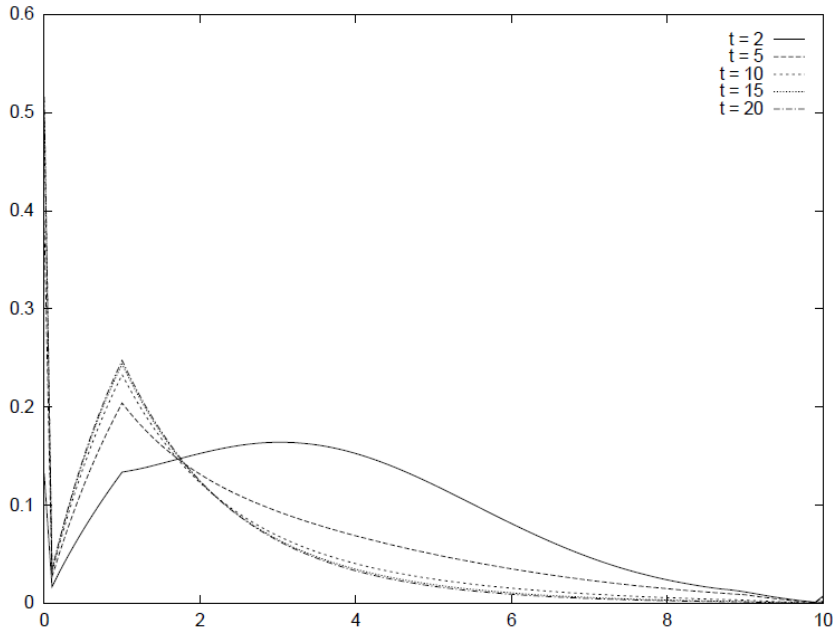


Figure 7.10: Solution of the time-dependent diffusion equation, $t = 2$, $t = 5$, $t = 10$, $t = 15$, $t = 20$; $\lambda = 1$, $\mu = 2$

The losses $L(t)$ caused by overflow of the output queue may be estimated as

$$L(t) = p_N(t) \Pr\left[N(t, t+H) \geq 1 \mid X(t) = N\right]. \tag{7.47}$$

Since it is difficult to calculate

$$\Pr\left[N(t, t+H) \geq 1 \mid X(t) = N\right],$$

we instead use

$$\Pr\left[N(t, t+1/D) \geq 1 \mid X(t) = N\right],$$

which is certainly not smaller than the original probability and therefore provides an upper bound on the loss probability.

7.3.1 Network of nodes, transient analysis

Consider a network of M stations type G/G/1/N with routing probabilities $r_{ij}(t)$. We follow the approach of [40] developed for the steady-state network model, then adapted to transient analysis in [28]. Here we introduce additionally, for the needs of SDN, the time-depending routing.

The first objective of the network model is to decompose the network: to determine the input flows at every station and then apply the single server model of the previous section to each station separately.

In the transient state, we should distinguish at any station i the input flow $\lambda_{i-in}(t)$ and the output flow $\lambda_{i-out}(t)$

$$\lambda_{i-out}(t) = [1 - p_{0i}(t)]\mu_i$$

which are different; $p_{0i}(t)$ denotes probability that the station i is idle at time t, i.e. the diffusion process related to this station is inside the barrier at $x = 0$. The term $1 - p_{0i}(t) = \varrho_i$ presents probability that the station i is busy and customers are leaving it with the rate μ_i.

The traffic equations balancing the flows of stations are

$$\lambda_{i-in}(t) = \lambda_{0i}(t) + \sum_{j=1}^{M} \lambda_{j-out}(t) r_{ji}(t)\,, \qquad i = 1, \ldots, M, \tag{7.48}$$

where the first term λ_{0i} represents traffic flow coming from the outside of the network directly to station i.

As mentioned earlier, routing probabilities $r_{ji}(t)$ are changing each interval Δ following decisions of the controler, remaining constant inside the interval, and flow parameters may change every interval $\delta < \Delta$; we assume for simplicity $\Delta = n\delta$, in numerical examples below $n = 10$. This way all model parameters are constant witin intervals δ when the solution (**??**) is computed.

Denote by $f_{Aj}(x,t)$ and $f_{Bj}(x,t)$ the density functions of interarrival and service times distributions at station j. The pdf $f_{Dj}(x,t)$ of the interdeparture times from this node at time t may be expressed as

$$f_{Dj}(x,t) = \varrho_j(t) f_{Bj}(x,t) + [1 - \varrho_j(t)] f_{Aj}(x,t) * f_{Bj}(x,t)\,, \quad j = 1, \ldots, M, \tag{7.49}$$

where * denotes the convolution. The first term of the right side in (7.49) represents the interdepature times of packets when the node j is working and the second term gives the interdeparture times when it is idle. The formula (7.49), known as Burke's theorem [**?**], is exact for Poisson input (the pdf of the idle period distribution that should be used in the second term of (7.49) is the same as $f_{Aj}(x,t)$) and approximate in other cases. From (7.49) we receive

$$C^2_{Dj}(t) = \varrho^2_j(t) C^2_{Bj}(t) + C^2_{Aj}(t)(1 - \varrho_j(t)) + \varrho_j(t)[1 - \varrho_j(t)]\,. \tag{7.50}$$

where $C^2_{Dj}(t)$, $C^2_{Bj}(t)$, $C^2_{Aj}(t)$ are time-dependent square coefficients of variation of interdeparture, service, and interarrival times, respectively. Packets leaving the node j according to the distribution $f_{Dj}(x,t)$ choose any node i with probability $r_{ji}(t)$ and the times between packets routed from node j to i has pdf $f_{ji}(x,t)$

$$\begin{aligned} f_{ji}(x,t) &= f_{Dj}(x,t) r_{ji}(t) + f_{Dj}(x,t) * f_{Dj}(x,t)[1 - r_{ji}(t)] r_{ji}(t) + \\ &\quad f_{Dj}(x,t) * f_{Dj}(x,t) * f_{Dj}(x,t)[1 - r_{ji}(t)]^2 r_{ji} + \cdots \end{aligned} \tag{7.51}$$

i.e. a packet leaving station j goes to station i with probability $r_{ji}(t)$ or with probability $1 - r_{ji}(t)$ it goes elswhere but the second goes to i with probability $r_{ji}(t)$, hence the gap has has pdf $f_{Dj}(x,t) * f_{Dj}(x,t)$ with probability $[1 - r_{ji}(t)] r_{ji}(t)$, etc, or, after Laplace transform

$$\begin{aligned} \bar{f}_{ji}(s,t) &= \bar{f}_{Dj}(s,t) r_{ji}(t) + \bar{f}_{Dj}(s,t)^2 [1 - r_{ji}(t)] r_{ji} + \\ &\quad + \bar{f}_{Dj}(s,t)^3 (1 - r_{ji}(t))^2 r_{ji} + \cdots \\ &= \frac{r_{ji}(t) \bar{f}_i(s,t)}{1 - [1 - r_{ji}(t)] \bar{f}_i(s,t)}\,, \end{aligned}$$

and we compute the squared coefficient of variation

$$C^2_{ji}(t) = r_{ji}(t)[C^2_{Dj}(t) - 1] + 1\,.$$

and then the parameters of the input flow at station i are given by (7.48) and (7.52)

$$C_{Ai}^2(t) = \frac{1}{\lambda_{i-in}(t)} \sum_{j=1}^{M} r_{ji}(t)\lambda_{i-out}(t)[(C_{Di}^2(t)-1)r_{ji}(t)+1] \; + \frac{C_{0i}^2(t)\lambda_{0i}(t)}{\lambda_{i-in}(t)} \; , \tag{7.52}$$

where the parameters λ_{0i} and C_{0i}^2 refer to the flow coming to station i from outside of the network.

Eqs. (7.50), (7.52) form a system of linear equations yielding $C_{Ai}^2(t)$ and, in consequence, the diffusion parameters $\beta_i(t)$, $\alpha_i(t)$ for every node i. At each interval δ, functions $f_i(x,t;\psi_i)$ giving queue distributions at every station i for $t \in \delta$ are computed. Their valuest at the end of the interval yield, among others, the current utilisations ϱ_i used to determine the flow parameters and diffusion parameters for the next interval δ. This way the flow parameters change each δ and routing changes each $\Delta = n\delta$.

The pdf $f_{Ri}(x,t)$ of the current response time (waiting time plus service) is determined using the first passage time from the end of the queue to zero.

The first passage time of the diffusion process from x_0 to $x = 0$ has the density function [**?**]

$$\gamma_{x_0,0}(t) \;\; = \;\; \frac{x_0}{\sqrt{2\Pi\alpha t^3}} e^{-\frac{(\beta t+1)^2}{2\alpha t}} \; .$$

with the Laplace transform

$$\bar{\gamma}_{x_0,0}(s) = e^{-x_0 \frac{\beta+\sqrt{\beta^2+2\alpha s}}{\alpha}} \; .$$

The starting point x_0 is determined by the function $f_i(x,t;\psi_i)$ hence

$$f_{Ri}(x,t) = \int_0^N \gamma_{\xi,0}(x) f_i(\xi,t;\psi_i) d\xi$$

If $f_{Ri}(x,t)$ is the response time pdf at node i, then the response time pdf $f_R(x,t)$ for the path $1,\ldots,n$ of n stations is

$$f_R(x,t) = f_{R1}(x,t) * f_{R2}(x,t) * f_{R3}(x,t) * \cdots * f_{Rn}(x,t),$$

or

$$\bar{f}_R(x,s) = \prod_{i=1}^{n} \bar{f}_{Ri}(x,s).$$

The loss probability $p_{loss}(t)$ for same entire path may be computed from

$$1 - p_{loss}(t) = (1-p_{N1}(t))(1-p_{N2}(t))(1-p_{N3}(t))\ldots(1-p_{Nn}(t)) \tag{7.53}$$

where $p_{Ni}(t)$ is probabiliity that the queue at station i is saturated at time t, i.e. the diffusion process for this station is at time t at the barrier $x = N$.

Example 7.1

The network consists of three stations connected in series — see Fig. **??**. *The input intensity is*

$$\lambda = 0.1 \quad \textit{for } t \in [0,10], \qquad \lambda = 4.0 \quad \textit{for } t \in [10,20].$$

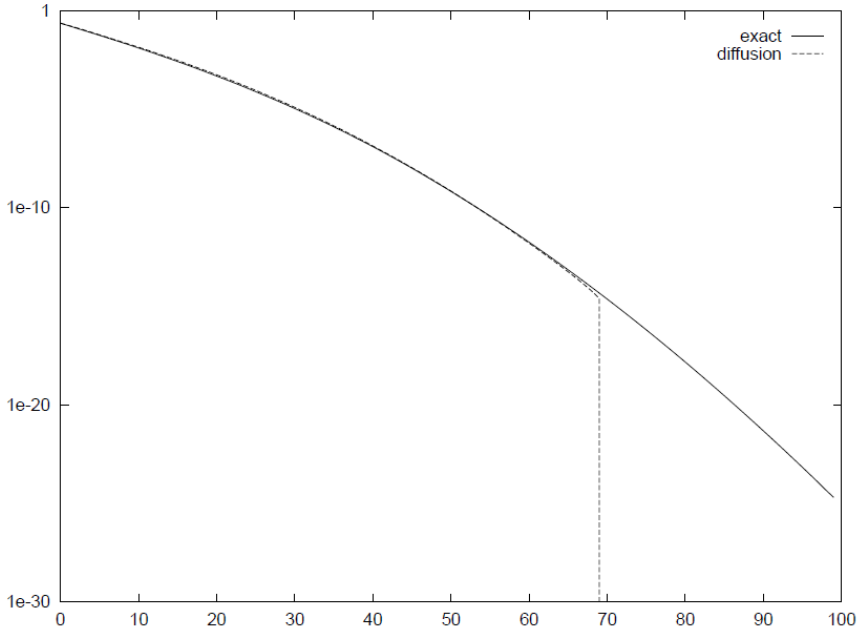


Figure 7.11: *M/M/1 station wit parameters* $\lambda = 0.75$ *and* $\mu = 1$ *Queue distribution at* $t = 75$ *results of diffusion approximation and simulatin.*

All stations are identical: initially, each queue contains

$$N_i(0) = 5$$

tasks, the maximum queue length is

$$N_i = 10,$$

and the parameters of the service-time distributions are

$$\mu_i = 2, \qquad C_{B_i}^2 = 1, \qquad i = 1, 2, 3.$$

Determine the evolution of the mean queue length at the stations over the interval

$$t \in [0, 40].$$

Figure 7.16 shows the changes in the mean queue length as a function of time. The first station reacts most strongly to changes in the load, whereas the changes are mildest at the third station.

■

Example 7.2

Figure 7.17 shows a system consisting of six stations and three sources. The source characteristics vary over time as follows:

source A:

$$\lambda_A = 0.1 \quad \textit{for } t \in [0, 10], \qquad \lambda_A = 4.0 \quad \textit{for } t \in [10, 21], \qquad \lambda_A = 0.1 \quad \textit{for } t \in [21, 40],$$

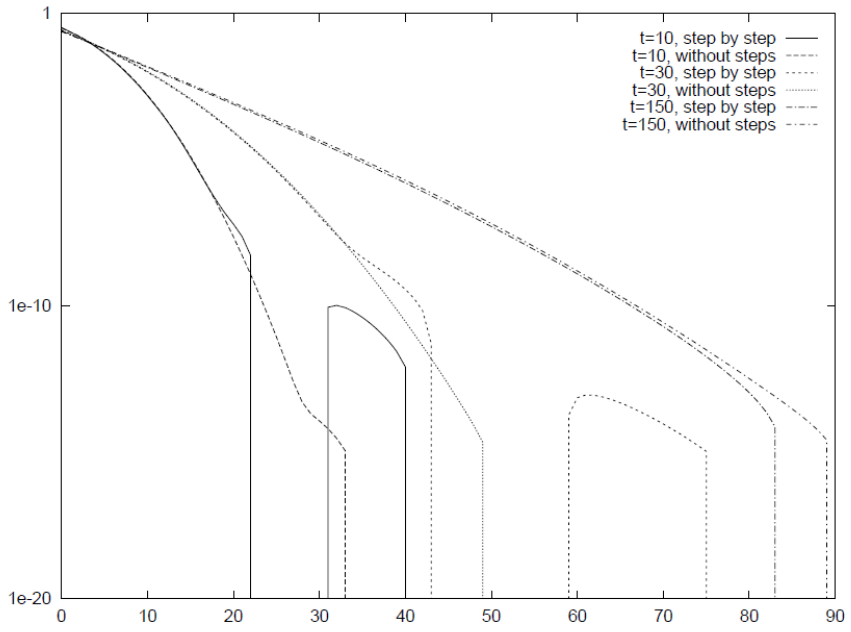


Figure 7.12: Model 1: mean queue lengths at successive stations as functions of time — results of the diffusion approximation and simulation (100,000 runs); also shown is the input intensity $\lambda(t)$.

source B:

$$\lambda_B = 0.1 \quad \textit{for } t \in [0, 11], \qquad \lambda_B = 4.0 \quad \textit{for } t \in [11, 20], \qquad \lambda_B = 0.1 \quad \textit{for } t \in [20, 40],$$

source C:

$$\lambda_C = 0.1 \quad \textit{for } t \in [0, 15], \qquad \lambda_C = 2.0 \quad \textit{for } t \in [15, 22], \qquad \lambda_C = 4.0 \quad \textit{for } t \in [22, 30],$$

$$\lambda_C = 2.0 \quad \textit{for } t \in [30, 31], \qquad \lambda_C = 0.1 \quad \textit{for } t \in [31, 40].$$

The service rate of station 6 also depends on time:

$$\mu_6 = 2 \quad \textit{for } t \in [0, 15] \textit{ and } t \in [31, 40], \qquad \mu_6 = 4 \quad \textit{for } t \in [15, 31].$$

The remaining parameters are constant:

$$N_1 = N_4 = 10, \qquad N_3 = 5, \qquad N_2 = N_5 = N_6 = 20,$$

$$\mu_1 = \cdots = \mu_5 = 2.$$

The transition probabilities between stations are

$$r_{12} = r_{13} = r_{14} = 0.33(3), \qquad r_{64} = 0.8.$$

The initial queue lengths are

$$N_1(0) = 5, \qquad N_2(0) = 10, \qquad N_3(0) = 10,$$

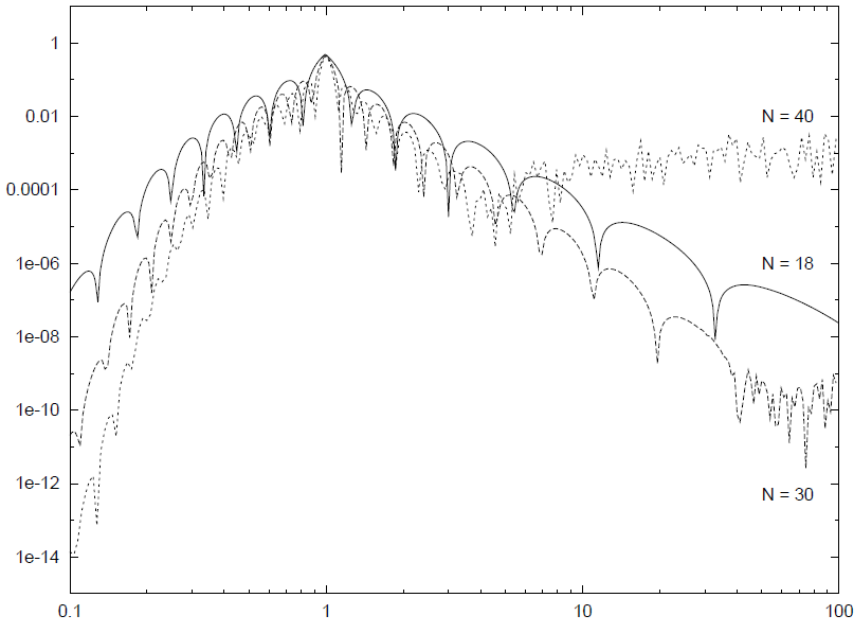


Figure 7.13: An error in the numerical inversion of the Laplace transform for the Heaviside step function $H(t)$ as a function of the parameter N in the Stehfest algorithm

$$N_4(0) = 5, \qquad N_5(0) = 5, \qquad N_6(0) = 10.$$

Determine the mean queue lengths as functions of time.

The results of the calculations are shown in Figs. 7.18, 7.19, and 7.20. As can be seen, they are not easy to predict intuitively. The results were compared with simulation data.

7.4 Diffusion model with state-dependent diffusion parameters

In the $G/G/1$ and $G/G/1/N$ systems, the input and output flows are independent of the number of tasks present at the station. The situation is different in the case of a station with multiple service channels, where the output flow depends on the number of occupied service channels, and in the case of a system with a limited customer population: the more customers are present at the station, the fewer remain outside it, and hence the less frequently new arrivals occur.

In such cases, changes in the number of tasks in the system depend on the number of tasks already present. We must take this into account in the diffusion model by choosing diffusion-process parameters that depend on the state of the process, that is, $\beta(x)$ and $\alpha(x)$.

7.4.1 The $G/G/1//N$ system model

In the model of a station with a limited population, we must take two facts into account:

- the number of potential customers is limited to N, and therefore the number of customers present at the station can never exceed N;

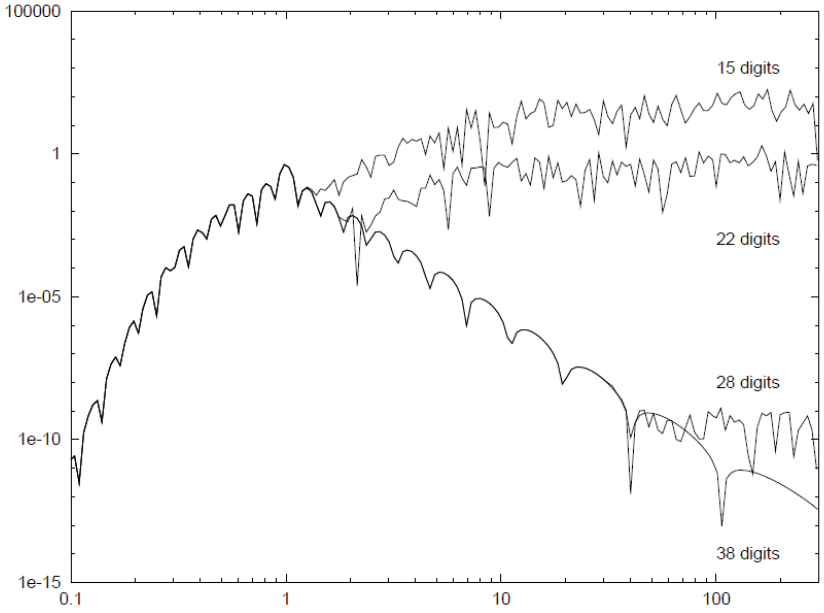


Figure 7.14: The error in the numerical inversion of the Laplace transform for the Heaviside step function $H(t)$ as a function of computational precision (the number of decimal places used in the calculations)

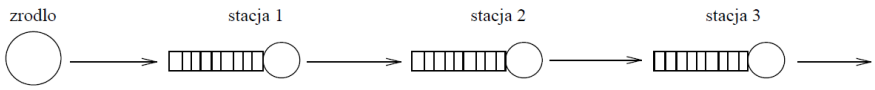


Figure 7.15: Model 1

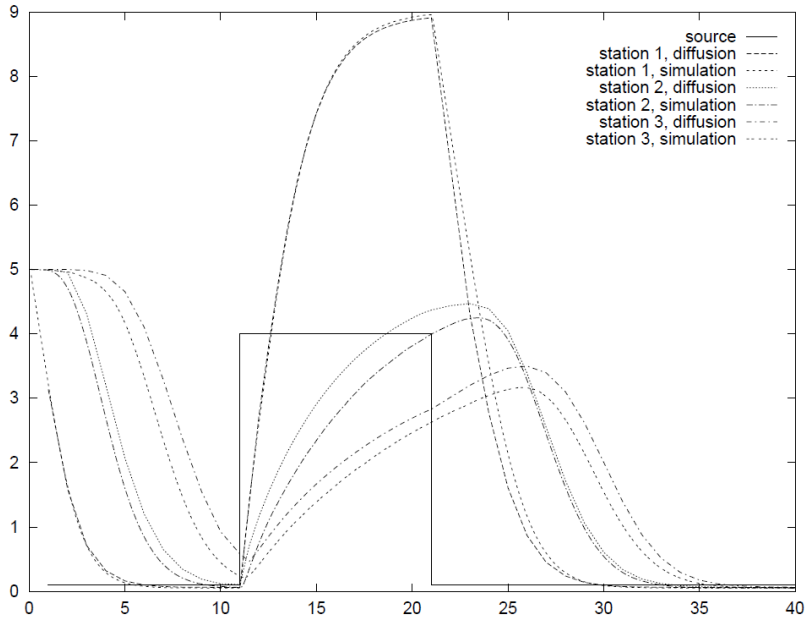


Figure 7.16: Model 1: Mean queue lengths at stations, results of the diffusion approximation and simulation (100,000 runs).

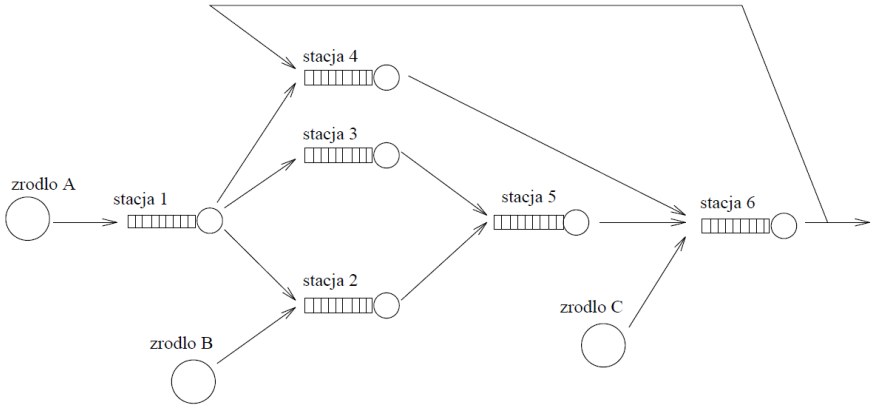


Figure 7.17: Model 2.

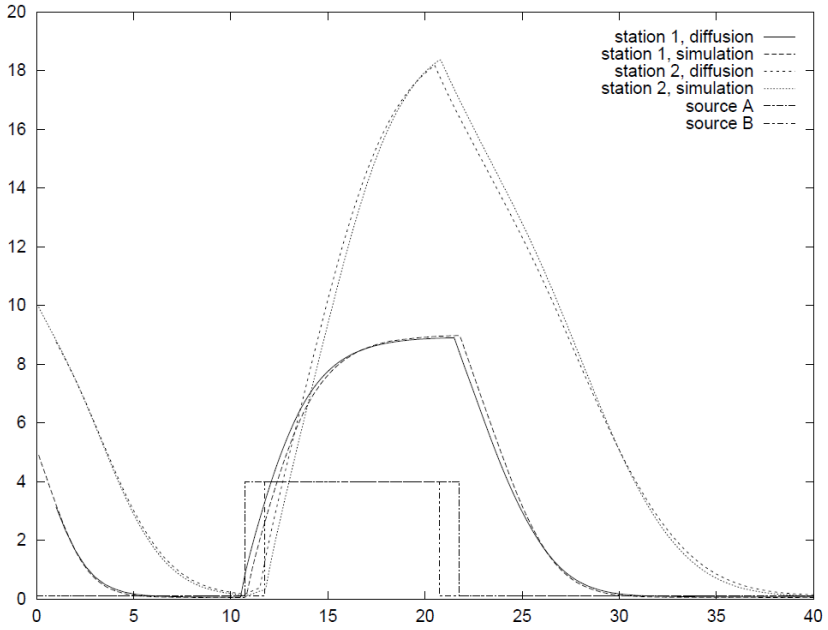


Figure 7.18: Model 2: Mean queue lengths at stations 1 and 2, results of the diffusion approximation and simulation (100,000 runs); the source A and B intensities are also shown.

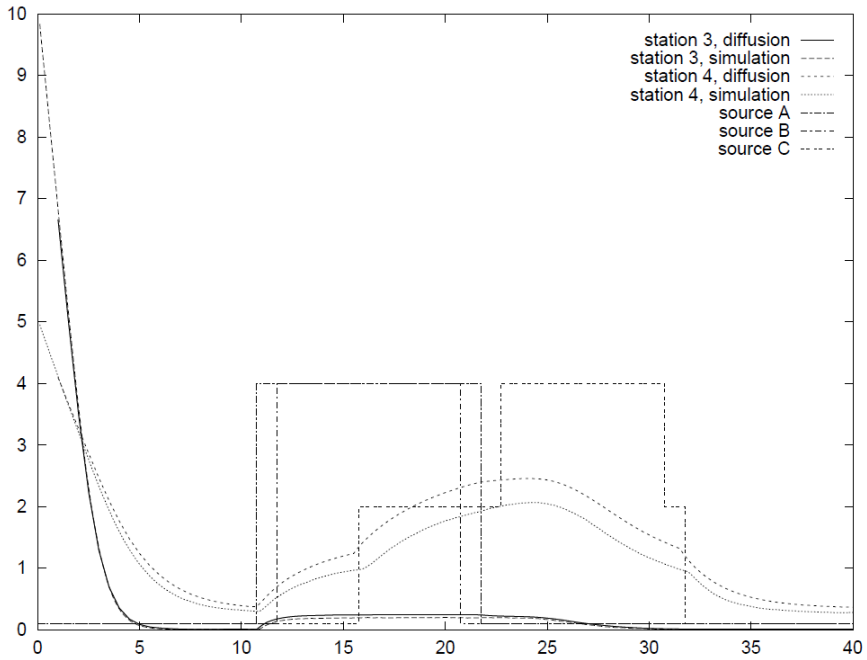


Figure 7.19: Model 2: Mean queue lengths at stations 3 and 4, results of the diffusion approximation and simulation (100,000 runs); the source A, B and C intensities are also shown

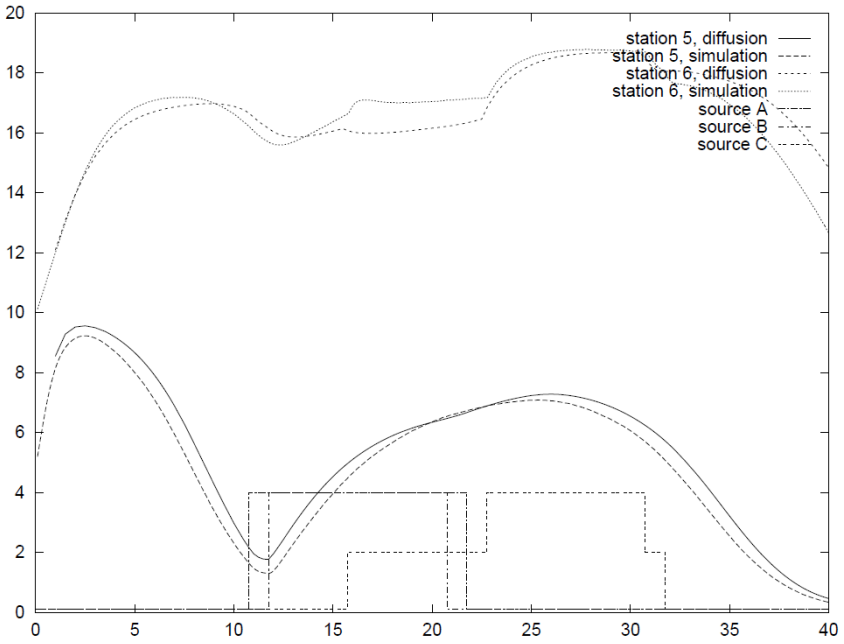


Figure 7.20: Model 2: Mean queue lengths at stations 5 and 6, results of the diffusion approximation and simulation (100,000 runs); the source A, B and C intensities are also shown

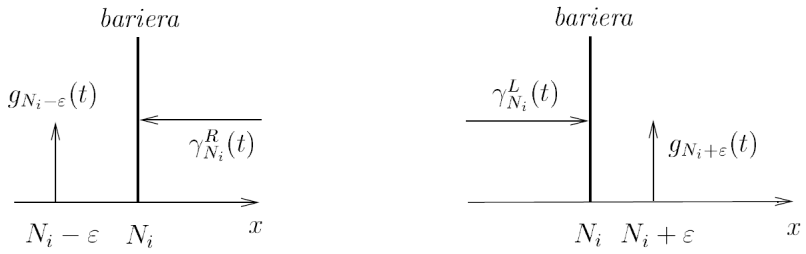


Figure 7.21: Flow balance for the barrier at $x = N_i$

- since the number of customers is limited, the arrival rate depends on the number of customers already present at the station — the more customers there are at the station, the less frequently the next one arrives.

The first of these facts can be represented by placing, as in the model of a bounded queue, a second barrier at $x = N$: when the process $X(t)$ reaches it, it remains there for a time having an exponential distribution with parameter λ_N (we assume this to be the reciprocal of the mean service time, that is, $\lambda_N = \mu$), and then jumps to the point $x = N - 1$.

The barrier at $x = 0$ behaves as before: it holds the process for a random time, after which diffusion resumes at $x = 1$. We assume that the time spent at the barrier $x = 0$ has an exponential distribution with parameter

$$\lambda_0 = N\lambda,$$

where $1/\lambda$ is the mean residence time of any task outside the station, that is, the time elapsing between the completion of service and the return of the customer to the queue.

The assumption concerning the distribution of the residence time at the barrier affects only the transient-state solution; the steady-state solution depends only on the first moment of this distribution [37].

The second fact is taken into account by assuming that the coefficients of the diffusion equation are not constant, but depend on the value taken by the process:

$$\alpha = \alpha(x), \qquad \beta = \beta(x).$$

The diffusion equations then take the form

$$\begin{aligned}
\frac{\partial f(x,t;x_0)}{\partial t} &= \frac{1}{2}\frac{\partial^2}{\partial x^2}\left[\alpha(x) f(x,t;x_0)\right] - \frac{\partial}{\partial x}\left[\beta(x) f(x,t;x_0)\right] \\
&\qquad + \lambda_0 p_0(t)\delta(x-1) + \lambda_N p_N(t)\delta(x-N+1)\,, \\
\frac{dp_0(t)}{dt} &= \lim_{x\to 0}\left\{\frac{1}{2}\frac{\partial}{\partial x}\left[\alpha(x) f(x,t;x_0)\right] - \beta(x) f(x,t;x_0)\right\} - \lambda_0 p_0(t)\,, \\
\frac{dp_N(t)}{dt} &= \lim_{x\to N}\left\{-\frac{1}{2}\frac{\partial}{\partial x}\left[\alpha(x) f(x,t;x_0)\right] + \beta(x) f(x,t;x_0)\right\} - \lambda_N p_N(t)\,, \qquad (7.54)
\end{aligned}$$

where $p_0(t)$ and $p_N(t)$ are the probabilities that the process is at the barriers:

$$p_0(t) = P[X(t) = 0], \qquad p_N(t) = P[X(t) = N]\,,$$

and $\delta(x)$ is the Dirac delta distribution.

Let us assume that the coefficients $\beta(x)$ and $\alpha(x)$ are constant on intervals:

$$\beta(x) = \beta_n, \qquad \alpha(x) = \alpha_n, \qquad x \in [n-1, n), \qquad n = 1, \ldots, N.$$

In steady state, equations (7.54) take the form

$$\begin{aligned}
0 &= \frac{\alpha_n}{2}\frac{d^2 f_n(x)}{dx^2} - \beta_n \frac{df_n(x)}{dx} + \lambda_0 p_0 \delta(x-1) + \lambda_N p_N \delta(x-N+1), \\
&\qquad\qquad n = 1, \ldots, N, \\
0 &= \lim_{x\to 0}\left[\frac{\alpha_1}{2}\frac{df_1(x)}{dx} - \beta_1 f_1(x)\right] - \lambda_0 p_0\,, \\
0 &= -\lim_{x\to N}\left[\frac{\alpha_N}{2}\frac{df_N(x)}{dx} - \beta_N f_N(x)\right] - \lambda_N p_N\,. \qquad (7.55)
\end{aligned}$$

If we assume that:

- apart from jump points, the barriers behave like absorbing barriers, which corresponds to the conditions
$$\lim_{x\to 0} f(x,t;x_0) = \lim_{x\to N} f(x,t;x_0) = 0,$$
- the function $f(x)$ is continuous:
$$f_n(n) = f_{n+1}(n), \qquad n = 1,\ldots,N-1,$$
- there is no probability-mass flow between intervals in regions where no jumps occur, that is, for $x \in (1, N-1)$,
$$\frac{\alpha_n}{2}\frac{df_n(x)}{dx} - \beta_n f_n(x) = 0, \qquad x \in (n-1,n), \qquad n = 2,\ldots,N-1,$$

then we obtain the solution of equation (7.55) [20]:

$$\begin{aligned}
f_1(x) &= \begin{cases} \dfrac{\lambda_0 p_0}{-\beta_1}\left(1 - e^{z_1 x}\right) & \text{for } \beta_1 \neq 0 \\ \dfrac{2\lambda_0 p_0}{\alpha_1} x & \text{for } \beta_1 = 0 \end{cases} \qquad 0 < x \leq 1, \\
f_n(x) &= \begin{cases} f_{n-1}(n-1)e^{z_n(x-n+1)} & \text{for } \beta_n \neq 0 \\ f_{n-1}(n-1) & \text{for } \beta_n = 0 \end{cases} \qquad \begin{array}{l} n-1 \leq x \leq n, \\ n = 2,\ldots,N-1, \end{array} \\
f_N(x) &= \begin{cases} \dfrac{\mu p_N}{-\beta_N}\left[e^{z_N(x-N)} - 1\right] & \text{for } \beta_N \neq 0 \\ \dfrac{-2\mu p_N}{\alpha_N}(x-N) & \text{for } \beta_N = 0 \end{cases} \qquad N-1 \leq x < N.
\end{aligned} \tag{7.56}$$

where
$$z_n = \frac{2\beta_n}{\alpha_n}\,.$$

In the case of a single customer class, we assume

$$\beta_n = (N-n+1)\lambda - \mu,$$

$$\alpha_n = (N-n+1)\lambda C_A^2 + \mu C_B^2, \qquad x \in [n-1,n), \qquad n = 1,\ldots,N.$$

In the case of **multiple customer classes**, let the customer population consist of K classes and be described by the vector

$$\boldsymbol{N} = [N^{(1)}, N^{(2)}, \ldots, N^{(K)}],$$

where $N^{(k)}$ denotes the number of customers in class k, and

$$\sum_{k=1}^{K} N^{(k)} = N.$$

Each class has its own interarrival-time distribution $A^{(k)}(x)$ and service-time distribution $B^{(k)}(x)$.

Let $\vartheta^{(k)}$ denote the probability that a customer chosen at random from the population outside the station belongs to class k, and let $q^{(k)}$ denote the probability that a customer chosen at random from the station queue belongs to class k.

To determine the parameters of the arrival flow, we sum the variances of the numbers of arriving customers of the individual classes per unit time:

$$\lambda C_A^2 = \sum_{k=1}^{K} \vartheta^{(k)} \lambda^{(k)} {C_A^{(k)}}^2.$$

We represent the aggregate service-time distribution as

$$b(x) = \sum_{k=1}^{K} q^{(k)} b^{(k)}(x)\,,$$

and then calculate its mean and variance:

$$\begin{aligned} \frac{1}{\mu} &= E[B] = \sum_{k=1}^{K} q^{(k)} E[B^{(k)}] = \sum_{k=1}^{K} \frac{q^{(k)}}{\mu^{(k)}}\,, \\ C_B^2 &= \mu^2 \sum_{k=1}^{K} \frac{q^{(k)}}{(\mu^{(k)})^2} \left({C_B^{(k)}}^2 + 1 \right) - 1\,. \end{aligned} \tag{7.57}$$

Hence the coefficients α_n and β_n are

$$\begin{aligned} \beta_n &= (N-n+1) \sum_{k=1}^{K} \vartheta^{(k)} \lambda^{(k)} - \mu\,, \\ \alpha_n &= (N-n+1)\lambda C_A^2 + \mu C_B^2\,. \end{aligned} \tag{7.58}$$

We determine the probabilities $q^{(k)}$ and $\vartheta^{(k)}$ by an iterative method [20]:

1. Initially, we assume
$$q^{(k)} = \vartheta^{(k)} = \frac{N^{(k)}}{N},$$
which is exact when all classes have identical distributions.

2. We determine the distribution $p(n)$ from the function (7.56), and in particular the station utilisation factor
$$\varrho = 1 - p(0).$$

3. We determine the average waiting time in the queue, $E[W]$, which is the same for customers of all classes:
$$E[W] = \frac{E[N] - \varrho}{\varrho\mu}\,.$$
Here, $E[N] - \varrho$ is the average queue length (excluding the customer currently in service), while $\varrho\mu = \lambda$ is the station throughput.

4. We recalculate the values of $\vartheta^{(k)}$ and $q^{(k)}$:

$$\vartheta^{(k)} = \frac{\dfrac{N^{(k)}\lambda^{(k)\,-1}}{\lambda^{(k)\,-1}+E[W]+\mu^{(k)\,-1}}}{\displaystyle\sum_{l=1}^{K}\frac{N^{(l)}\lambda^{(l)\,-1}}{\lambda^{(l)\,-1}+E[W]+\mu^{(l)\,-1}}},$$

$$q^{(k)} = \frac{\dfrac{N^{(k)}\left(E[W]+\mu^{(k)\,-1}\right)}{\lambda^{(k)\,-1}+E[W]+\mu^{(k)\,-1}}}{\displaystyle\sum_{l=1}^{K}\frac{N^{(l)}\left(E[W]+\mu^{(l)\,-1}\right)}{\lambda^{(l)\,-1}+E[W]+\mu^{(l)\,-1}}}. \tag{7.59}$$

5. We return to step (2) until convergence is achieved.

A system with a finite population of tasks that return to the queue after completion of service is the classical model known as the machine-repairman (or finite-source) model and is used in operations research to describe many situations.

The diffusion approximation allows this model to be generalised to multiple customer classes and arbitrary distributions of interarrival times and service times. The solution is, of course, approximate, but its accuracy is acceptable [20]. Below, we apply the finite-population model as a component of the closed-loop system $G/G/1$.

7.5 Approximation of the $G/G/c/N$ system

We can also use the solution (7.56) to describe a system with multiple service channels: let this be a $G/G/c/N$ system, i.e. a system with the number of tasks limited to N, equipped with c parallel service channels.

The coefficients of the diffusion equation for this model are:

$$\begin{aligned}
\beta_n &= \lambda - n\mu\,, \\
\alpha_n &= \lambda C_A^2 + n\mu C_B^2\,, \qquad x\in[n-1,n), \\
&\qquad\qquad\qquad\quad n=1,\ldots,c-1\,, \\
\beta_c &= \lambda - c\mu\,, \\
\alpha_c &= \lambda C_A^2 + c\mu C_B^2\,, \qquad x\in[c-1,N)\,.
\end{aligned} \tag{7.60}$$

That is, we assume that the interval $x\in(0,1)$ corresponds to the operation of one service channel, the interval $x\in(1,2)$ corresponds to the operation of two channels, and for $x\in(c-1,N)$ all service channels are active.

7.6 Approximation of a closed network of $G/G/1$ systems

Let a closed network of M systems of type $G/G/1$ contain N customers belonging to a single class. The matrix

$$\boldsymbol{R} = [r_{ij}]_{i=1,\ldots,M;\ j=1,\ldots,M}$$

defines the transition probabilities between stations.

To analyse the operation of individual stations in this network, one can apply the $G/G/1//N$ model of a service station serving a finite population of at most N customers. A given station views the remainder of the network as an external source capable of generating no more than N customers.

The parameters of this equivalent source must be determined iteratively. This is necessary because the parameters of the input stream to a given station depend on the utilisation factors of the other stations, which in turn depend on the parameters of their own input streams; this dependence is nonlinear.

The following network analysis algorithm is proposed:

1. For each station i, $i = 1, \ldots, M$, replace the rest of the network by an equivalent source generating customers at time intervals with a Poisson distribution having parameter $\lambda(n)$, $n = 1, \ldots, N$.

   Initially, we assume that all stations other than station i have exponential service-time distributions and solve the remainder of the network using the BCMP model.

   The parameter $\lambda(n)$, $n = 1, \ldots, N$, is the throughput of the network in which station i is excluded from service, that is, we assume $1/\mu_i = 0$ when there are n customers in the network.

   The Markov model then allows us to determine the utilisation factors

   $$\varrho_j, \qquad j = 1, \ldots, i-1, i+1, \ldots, M,$$

   for all remaining stations in the network.

2. We then take into account the actual service-time distribution and, using the approximate utilisation coefficients ϱ_j, calculate the second moments of the output flows of the individual stations, C_{Dj}^2:

   $$C_{Dj}^2 = C_{Aj}^2(1 - \varrho_j) + \varrho_j^2 C_{Bj}^2 + \varrho_j(1 - \varrho_j) \tag{7.61}$$

   and the second moment of the input flow to the station i under consideration:

   $$C_{Ai}^2 = \frac{1}{\lambda_i} \sum_{j=1}^{M} r_{ji}\lambda_j \left[(C_{Dj}^2 - 1)r_{ji} + 1\right]. \tag{7.62}$$

3. Using the solution (7.56), we determine $p_i(n)$, and subsequently the new values of the utilisation factor

   $$\varrho_i = 1 - p_i(0),$$

   the throughput

   $$\lambda_i = \varrho_i \mu_i,$$

   and the coefficient C_{Di}^2.

4. We use the parameters determined in the previous step to recalculate C_{Aj}^2 and $p_i(n)$ for each station, and then return to step (2) until the iteration converges to a fixed point.

5. Finally, we check whether

$$\sum_{i=1}^{M} E[N_i] = N,$$

and, if necessary, rescale the results.

Comparison with simulation indicates that this approach is recommended for a single customer class; for multiple classes the accuracy of the results is unsatisfactory, particularly when the service-time distributions in the network differ significantly from the exponential distribution [115].

7.7 Approximation of the $G/G/1$ station with preemptive priority policy

Let K be the number of task classes arriving at the station. The method for determining the parameters specific to class k, $k = 1, \ldots, K$, is the same as in the case of a multi-class queue with natural priority, Subsection 7.1.1.

We assume that priority decreases as the value of k increases: tasks of class 1 have the highest priority, whereas tasks of class K have the lowest priority.

For customers of class k, the presence of tasks of lower-priority classes, that is, classes $k+1, \ldots, K$, is irrelevant. Let $\varrho^{(k)}$ denote the load on the station generated by class-k tasks, and let

$$R^{(k)} = \sum_{i=1}^{k} \varrho^{(i)}$$

denote the total load due to tasks of classes $1, \ldots, k$.

Taking into account that a task can complete service and leave the system at time t only if it is present in the system and there are no higher-priority tasks waiting, the departure process for tasks of classes $1, \ldots, k$ can be expressed as

$$\sum_{i=1}^{k} \frac{\varrho^{(i)}}{R^{(k)}} D^{(i)},$$

where $D^{(i)}$ denotes the departure process of class i in the case where only that class uses the station.

The diffusion process describing the total number of tasks of classes $1, \ldots, k$ at the station has parameters

$$\beta^{(k)} = \sum_{i=1}^{k} \lambda^{(i)} - \sum_{i=1}^{k} \frac{\varrho^{(i)}}{R^{(k)}} \mu^{(i)},$$

$$\alpha^{(k)} = \sum_{i=1}^{k} \lambda^{(i)} {C_A^{(i)}}^2 + \sum_{i=1}^{k} \frac{\varrho^{(i)}}{R^{(k)}} \mu^{(i)} {C_B^{(i)}}^2.$$

By solving the diffusion equation with these coefficients, we obtain a density function approximating the number of customers of classes $1, \ldots, k$ present at the station.

Let $P^{(k)}(n)$ denote the distribution of the total number of customers of classes $1, \ldots, k$, and let $p^{(k)}(n)$ denote the distribution of the number of customers of class k alone.

For the highest-priority class,

$$P^{(1)}(n) = p^{(1)}(n).$$

For the remaining classes,

$$P^{(k)}(n) = \sum_{i=0}^{n} P^{(k-1)}(n-i)\, p^{(k)}(i), \tag{7.63}$$

which implies

$$p^{(k)}(n) = \frac{P^{(k)}(n) - \sum_{i=0}^{n-1} P^{(k-1)}(n-i)\, p^{(k)}(i)}{P^{(k-1)}(0)}. \tag{7.64}$$

The time from the start of service of a task of class k until the completion of its processing is called the completion time, denoted $C^{(k)}$.

It is equal to the sum of the service time of the task itself and all interruption intervals caused by the arrival of higher-priority tasks of classes $1, \ldots, k-1$. Each such interruption interval has a duration corresponding to the busy period of the station generated by tasks of classes $1, \ldots, k-1$.

7.8 Fork–Join Model

A diffusion approximation can also be used to describe the **fork–join system**. An input stream with parameters $1/\lambda_0$ and C_{A0}^2 splits at the *fork* into L streams. More generally, we may assume that a task is directed to branch i with probability p_i. We may also assume that service times in the individual branches are different and arbitrary, with parameters $1/\mu_i$ and σ_{Bi}^2.

The input parameters of the individual branches are then

$$\lambda_i = p_i \lambda_0 \,, \qquad\qquad C_{Ai}^2 = (C_{A0}^2 - 1)p_i + 1 \,. \tag{7.65}$$

Knowing these parameters, each of the L parallel stations may be treated as independent, and their analysis presents no difficulty. We can also calculate the output flow from each of these stations. The input flow to the *join* queue is then determined as the superposition of the output flows from the parallel stations:

$$\lambda_f = \sum_{i=1}^{L} \lambda_i \,, \qquad\qquad C_f^2 = \frac{1}{\lambda_f} \sum_{i=1}^{L} \lambda_i C_{Di}^2 \,. \tag{7.66}$$

For the *join* queue, a new diffusion model must be developed, based on supplementing the diffusion process with discrete downward jumps.

7.8.1 Diffusion model of the join queue

Let the process $X(t)$ be defined on the interval $[0, S]$. The interval $(0, S)$ is divided into Q subintervals of equal length L, such that $S = LQ$. We assume that $L > 1$.

We call the interval $(0, L)$ the first interval, and the interval $[(i-1)L,\, iL)$, $i = 2, \ldots, Q$, the i-th interval.

In the first interval, $X(t)$ is an ordinary diffusion process. In the remaining intervals, in addition to diffusion, the process may make jumps of fixed length L, from point x to point $x - L$, thus moving from interval i to interval $i - 1$.

Jumps in interval i occur with rate γ_i, which is constant within that interval. The density function of the process in interval i is denoted by $f_i(x)$.

The steady-state equations describing the process take the form

$$\begin{aligned}
0 &= \frac{\alpha}{2}\frac{d^2 f_1(x)}{dx^2} - \beta\frac{df_1(x)}{dx} + \lambda_0 p_0 \delta(x-1) + \gamma_2 f_2(x+L) \ , \\
0 &= \frac{\alpha}{2}\frac{d^2 f_i(x)}{dx^2} - \beta\frac{df_i(x)}{dx} - \gamma_i f_i(x) + \gamma_{i+1} f_{i+1}(x+L) \ , \qquad i = 2, \ldots, Q-1, \\
0 &= \frac{\alpha}{2}\frac{d^2 f_Q(x)}{dx^2} - \beta\frac{df_Q(x)}{dx} - \gamma_Q f_Q(x) + \mu_S p_S \delta(x - LQ + L) \ , \\
0 &= \lim_{x\to 0}\left[\frac{\alpha}{2}\frac{df_1(x)}{dx} - \beta f_1(x)\right] - \lambda_0 p_0 \ , \\
0 &= -\lim_{x\to S}\left[\frac{\alpha}{2}\frac{df_Q(x)}{dx} - \beta f_Q(x)\right] - \mu_S p_S \ . \qquad (7.67)
\end{aligned}$$

We assume that the function $f(x)$ is continuous:

$$f_i(iL) = f_{i+1}(iL) \ , \qquad i = 1, \ldots, Q-1, \qquad (7.68)$$

and that the probability mass flows between adjacent intervals are balanced:

$$\lim_{x\to iL}\left[\frac{\alpha}{2}\frac{df_i(x)}{dx} - \beta f_i(x)\right] = \gamma_{i+1}\int_{iL}^{(i+1)L} f_{i+1}(x)\,dx \ , \qquad i = 1, \ldots, Q-1. \qquad (7.69)$$

We first solve the equations for the Q-th interval. The function $f_Q(x)$ is the sum of two exponential terms with exponents $z_{Q,1}$ and $z_{Q,2}$:

$$f_Q(x) = c_{Q,1} e^{z_{Q,1}x} + c_{Q,2} e^{z_{Q,2}x} \ . \qquad (7.70)$$

The coefficients $z_{i,1}$ and $z_{i,2}$ are determined from

$$z_i^2 - \frac{2\beta}{\alpha} z_i - \frac{2\gamma_i}{\alpha} = 0 \ ,$$

hence

$$z_{i,1} = \frac{\beta}{\alpha} - \sqrt{\frac{\beta^2}{\alpha^2} + \frac{2\gamma_i}{\alpha}} \ ,$$

$$z_{i,2} = \frac{\beta}{\alpha} + \sqrt{\frac{\beta^2}{\alpha^2} + \frac{2\gamma_i}{\alpha}} \ , \qquad i = 2, \ldots, Q.$$

From the boundary condition at $x = S$, we calculate the coefficients $c_{Q,1}$ and $c_{Q,2}$ as functions of the product $\mu_S p_S$, whose value is not yet known:

$$c_{Q,1} = \frac{2\mu_S p_S}{\alpha\,(z_{Q,1} - z_{Q,2})} e^{-z_{Q,1}S} \ ,$$

$$c_{Q,2} = \frac{-2\mu_S p_S}{\alpha\,(z_{Q,1} - z_{Q,2})} e^{-z_{Q,2}S} \ .$$

For each interval i, $i = Q-1, \ldots, 1$, the diffusion equation defining $f_i(x)$ contains the term $f_{i+1}(x+L)$. Therefore, the function $f_i(x)$ consists of exponentials containing $z_{i,1}$, $z_{i,2}$, and all z-coefficients that appear in $f_{i+1}(x+L)$.

Thus:

$$f_{Q-1}(x)$$

is the sum of four exponential terms,

$$f_{Q-2}(x)$$

is the sum of six exponential terms, and so on.

The general solution therefore takes the form

$$f_i(x) = \sum_m c_{i,m} e^{z_{i,m} x} \ ,$$

where the coefficients $c_{i,m}$ are determined recursively from the continuity conditions (7.68), the balance conditions (7.69), and the boundary conditions at $x = 0$ and $x = S$.

$$\begin{aligned}
f_1^a(x) &= c_{1,1} + c_{1,2} e^{z_{1,2} x} + \sum_{l=0}^{Q-2} [c_{1,2l+3} e^{z_{l+2,1} x} + c_{1,2l+4} e^{z_{l+2,2} x}] \\
&\qquad\qquad \text{for } 0 < x \leq 1, \\
f_1^b(x) &= \sum_{l=0}^{Q-1} [c_{1,2l+1} e^{z_{l+1,1} x} + c_{1,2l+2} e^{z_{l+1,2} x}] \\
&\qquad\qquad \text{for } 1 \leq x \leq L, \\
f_i(x) &= \sum_{l=0}^{Q-i} [c_{i,2l+1} e^{z_{i+l,1} x} + c_{i,2l+2} e^{z_{i+l,2} x}] \\
&\qquad\qquad \text{for } (i-1)L \leq x < iL, \qquad i = 2, \ldots, Q. \quad (7.71)
\end{aligned}$$

Conditions (7.68) and (7.69) allow the coefficients $c_{i,j}$ to be expressed in terms of previously calculated coefficients:

$$\begin{aligned}
c_{i,2l+1} &= \frac{-2\gamma_{i+1}}{\alpha\,(z_{i,1} - z_{i,2})} c_{i+1,2l-1} e^{z_{i+l,1} L} \left(\frac{1}{z_{i+l,1} - z_{i,1}} - \frac{1}{z_{i+l,1} - z_{i,2}} \right) \ , \\
c_{i,2l+2} &= \frac{-2\gamma_{i+1}}{\alpha\,(z_{i,1} - z_{i,2})} c_{i+1,2l-1} e^{z_{i+l,2} L} \left(\frac{1}{z_{i+l,2} - z_{i,1}} - \frac{1}{z_{i+l,2} - z_{i,2}} \right) \ .
\end{aligned}$$

Normalisation determines the value of the product $\mu_S p_S$, and thus the correct values of all coefficients $c_{i,j}$.

The back-stepping diffusion process represents the total number of tasks from all parallel branches and different generations present in the join queue. When all L sibling tasks from a given generation are present, they merge and leave together; consequently, the queue length decreases abruptly by L.

The length of the join queue is unlimited. The probability of completing a generation and leaving the queue increases with the queue length. In the model, this is described by the dependence of the coefficient γ on the value of the process:

$$\gamma = \gamma(x) \ .$$

This probability also depends on the distribution of service times in the individual branches, which affects the time interval between the completion of the first and last task in a given generation.

If all L parallel branches operate at the same rate, we may, as a first approximation, assume

$$\gamma_i = (i-1)\lambda_0 \ , \qquad i = 2, \ldots, Q.$$

The diffusion process is restricted to the interval

$$0 \le x \le S.$$

The value of S must be sufficiently large so that the probability p_S of the process being at the right-hand barrier is very small, that is, increasing S further does not significantly change p_S. However, p_S cannot be exactly zero, since all coefficients $c_{i,j}$ are determined as functions of $\mu_S p_S$.

Note that any configuration may be placed between the fork and join stations, and this does not create difficulties in determining the coefficients λ_j and C_{Aj}^2.

The advantage of the method is that it can handle arbitrary input streams and arbitrary load distributions across the parallel branches. Its disadvantages are the risk of numerical instability and the unresolved problem of selecting the coefficients γ_i.

7.9 Approximation of waiting time in the $G/G/1$ system

Apart from approximating the number of tasks in the system, the diffusion process may also be used to approximate the waiting time. There are far fewer works devoted to such models; see, for example, [35, 46]. The following results are taken from [21, 22].

Let $W(t)$ denote the waiting time in the queue for a task arriving at the system at time t. We retain all previous notation concerning the arrival process and the service-time distribution.

The arrival of a new task increases the value of this random process by a jump equal to the service time of the new task, which is a random variable with density $b(x)$.

Apart from arrival epochs, provided that $W(t) > 0$, the process decreases linearly as the server operates and the waiting time is reduced.

The mean and variance of the diffusion process $X(t)$, which approximates the waiting time, are described by the parameters (cf. [38])

$$\beta = \frac{\lambda}{\mu} - 1 \ , \qquad \alpha = \frac{\lambda}{\mu^2}\left(C_A^2 + C_B^2\right) \ .$$

Jumps from the barrier placed at $x = 0$ correspond to the arrival of the first customer in a new busy period and have a random magnitude determined by the density $b(x)$. Thus, the first term of equation (7.7) takes the form

$$\frac{\partial f(x,t;x_0)}{\partial t} = \frac{\alpha}{2}\frac{\partial^2 f(x,t;x_0)}{\partial x^2} - \beta\frac{\partial f(x,t;x_0)}{\partial x} + \lambda p_0(t) b(x) \ . \tag{7.72}$$

which, in the steady state, yields a solution that depends on the form of the service-time distribution $B(x)$:

$$f(x) = \frac{2\lambda p_0}{\alpha} e^{zx} \int_x^\infty \left[1 - B(u)\right] e^{-zu}\, du \ ,$$

where

$$z = \frac{2\beta}{\alpha} .$$

In the case of an exponential service-time distribution,

$$B(x) = 1 - e^{-\mu x} ,$$

we obtain

$$f(x) = \frac{2\lambda p_0}{\alpha} \frac{1}{\mu + z} \left(e^{zx} - e^{-\mu x} \right) .$$

This result can easily be extended to service-time distributions that are linear combinations of exponential distributions. For example, for the second-order Cox distribution,

$$b(x) = (1-a)\mu_1 e^{-\mu_1 x} + a\,\mu_1 e^{-\mu_1 x} * \mu_2 e^{-\mu_2 x} ,$$

the density function of the diffusion process takes the form

$$f(x) = \frac{2\lambda p_0}{\alpha} \left[\frac{b_1}{\mu_1 + z} \left(e^{zx} - e^{-\mu_1 x} \right) + \frac{b_2}{\mu_2 + z} \left(e^{zx} - e^{-\mu_2 x} \right) \right] .$$

The Chapman–Kolmogorov equation

$$f(x,t;y,\tau) = \int_{-\infty}^{\infty} f(w,u;y,\tau)\, f(x,t;w,u)\, dw$$

follows from the relation

$$\frac{\partial f(x,t)}{\partial t} = \sum_{n=1}^{\infty} \frac{(-1)^n}{n!} \frac{\partial^n}{\partial x^n} \left[A_n(x,t) \right] .$$

For the **transient state**, the solution can be determined through the superposition of densities ϕ; see equation (7.15), which describes the diffusion process with an absorbing barrier at $x = 0$, similarly to the case of the diffusion process approximating the number of customers; cf. equation (7.17).

The only difference is that the component processes described by ϕ no longer start at the point $x_0 = 1$, but rather at points ξ, distributed according to the probability density $b(\xi)$:

$$f(x,t;x_0) = \phi(x,t;x_0) + \int_0^t g(\tau) \int_0^{\infty} \phi(x, t-\tau; \xi)\, b(\xi)\, d\xi\, d\tau . \tag{7.73}$$

The probability flow $g(\tau)$ of the process leaving the barrier at $x = 0$ and jumping to an arbitrary point ξ is determined in the same way as the flow $g_1(\tau)$ in equation (7.19).

When determining the flow $\gamma_0(t)$ entering the barrier, we must now take into account the different starting points ξ of the process:

$$\gamma_0(t) = p_0(0)\delta(t) + (1 - p_0(0))\, \gamma_{x_0,0}(t) + \int_0^t g(\tau) \int_0^{\infty} \gamma_{\xi,0}(t-\tau)\, b(\xi)\, d\xi\, d\tau . \tag{7.74}$$

Chapter 8

Diffusion approximation, applications

The examples of the use of diffusion approximations presented in this chapter relate to the modelling of phenomena associated with information transmission in modern wide-area integrated-services networks. This is an area in which queueing models are currently proving particularly useful.

Information streams are heterogeneous in nature; some, such as voice transmission, have a nearly constant intensity, which results either from the nature of the medium or from the use of buffers that smooth fluctuations in the intensity of the transmitted information. For such streams, a constant link bandwidth can be allocated; other services, such as video transmission, distributed computing and the associated data exchange, interconnection of local-area networks via an ATM network, or access to multimedia databases, are characterised by highly variable information flow rates. The maximum transmission rate may exceed the average traffic rate many times over, sometimes by an order of magnitude or more. Periods of high traffic intensity alternate with periods during which traffic is relatively low.

Allocating to such users a bandwidth corresponding to their maximum requirements would result in highly inefficient use of network resources. Therefore, variable bit rate services are introduced, in which bandwidth is allocated statistically rather than deterministically. The total link bandwidth is significantly lower than the sum of the maximum traffic rates generated by all users. It is assumed that the probability of simultaneous peak demand from several users is sufficiently small.

As a consequence, network traffic is subject to random fluctuations, and the average traffic load should remain sufficiently high to ensure good utilisation of network resources. However, it also introduces the risk of congestion.

At times of increased traffic intensity, queues of packets waiting to be transmitted at network nodes begin to grow. Packets are stored in finite buffers, and therefore it may happen that, during periods of heavy traffic, the buffer becomes full and newly arriving packets are lost. The buffer size must therefore be chosen so that the probability of loss is sufficiently low.

Depending on the packet contents, packet loss and the delay caused by retransmission may be more or less significant. For ordinary data transmission, losses are usually less critical; for moving-image transmission, such as teleconferencing, packet loss is much more problematic.

For this reason, packets are often divided into two categories: high-priority and ordinary packets. The buffer size and the queueing discipline must be selected so that the probability of

losing a high-priority packet is, for example, of the order of 10^{-12}, whereas the probability of losing a lower-priority packet may be of the order of 10^{-6}.

Buffers should therefore be dimensioned so that, given the statistical properties of the traffic stream, they can absorb temporary increases in traffic intensity, while ensuring that the probability of loss due to buffer overflow remains below the required threshold. This is one of the fundamental parameters determining the quality of network service.

Since packet-loss events are very rare, the use of simulation for modelling network traffic becomes extremely time-consuming. For example, if the loss of a high-priority packet occurs on average once per 10^{12} transmitted packets of that class, then it would be necessary to simulate the transmission of approximately 10^{16} such packets in order to obtain statistically significant results.

8.1 Packet-switched networks

We now present an example of the use of the diffusion model of an open network of $G/G/1$ nodes to analyse the steady-state behaviour of a packet-switched network with variable-length routing paths.

Example 8.1

A packet-switched network is to be analysed whose topology is shown in Fig. 8.1. The black points denote the network nodes. We assume that the packet streams generated by users at node i arrive according to a Poisson process with intensity λ_{0i}.

Packet traffic in the network is described using two matrices. The matrix $\mathbf{D}$ *contains elements d_{ij}, which specify the proportion of packets generated at node i whose destination is node j. The element c_{ij} of matrix* $\mathbf{C}$ *specifies to which outgoing link packets currently at node i should be directed if their final destination is node j.*

Packet routes are therefore determined statically and do not depend on the current state of the network.

Assume the following data:

$$\mathbf{D} = \begin{bmatrix} 0.00 & 0.20 & 0.10 & 0.40 & 0.30 \\ 0.20 & 0.00 & 0.10 & 0.25 & 0.45 \\ 0.15 & 0.15 & 0.00 & 0.30 & 0.40 \\ 0.20 & 0.10 & 0.60 & 0.00 & 0.10 \\ 0.25 & 0.15 & 0.50 & 0.10 & 0.00 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 1 & 1 & 10 \\ 2 & 0 & 3 & 12 & 12 \\ 4 & 4 & 0 & 5 & 5 \\ 7 & 7 & 6 & 0 & 7 \\ 9 & 11 & 11 & 8 & 0 \end{bmatrix}.$$

All packets are assumed to have the same size of 1000 bits. The transmission rates in the links between the nodes are: for links l_1, l_2, l_3, and l_4, 9600 *bits/s; for the remaining links,* 96000 *bits/s.*

Consequently, the service rates at the stations representing the links are

$$\mu_1 = \mu_2 = \mu_3 = \mu_4 = 9.6 \text{ packets/s},$$

$$\mu_5 = \cdots = \mu_{12} = 96 \text{ packets/s.}$$

The intensities of the external packet streams, expressed in packets per second, are

$$\lambda_{01} = 4, \qquad \lambda_{02} = 6.3, \qquad \lambda_{03} = 7.8, \qquad \lambda_{04} = 8.5, \qquad \lambda_{05} = 9.6.$$

The network configuration shown in Fig. 8.1 corresponds to the queueing-network model shown in Fig. 8.2. In addition to the five external arrival streams λ_{0i}, there are also output streams λ_{i0}, which represent packets leaving the network after reaching their destination.

Since each of the five input streams follows a specific route through the network, it is natural to interpret them as separate customer classes. For example, packets entering through stream λ_{01} form class 1 and propagate through the network as shown in Fig. 8.3a, while packets entering through stream λ_{05} form class 5, whose path is shown in Fig. 8.3b.

The virtual packet routes defined in this way mean that class 1 sees the network as a sequence of service stations as shown in Fig. 8.4a, while class 5 sees the network as the system shown in Fig. 8.4b.

The matrix **C** determines which stations belong to the same route, while the matrix **D** determines how traffic is distributed across the stations on that route. The non-zero transition probabilities $r_{ij}^{(1)}$ and $r_{ij}^{(5)}$ between stations i and j for the two classes mentioned above are shown in Fig. 8.4.

Since all packets have the same size, the analysis can be simplified by treating all packets as belonging to a single class and calculating transition probabilities r_{ij} common to all packets, while taking into account the relative proportions of the individual traffic streams in the aggregate flow.

The network fragments shown in Fig. 8.4 then take the form shown in Fig. 8.5, where the corresponding non-zero transition probabilities for the network elements are also given.

Table 8.1 contains a comparison of the simulation results, the BCMP Markov model (using calculations based on the *MVA* algorithm), and the diffusion approximation. The calculations were performed using the AMOK package.

In this case, the BCMP model is only approximate, since it requires the assumption of exponentially distributed transmission times, whereas in reality the transmission times are deterministic.

In the diffusion model, we assume

$$C_{Bi}^2 = 0, \qquad i = 1, \ldots, 12,$$

because the service time is constant and therefore its variance is zero. The service rates μ_i are given directly in the problem statement, while the traffic intensities λ_i are calculated from equation (7.23). The coefficients C_{Ai}^2 are determined from equations (7.50) and (7.52). These values are then used to calculate the diffusion parameters β_i and α_i for each station, which is subsequently treated as an independent $G/G/1$ system.

The table presents the average number of packets in each link. The corresponding transmission delay for a link can be obtained by multiplying the average number of packets by the constant transmission time of a single packet.

It can be seen that, under light traffic load, the diffusion model and the BCMP model give similar results. However, for the more heavily loaded links $l_1, \ldots, l_4$, and especially for the overloaded link l_3, the BCMP model overestimates the queue lengths. The assumption of exponential service times in the BCMP model generally leads to larger queue estimates than those observed in simulation.

In contrast, the diffusion approximation tends to underestimate the actual queue lengths in this example, although it remains closer to the simulation results than the BCMP model for the more heavily loaded links.

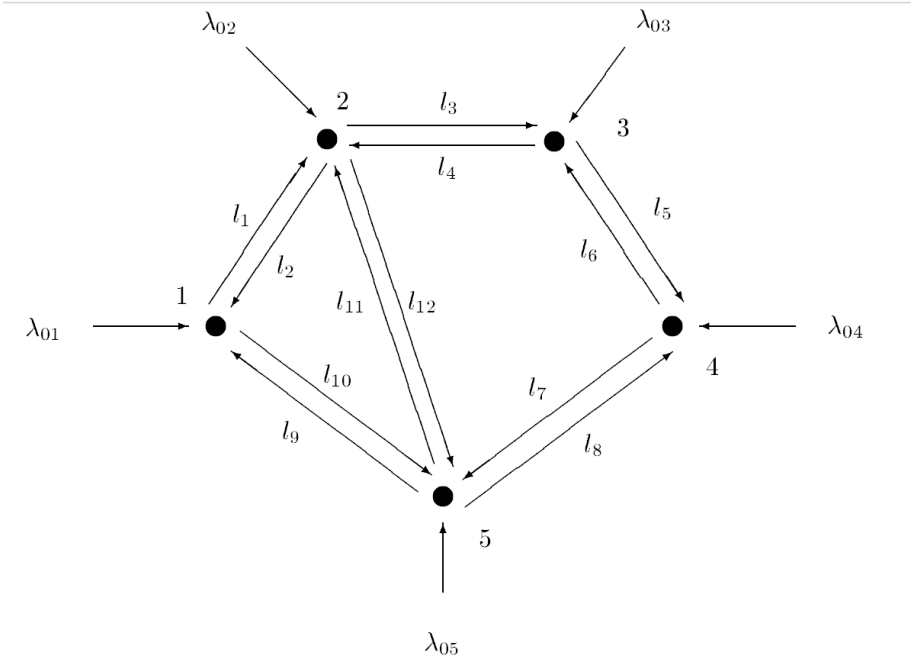


Figure 8.1: Configuration of the modelled network

■

8.2 Information flow control at the network input

The widespread use of devices transmitting information at variable bit rates (VBR devices), together with the use of statistical multiplexing, means that traffic-control mechanisms must ensure a reasonable compromise between link utilisation and quality of service. Quality of service, measured for example by the probability of packet loss and the variation in packet delay, naturally deteriorates as the network load increases.

Modern high-speed networks incorporate traffic-control functions that differ from the traditional window-based mechanisms (cf. Example 2.3). These functions may be classified as either preventive or corrective in nature [118, 117].

Preventive functions are based on checking, at the moment when a user requests a new connection and specifies the bandwidth required, how the additional traffic is likely to affect the performance of the network, and in particular what packet-loss probability will result from the increased load. If the predicted packet-loss probability remains below a prescribed threshold, the new connection is accepted.

Such admission-control mechanisms may be analysed using queueing-network models of the type $G/G/1/N$, where the effect of increased traffic intensity on the probability of packet loss can be evaluated.

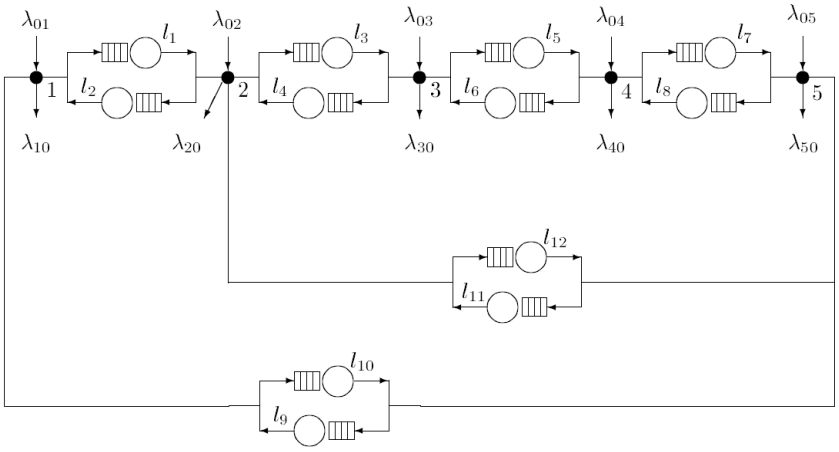


Figure 8.2: Queueing-network representation of the packet-switched network

Link	Simulation	BCMP	Diffusion	ρ_i
l_1	0.435	0.500	0.418	0.320
l_2	0.296	0.339	0.285	0.243
l_3	3.008	4.279	3.001	0.778
l_4	0.294	0.322	0.286	0.234
l_5	0.081	0.085	0.078	0.075
l_6	0.054	0.056	0.052	0.051
l_7	0.068	0.073	0.066	0.065
l_8	0.026	0.026	0.025	0.025
l_9	0.042	0.044	0.041	0.040
l_{10}	0.008	0.008	0.008	0.008
l_{11}	0.074	0.077	0.070	0.068
l_{12}	0.047	0.048	0.046	0.044

Table 8.1: Comparison of the average number of packets present in the links: simulation results, BCMP model results, and diffusion approximation results.

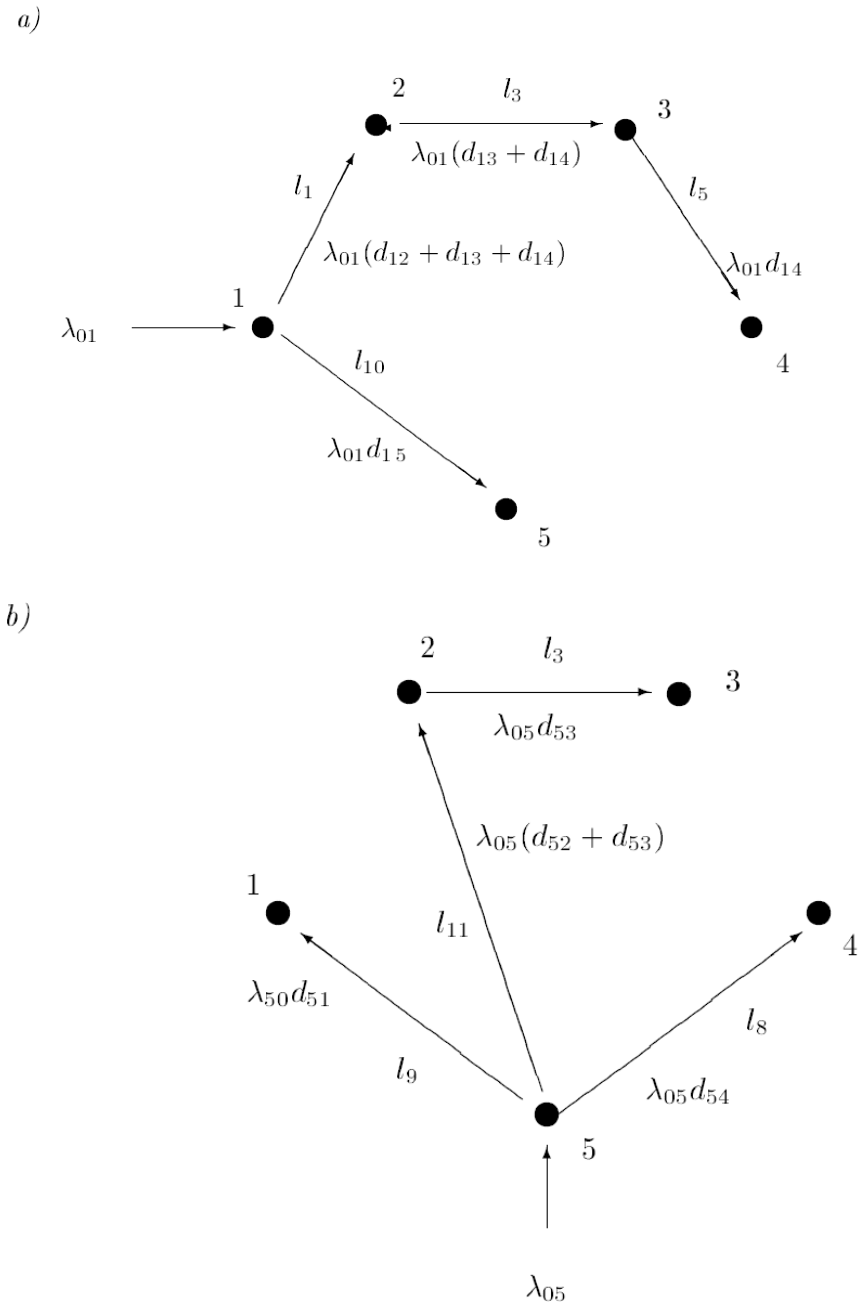
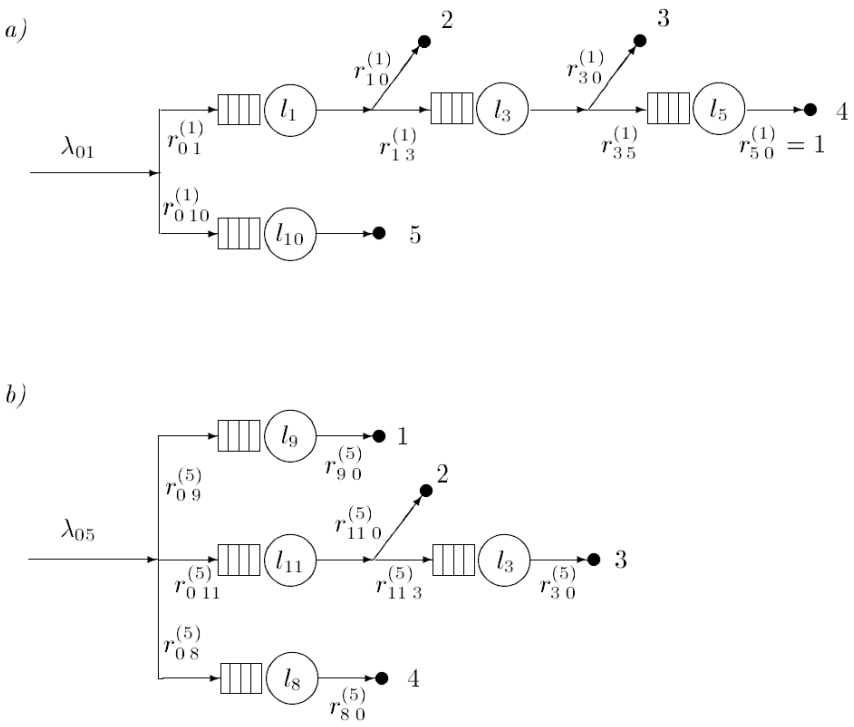


Figure 8.3: Routes followed by first-class and fifth-class packet flows



$$r^{(1)}_{0\,1} = d_{1\,2} + d_{1\,3} + d_{1\,4} \qquad r^{(5)}_{0\,9} = d_{5\,1}$$

$$r^{(1)}_{0\,10} = d_{1\,5} \qquad r^{(5)}_{0\,11} = d_{5\,2} + d_{5\,3}$$

$$r^{(1)}_{1\,3} = \frac{d_{1\,3} + d_{1\,4}}{d_{1\,2} + d_{1\,3} + d_{1\,4}} \qquad r^{(5)}_{11\,0} = \frac{d_{5\,2}}{d_{5\,2} + d_{5\,3}}$$

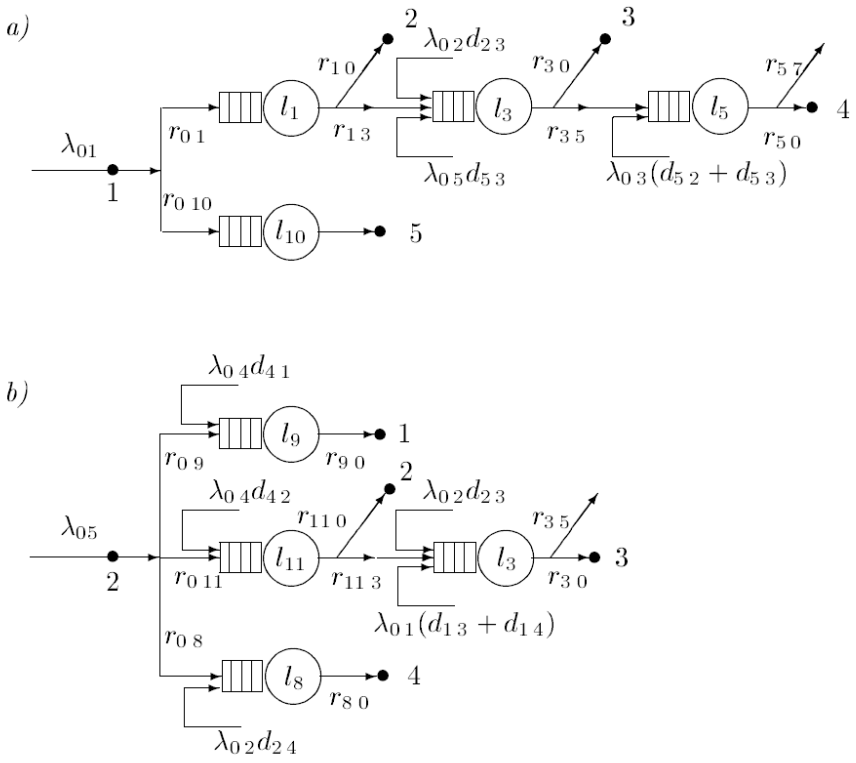
$$r^{(1)}_{1\,0} = \frac{d_{1\,2}}{d_{1\,2} + d_{1\,3} + d_{1\,4}} \qquad r^{(5)}_{0\,8} = d_{5\,4}$$

$$r^{(1)}_{3\,0} = \frac{d_{1\,3}}{d_{1\,3} + d_{1\,4}} \qquad r^{(5)}_{11\,3} = \frac{d_{5\,3}}{d_{5\,2} + d_{5\,3}}$$

$$r^{(1)}_{3\,5} = \frac{d_{1\,4}}{d_{1\,3} + d_{1\,4}} \qquad r^{(5)}_{9\,0} = r^{(5)}_{3\,0} = r^{(5)}_{8\,0} = 1$$

$$r^{(1)}_{5\,0} = r^{(1)}_{10\,0} = 1$$

Figure 8.4: Paths followed by first-class and fifth-class packets



$$r_{01} = d_{12} + d_{13} + d_{14}$$

$$r_{09} = d_{51}$$

$$r_{010} = d_{15}$$

$$r_{011} = d_{52} + d_{53}$$

$$r_{13} = \frac{d_{13} + d_{14}}{d_{12} + d_{13} + d_{14}}$$

$$r_{110} = \frac{\lambda_{05} d_{52} + \lambda_{04} d_{42}}{\lambda_{05}(d_{52} + d_{53}) + \lambda_{04} d_{42}}$$

$$r_{10} = \frac{d_{12}}{d_{12} + d_{13} + d_{14}}$$

$$r_{08} = d_{54}$$

$$r_{30} = \frac{\lambda_{01} d_{13} + \lambda_{02} d_{23} + \lambda_{05} d_{53}}{\lambda_{01} d_{13} + \lambda_{02} d_{23} + \lambda_{05} d_{53} + \lambda_{01} d_{14}}$$

$$r_{113} = \frac{\lambda_{05} d_{53}}{\lambda_{05}(d_{52} + d_{53}) + \lambda_{04} d_{42}}$$

$$r_{35} = \frac{\lambda_{01} d_{14}}{\lambda_{01}(d_{13} + d_{14}) + \lambda_{02} d_{23} + \lambda_{05} d_{53}}$$

$$r_{90} = r_{80} = r_{100} = 1$$

$$r_{50} = \frac{\lambda_{01} d_{14} + \lambda_{03} d_{34}}{\lambda_{01} d_{14} + \lambda_{03}(d_{34} + d_{35})}$$

$$r_{57} = \frac{\lambda_{03} d_{35}}{\lambda_{01} d_{14} + \lambda_{03}(d_{34} + d_{35})}$$

Figure 8.5: Transition probabilities in the aggregated single-class model

Preventive traffic-control functions also involve continuously checking whether users comply with the conditions of the traffic contract, that is, whether they transmit no more data than the allocated bandwidth permits. This applies not only to average traffic rates, but also to short-term instantaneous peaks.

8.2.1 The *leaky bucket* mechanism

The *leaky bucket* mechanism, see for example [68], limits excessive deviations of the traffic stream from the characteristics specified in the contract between the user and the network. This reduces the probability of congestion and therefore improves quality of service.

Packets (cells) generated by the user are allowed to enter the network only if they can acquire a token. Tokens are generated by the network at regular time intervals. The token-generation rate corresponds to the bandwidth allocated to the user.

Some irregularity in the packet stream is permitted. A packet buffer is provided, which fills when packets arrive too quickly, and a token buffer is provided, which fills when packets arrive less frequently than assumed in the traffic contract. Both buffers have finite capacity, and their sizes determine the admissible deviations of the traffic stream from the nominal rate; see Fig. 8.6.

When the packet buffer is full, newly arriving packets are discarded. When the token buffer is full, newly generated tokens are lost. It is therefore important to determine the effect of the leaky-bucket mechanism on the resulting packet stream. Existing models of the leaky bucket are based on discrete-time Markov chains [48] and fluid approximations [29, 92]. Here we use a diffusion approximation.

Let B denote the capacity of the packet buffer, measured in packets, and let M denote the capacity of the token buffer. The current numbers of packets and tokens are denoted by b and m, respectively. The diffusion process $X(t)$ is defined on the interval

$$x \in [0, N], \qquad N = B + M,$$

with state variable

$$x = b - m + M.$$

Let packet arrivals occur according to a process with mean interarrival time $1/\lambda_c$ and squared coefficient of variation C_{Ac}^2. Tokens are generated periodically every $1/\lambda_t$ units of time, so that

$$C_{At}^2 = 0.$$

Each arriving packet increases the value of the diffusion process, whereas each arriving token decreases it. Accordingly, the diffusion parameters are chosen as

$$\beta = \lambda_c - \lambda_t, \qquad \alpha = \lambda_c C_{Ac}^2.$$

The process evolves between two barriers located at $x = 0$ and $x = M + B$. The state $x = 0$ corresponds to a full token buffer and an empty packet buffer, so that newly generated tokens are lost. The state $x = M + B$ corresponds to an empty token buffer and a full packet buffer, so that newly arriving packets are discarded.

The residence time at the upper barrier $x = M + B$ corresponds to the waiting time until the next token arrival, that is, the time between the moment when the packet buffer becomes

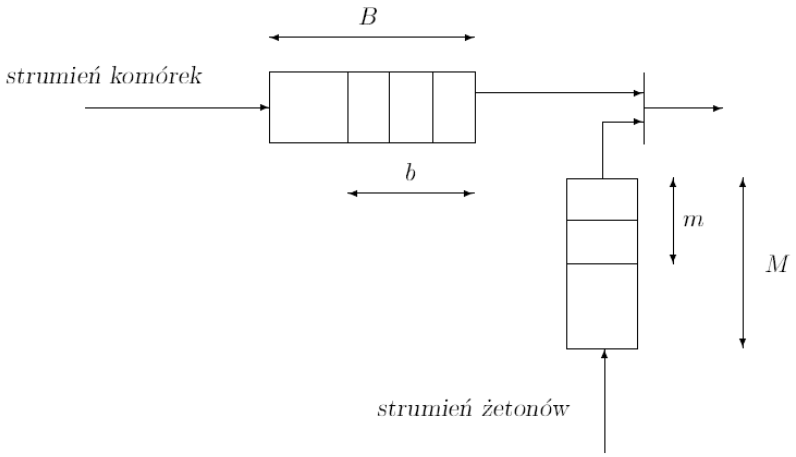


Figure 8.6: Structure of the leaky bucket mechanism

full and the arrival of the next token. We use the completion-time distribution derived in [39] for the $M/D/1/N$ model and assume that the density function $l_{M+B}(x)$ has the form

$$l_{M+B}(x) = \frac{\lambda_c e^{-\lambda_c x}}{1 - e^{-\lambda_c/\lambda_t}}. \tag{8.1}$$

The packet-loss probability $L(t)$ is bounded by [39]

$$L(t) \leq p_N(t)\, P\left[Arr\left(t, t + \frac{1}{\lambda_t}\right) \geq 1\right],$$

where $Arr(t, t + 1/\lambda_t)$ denotes the number of arriving packets in the interval

$$\left[t, t + \frac{1}{\lambda_t}\right].$$

If the arrival process is Poisson, then the density function $l_0(x)$ of the residence time at the lower barrier $x = 0$ is simply the density of the interarrival-time distribution. For more general arrival processes, we use this density as an approximation.

States with $x > M$ correspond to situations in which packets are waiting for tokens. In that case, $x - M$ gives the number of packets waiting in the packet buffer. States with $x < M$ correspond to situations in which tokens are waiting for arriving packets; in that case, $M - x$ gives the number of available tokens.

The probability of having b packets in the packet buffer at time t is given by

$$f(M + b, t),$$

while the probability that the packet buffer is empty is

$$p_b(t) = p_0(t) + \int_0^M f(x, t)\, dx.$$

Similarly, the probability of having m tokens in the token buffer is given by

$$f(M - m, t),$$

and the probability that the token buffer is empty is

$$\int_{M}^{M+B} f(x,t)\,dx + p_N(t),$$

where

$$p_0(t) = P[X(t) = 0], \qquad p_N(t) = P[X(t) = N].$$

Since the token-generation process is deterministic, the waiting-time density of packets for tokens (the response time of the leaky bucket) can be represented as

$$r(x,t) = \lambda_t f(\lambda_t x + M, t).$$

The density function $f(x,t)$ is obtained from the $G/G/1/N$ diffusion model, using equations (7.41) and (7.42), followed by numerical inversion of the Laplace transform $\bar{f}(x,s)$.

In this way, one can determine:

- the distribution of the number of tokens in the token buffer,
- the distribution of the number of packets waiting for tokens,
- the waiting-time distribution of packets,
- the probability of packet loss,
- the probability of token loss.

The model also allows the special cases $B = 0$ or $M = 0$, which correspond to simplified variants of the leaky-bucket mechanism.

If there are tokens available, which occurs with probability $p_t(t)$, the output process of the cells is identical to the cell arrival process. If cells have to wait for tokens, which occurs with probability $1 - p_t(t)$, the output process of the cells follows the token-arrival process.

At time t, the density function $d(x,t)$ of the intervals between cell departures is therefore given by

$$d(x,t) = p_t(t)\,a(x,t) + [1 - p_t(t)]\,\delta\left(x - \frac{1}{\lambda_t}\right), \tag{8.2}$$

where $a(x,t)$ is the time-dependent density function of the interarrival times in the input stream.

From equation (8.2) one can calculate the mean value and the coefficient of variation of the output process. These parameters are sufficient to include a *leaky bucket* station in a network of $G/G/1/N$ queues representing a computer network, where the output of the leaky bucket serves as the input flow to the network. The model can also be used to describe a cascade of leaky-bucket nodes with different parameter settings.

Example 8.2

Let us assume that at time $t = 0$ the packet buffer is empty, while the token buffer initially contains $M(0)$ tokens. Tokens are generated regularly every unit of time. The cell-arrival process is Poisson.

During the interval $0 \le t < 100$, the mean interarrival time of cells is 0.5 time units. For $t \ge 100$, the mean interarrival time increases to 1.5 time units. Thus, during the first 100 time units, the traffic intensity exceeds the limit imposed by the token rate, whereas afterwards the traffic intensity falls below that limit. The buffer capacities are $B = M = 100$.

Figure 8.7 shows the results of the diffusion approximation and the simulation for the output flow from the leaky bucket, for initial token-buffer occupancies $M(0) = 0$, 50, and 100. The diffusion approximation reproduces the simulation results very closely. The simulation results were obtained by averaging over 100 000 independent runs.

If the token buffer is initially empty, the output rate is almost immediately limited to one cell per unit of time, and excess incoming cells are accumulated in the packet buffer. The evolution of the mean number of cells in the packet buffer is shown in Fig. 8.8.

After the period of excessive traffic, that is for $t > 100$, the cells accumulated in the packet buffer are transmitted as tokens become available. Consequently, the output intensity remains equal to one cell per unit time for some time after $t = 100$, and only later decreases to the level of the input-stream intensity.

If the token buffer is initially half full, $M(0) = 50$, part of the burst of arriving cells passes through the leaky bucket without delay, and the number of buffered cells is smaller. This effect is even more pronounced when the token buffer is initially full, $M(0) = 100$: in this case, almost the entire burst of cells passes through the leaky bucket without restriction.

Figure 8.9 shows, for the case $M(0) = 0$, the distribution of the number of cells in the packet buffer at the end of the high-traffic period, that is at $t = 100$, when the packet buffer is most heavily loaded. Although the average number of cells in the buffer at that time is lower than the buffer capacity (see Fig. **??**), the probability of cell loss is still high, approximately 30%, which is correctly predicted by the diffusion model.

Figures 8.10, 8.12, and **??** show the distribution of the number of cells in the packet buffer at different time instants during periods of high and low traffic intensity. The results of the diffusion approximation are compared with the corresponding simulation results. ■

8.2.2 Leaky bucket mechanism with two types of tokens

One possible extension of the leaky bucket mechanism introduces two types of tokens: green and red, Fig. 8.13. Green tokens are generated at a rate corresponding to the bandwidth allocated to the user. Red tokens allow an additional number of cells to be admitted, but with reduced priority.

The generation of red tokens begins when the number of cells waiting for tokens exceeds a specified threshold B_1, and stops when the number of waiting cells drops below this level.

We denote the parameters of the cell stream by λ_c and C^2_{Ac}, the parameters of the green-token stream by λ_{gt} and C^2_{Agt}, and the parameters of the red-token stream by λ_{rt} and C^2_{Art}.

The diffusion process is defined on the interval

$$x \in [0, M + B],$$

where B is the capacity of the cell buffer and M is the capacity of the green-token buffer. The instantaneous value of the process is defined as

$$x = b - m + M,$$

where b and m denote, respectively, the current contents of the cell buffer and the token buffer.

If $x < M$, then $M - x$ green tokens are waiting for cells. If $x > M$, then $x - M$ cells are waiting for tokens; see Fig. 8.14.

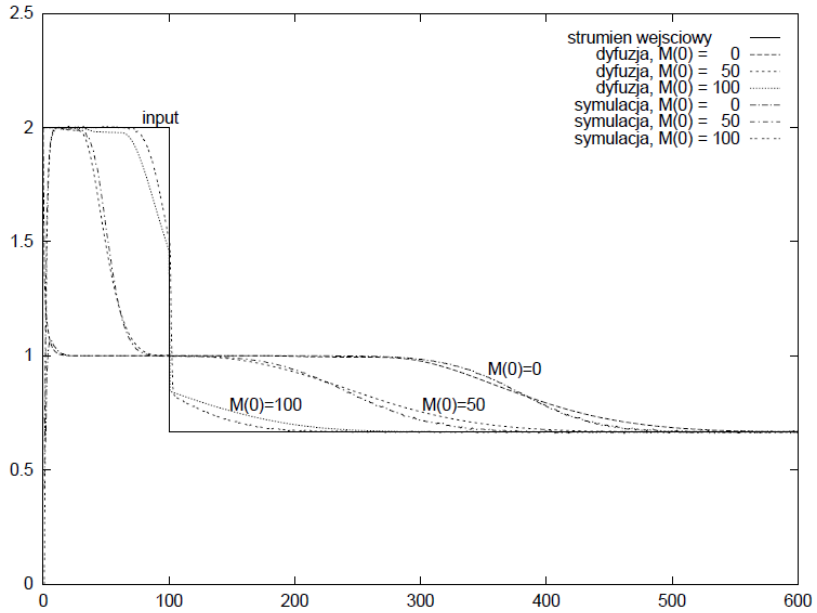


Figure 8.7: Input and output intensities in the leaky bucket as functions of time, for initial token-buffer occupancies $M(0) = 0$, 50, and 100; comparison of diffusion approximation and simulation results

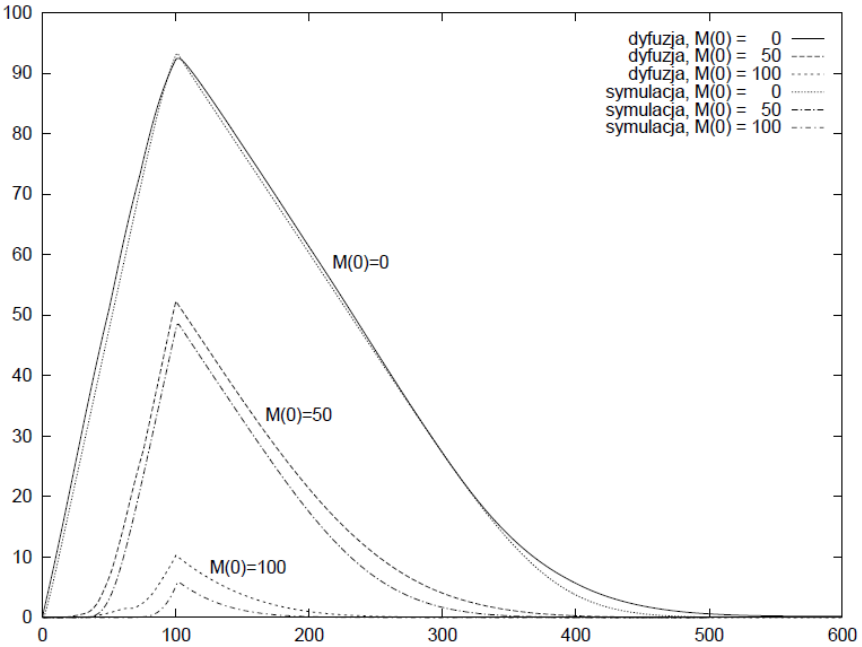


Figure 8.8: Mean number of cells in the packet buffer as a function of time, for $M(0) = 0$, 50, and 100; comparison of diffusion approximation and simulation results

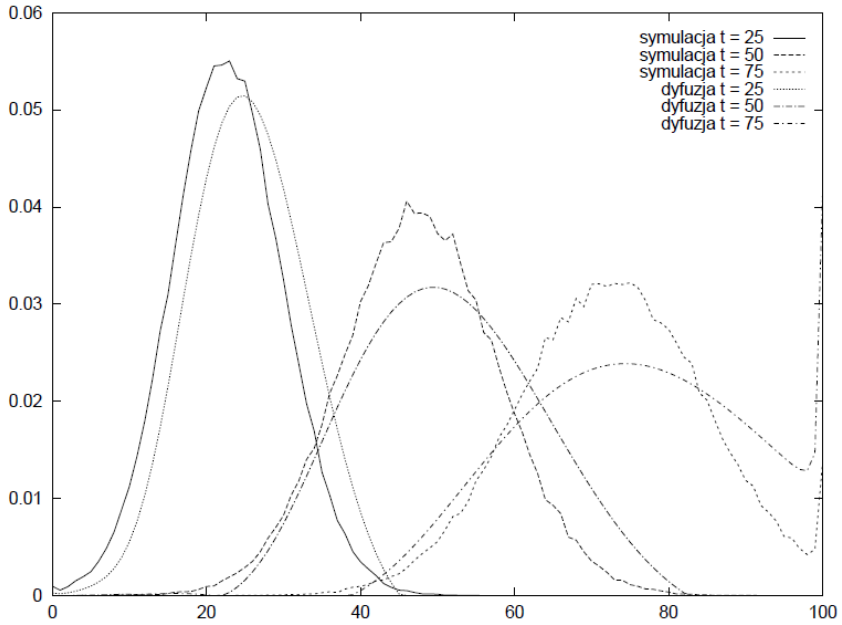


Figure 8.9: Distribution of the number of cells in the packet buffer during the high-traffic period, for $t = 25, 50, 75$, $M(0) = 0$; comparison of diffusion approximation and simulation results

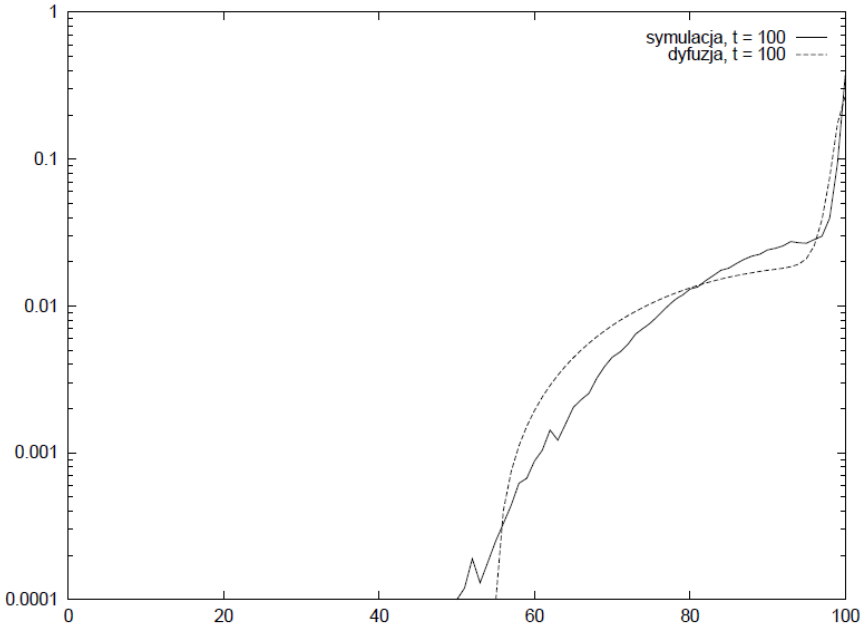


Figure 8.10: Distribution of the number of cells in the packet buffer at the end of the high-traffic period, for $t = 100$, $M(0) = 0$; comparison of diffusion approximation and simulation results

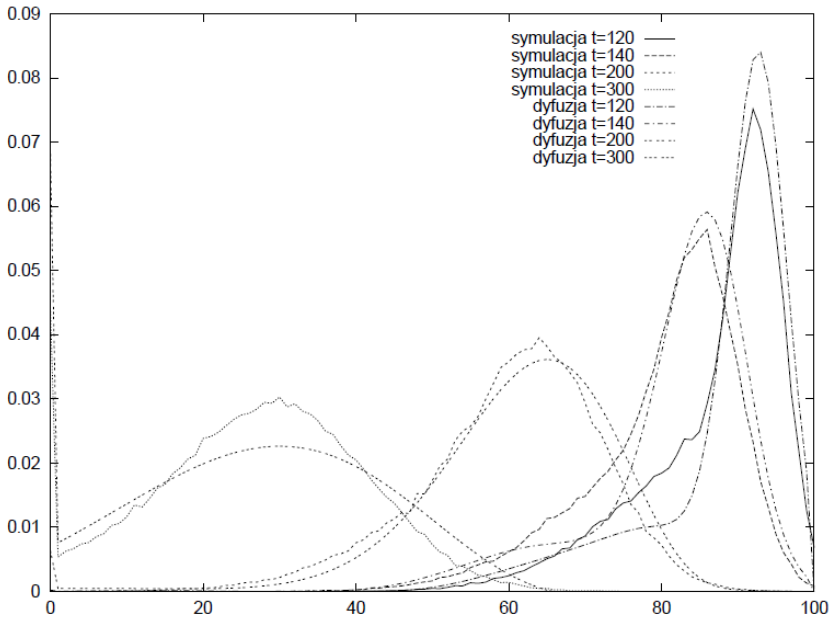


Figure 8.11: Distribution of the number of cells in the packet buffer during the first part of the low-traffic period, for $t = 120, 140, 200, 300$, $M(0) = 0$; comparison of diffusion approximation and simulation results

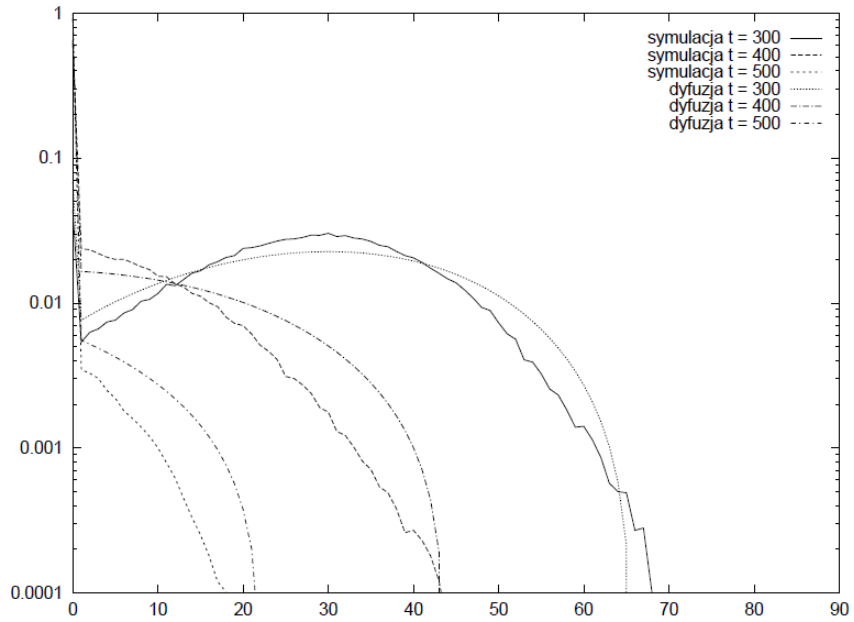


Figure 8.12: Distribution of the number of cells in the packet buffer during the later part of the low-traffic period, for $t = 300, 400, 500$, $M(0) = 0$; comparison of diffusion approximation and simulation results

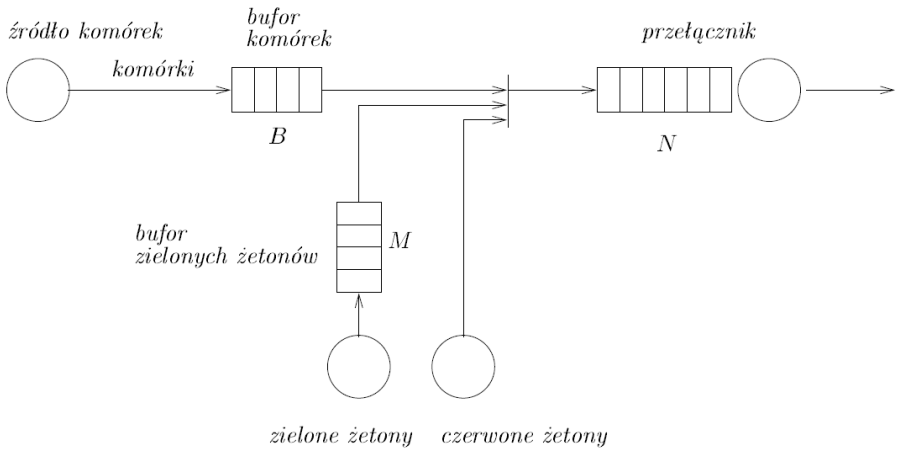


Figure 8.13: Leaky bucket mechanism with two types of tokens

If $x > M + B_1$, then red tokens are generated in addition to green tokens. The model therefore has two subregions with different diffusion parameters.

For

$$0 < x \leq M + B_1,$$

we have

$$\beta_1 = \lambda_c - \lambda_{gt}, \qquad \alpha_1 = \lambda_c C_{Ac}^2 + \lambda_{gt} C_{Agt}^2.$$

For

$$M + B_1 \leq x < M + B,$$

we have

$$\beta_2 = \lambda_c - \lambda_{gt} - \lambda_{rt},$$

$$\alpha_2 = \lambda_c C_{Ac}^2 + \lambda_{gt} C_{Agt}^2 + \lambda_{rt} C_{Art}^2.$$

We assume that both types of tokens are generated at constant time intervals, so

$$C_{Agt}^2 = C_{Art}^2 = 0.$$

The solution consists of two diffusion processes, $X_1(t)$ and $X_2(t)$, defined on the intervals

$$x \in (0, M + B_1) \qquad \text{and} \qquad x \in (M + B_1, M + B),$$

respectively, and communicating through a barrier located at

$$x = M + B_1.$$

At the points $x = 0$ and $x = M + B$, barriers are placed as in the classical diffusion model. These barriers retain the process for some time and then reintroduce it at $x = 1$ and $x = M + B - 1$, respectively.

The process $X_1(t)$, when absorbed at the barrier $x = M + B_1$, is immediately reinitiated at the point $x = M + B_1 + \varepsilon$ as process $X_2(t)$. Conversely, process $X_2(t)$, when absorbed at the same barrier, is immediately reinitiated at the point $x = M + B_1 - \varepsilon$ as process $X_1(t)$.

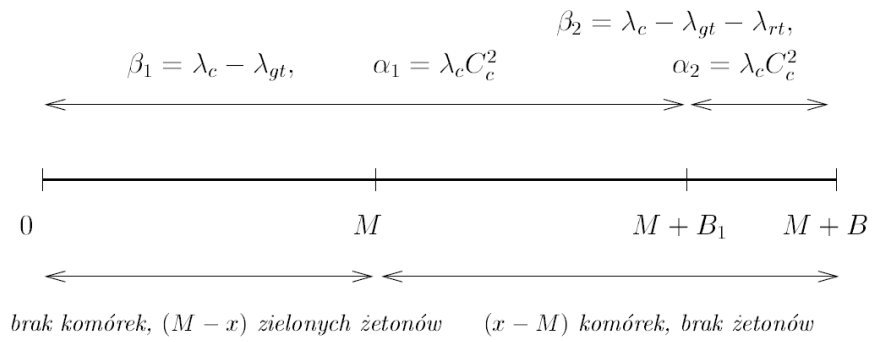


Figure 8.14: Leaky bucket mechanism with two types of tokens: selection of diffusion-model parameters

The density functions $f_1(x,t;\psi_1)$ and $f_2(x,t;\psi_2)$ of the two processes are expressed in terms of the density functions $\phi_1(x,t;x_0)$ and $\phi_2(x,t;x_0)$ of processes with two absorbing barriers: at $x = 0$ and $x = M + B_1$ for the first process, and at $x = M + B_1$ and $x = M + B$ for the second process:

$$\begin{aligned} f_1(x,t;\psi_1) &= \phi_1(x,t;\psi_1) + \int_0^t g_1(\tau)\phi_1(x,t-\tau;1)\,d\tau \\ &\quad + \int_0^t g_{M+B_1-\varepsilon}(\tau)\phi_1(x,t-\tau;M+B_1-\varepsilon)\,d\tau, \qquad (8.3) \\ f_2(x,t;\psi_2) &= \phi_2(x,t;\psi_2) + \int_0^t g_{M+B-1}(\tau)\phi_2(x,t-\tau;M+B-1)\,d\tau \\ &\quad + \int_0^t g_{M+B_1+\varepsilon}(\tau)\phi_2(x,t-\tau;M+B_1+\varepsilon)\,d\tau. \qquad (8.4) \end{aligned}$$

The functions ϕ_1 and ϕ_2 are known; see equation (7.35). The reinitialisation processes

$$g_1(t), \quad g_{M+B_1-\varepsilon}(t), \quad g_{M+B_1+\varepsilon}(t), \quad g_{M+B-1}(t)$$

are determined from equations (7.38) and (7.40), extended by the balance equations describing the probability flows between the two processes.

As in the $G/G/1$ and $G/G/1/N$ models, the practical solution is obtained by transforming the equations into the Laplace domain and then numerically recovering the original functions $f_1(x,t;\psi_1)$ and $f_2(x,t;\psi_2)$.

The probability of cell loss is estimated as

$$p_{M+B}(t).$$

The output stream of higher-priority cells, i.e. cells that have received a green token, has density

$$\begin{aligned} d^{(1)}(x,t) &= a(x,t)\,P[X(t) \le M] + d_{gt}(x)\,P[M \le X(t) \le M+B] \\ &= a(x,t)\left(p_0(t) + \int_0^M f_1(x,t)\,dx\right) + d_{gt}(x)\int_M^{M+B} f_1(x,t)\,dx, \qquad (8.5) \end{aligned}$$

from which we determine the parameters $\lambda^{(1)}$ and $C_A^{(1)^2}$ of the higher-priority flow at the switch input.

The intensity of lower-priority cells, i.e. cells that have received a red token, is

$$\lambda^{(2)}(t) = \lambda_{rt}\, P[X(t) \geq M + B_1] = \lambda_{rt} \left(p_{M+B}(t) + \int_{M+B_1}^{M+B} f_2(x,t)\, dx \right).$$

The output flow of lower-priority cells leaving the leaky bucket is similar in nature to the traffic generated by an *ON–OFF* source. During periods when the process $X(t)$ continuously satisfies

$$X(t) > M + B_1,$$

cells with reduced priority leave the leaky bucket; this corresponds to the *ON* period. The *OFF* period begins when

$$X(t) < M + B_1,$$

that is, when the generation of red tokens is interrupted and the priority of the cells is no longer reduced.

To estimate the distribution of the durations of these periods, we use the density $\gamma_{x_0,a}(x)$ of the first-passage time from the starting point x_0 to the absorbing barrier at $x = a$.

Thus, the density of the duration of the *ON* period can be written as

$$\begin{aligned} f_{ON}(x) \;=\; & P_1\, \gamma_{M+B_1+1,\, M+B_1}(x) \\ & +(1-P_1)P_2\, \gamma_{M+B_1+1,\, M+B}(x) * \delta(x-1) * \gamma_{M+B-1,\, M+B_1}(x) \\ & +(1-P_1)(1-P_2)P_2\, \gamma_{M+B_1+1,\, M+B}(x) * \delta(x-1) * \gamma_{M+B-1,\, M+B}(x) \\ & *\delta(x-1) * \gamma_{M+B-1,\, M+B_1}(x) + \cdots \end{aligned}$$

where P_1 is the probability that the process starting at

$$x = M + B_1 + 1$$

first reaches

$$x = M + B_1,$$

that is,

$$P_1 = \int_0^\infty \gamma_{M+B_1+1,\, M+B_1}(x)\, dx,$$

and P_2 is the probability that the process starting at

$$x = M + B - 1$$

first reaches

$$x = M + B_1,$$

that is,

$$P_2 = \int_0^\infty \gamma_{M+B-1,\, M+B_1}(x)\, dx.$$

Here, $\delta(x-1)$ is the density of the time spent at the barrier located at $x = M + B$.

Applying the Laplace transform, we obtain

$$\bar{f}_{ON}(s) = P_1\,\bar{\gamma}_{M+B_1+1,\,M+B_1}(s) + \frac{(1-P_1)P_2\,\bar{\gamma}_{M+B_1+1,\,M+B}(s)\,\bar{\gamma}_{M+B-1,\,M+B_1}(s)\,e^{-s}}{1-(1-P_2)\bar{\gamma}_{M+B-1,\,M+B}(s)e^{-s}}.$$

In a similar manner, we determine the probability density function $f_{OFF}(x)$ of the duration of the *OFF* period for red tokens, which allows us to determine the value of

$$E[A_i^2]$$

in equation (8.49).

8.2.3 The *jumping window* mechanism

The *jumping window* mechanism enforces compliance with the traffic parameters allocated to a user in the following way: it checks whether the user does not exceed the assigned bandwidth, the so-called *CIR* (*Committed Information Rate*), denoted by λ. The purpose of this mechanism is also to ensure that, while maintaining the prescribed average rate, the instantaneous traffic rate does not exceed a certain maximum value.

Time is divided into intervals of length T_c. According to the traffic contract, the amount of information that the source may transmit during one such interval is

$$B_c = CIR \cdot T_c.$$

Incoming packet traffic is monitored, and as long as the amount of information transmitted during the current interval T_c does not exceed B_c, the packets are assigned high priority.

Once the threshold B_c has been exceeded, subsequent packets are still admitted to the network, but they are assigned lower priority and, in the event of congestion, they are discarded first, in accordance with the queueing discipline used at the link. Thus, packet priority may result not only from the content of the packet and the corresponding quality of service guaranteed by the network, but also from the distinction between packets transmitted within the bandwidth allocated by contract and excess packets transmitted beyond that limit.

The packet priority is recorded in its control field, the so-called header, by means of a single bit, the *DE* bit (*discard eligibility bit*); a value $DE = 1$ indicates the lowest priority.

The amount of excess traffic is limited to B_e. Once the threshold $B_c + B_e$ is exceeded, packets from the given source are discarded until the end of the current interval T_c. At the start of the next interval T_c, packets again regain high priority.

Let us assume that the packet stream entering the network is described by the intensity λ and the squared coefficient of variation of interarrival times C_A^2.

As a result of the *jumping window* policy described above, two packet streams are created: a higher-priority stream (class 1) and a lower-priority stream (class 2).

Within an interval T_c, we distinguish three periods: T_1, T_2, and T_3. Their durations are random and depend on the parameters of the input stream.

During T_1, the contractual amount of information for the interval T_c, namely B_c, is transmitted; cells or packets leaving the control device during this period have high priority.

During T_2, packets corresponding to the conditionally admissible excess traffic (between B_c and $B_c + B_e$) are transmitted and receive lower priority.

After the level $B_c + B_e$ is reached, the period T_3 begins, during which incoming packets are discarded; this period lasts until the end of the interval T_c.

In the queueing model of the station representing the control mechanism, customers of a single class enter in a stream of intensity λ, whereas two output streams leave the station: the first-class stream, representing higher-priority packets, and the second-class stream, representing conditionally admitted packets.

Let $\lambda^{(1)}$ denote the intensity of the first-class stream. It is the average intensity of a stream that has value λ during period T_1, and value zero during periods T_2 and T_3.

Similarly, the output stream of second-class packets has intensity λ during period T_2 and zero during the remaining periods, which gives an average value denoted by $\lambda^{(2)}$.

The periods T_1, T_2, and T_3 are random variables. Let $h_1(x)$, $h_2(x)$, and $h_3(x)$ denote the probability-density functions of their distributions.

In the diffusion approximation, the density $h_1(x)$ can be approximated by the first-passage-time distribution of a diffusion process between $x = 0$ and $x = B_c$. Similarly, $h_2(x)$ is approximated by the distribution of the first-passage time between $x = B_c$ and $x = B_c + B_e$.

Of course, the second period occurs only if the preceding period T_1 is shorter than T_c.

When determining the first-passage times, we are interested in the number of packets arriving at the station, so we choose the diffusion parameters

$$\beta = \lambda, \qquad \alpha = \sigma_A^2 \lambda^3 = C_A^2 \lambda.$$

The time-varying cumulative number of arrived packets is then described as a diffusion process starting at time $t = 0$ from the point $x_0 = 0$ and bounded by an absorbing barrier at $x = B_c$: once the process reaches $x = B_c$, it remains there permanently.

Since the process has a constant positive drift ($\beta > 0$), for simplicity we neglect the boundary condition at $x = 0$.

The probability density function of such a process is [18]

$$p(x,t) = \frac{1}{\sqrt{2\alpha\Pi t}} \left[\exp\left\{ -\frac{(x-\beta t)^2}{2\alpha t} \right\} - \exp\left\{ \frac{2\beta B_c}{\alpha} - \frac{(x - 2B_c - \beta t)^2}{2\alpha t} \right\} \right]. \tag{8.6}$$

The probability density function of the first-passage time from $x_0 = 0$ to the barrier at $x = B_c$ is

$$\begin{aligned} h_1(t) &= -\frac{d}{dt} \int_{-\infty}^{B_c} p(x,t)\, dx \\ &= -\frac{d}{dt} \left\{ \Phi\left(\frac{B_c - \beta t}{\sqrt{\alpha t}} \right) - \exp\left(\frac{2\beta B_c}{\alpha} \right) \Phi\left(\frac{-B_c - \beta t}{\sqrt{\alpha t}} \right) \right\} \\ &= \frac{B_c}{\sqrt{2\alpha\Pi t^3}} \exp\left[-\frac{(B_c - \beta t)^2}{2\alpha t} \right]. \end{aligned} \tag{8.7}$$

Its Laplace transform is

$$\bar{h}_1(s) = \exp\left[\frac{B_c}{\alpha} \left(\beta - \sqrt{\beta^2 + 2\alpha s} \right) \right] = \exp\left[\frac{B_c}{\alpha} \left(\beta - A(s) \right) \right], \tag{8.8}$$

where

$$A(s) = \sqrt{\beta^2 + 2\alpha s}.$$

The density function $h_2(t)$ of the second period is obtained in the same way, by considering the first-passage time from $x = B_c$ to $x = B_c + B_e$:

$$h_2(t) = \frac{B_e}{\sqrt{2\alpha\Pi t^3}} \exp\left[-\frac{(B_e - \beta t)^2}{2\alpha t}\right]. \tag{8.9}$$

The moments of order n for the durations T_1 and T_2 are calculated as

$$E[T_1^n] = \int_0^{T_c} h_1(x)x^n\,dx + \left[1 - \int_0^{T_c} h_1(x)\,dx\right] T_c^n, \tag{8.10}$$

$$E[T_2^n] = \int_0^{T_c} h_1(x)\left\{\int_0^{T_c-x} \xi^n h_2(\xi)\,d\xi + \left[1 - \int_0^{T_c-x} h_2(\xi)\,d\xi\right](T_c - x)^n\right\} dx. \tag{8.11}$$

Since the sum of the three periods is equal to T_c, we have

$$T_1 + T_2 + T_3 = T_c,$$

which implies

$$h_1(x) * h_2(x) * h_3(x) = \delta(x - T_c),$$

and therefore

$$\bar{h}_3(s) = \frac{e^{-T_c s}}{\bar{h}_1(s)\bar{h}_2(s)} = \exp\left[-T_c s - \frac{B_c + B_e}{\alpha}(\beta - A(s))\right].$$

The average intensities of the higher-priority and lower-priority packet streams leaving the station are therefore

$$\lambda^{(1)} = \frac{E[T_1]}{T_c}\lambda, \qquad \lambda^{(2)} = \frac{E[T_2]}{T_c}\lambda.$$

The squared coefficients of variation of the interdeparture times for both output streams may be approximated by

$$C_A^{(1)^2} \approx C_A^2\left(\frac{E[T_1]}{T_c}\right)^2, \qquad C_A^{(2)^2} \approx C_A^2\left(\frac{E[T_2]}{T_c}\right)^2. \tag{8.12}$$

Example 8.3

The system consists of an ON-OFF source, whose active and inactive periods follow an exponential distribution with mean durations of 40 and 120 time units, respectively. The data rate during the ON period is 1 packet per unit of time, so the average data rate is 0.25 packets per unit of time. The network guarantees this data rate to the user. Traffic is controlled by a sliding window mechanism. Let $T_c = 250$ *time units,* $B_c = \lambda_g T_c$*, and we choose* $B_e = 160$.

Figure 8.15 shows the average number of packets as a function of time, for a station having input stream controlled by the sliding window; these are simulation results. It can be seen that, that towards the end of the control period, the number of packets of the first class drops sharply, whilst the number of packets of the second class rises (the B_c threshold has been exceeded). Fig. 8.16 shows a similar graph for both classes of clients combined, comparing the results of the diffusion approximation and the simulation.

Example 8.4

Let us assume that the buffer length is limited to $N = 40$*, and the source is of* $ON = OFF$

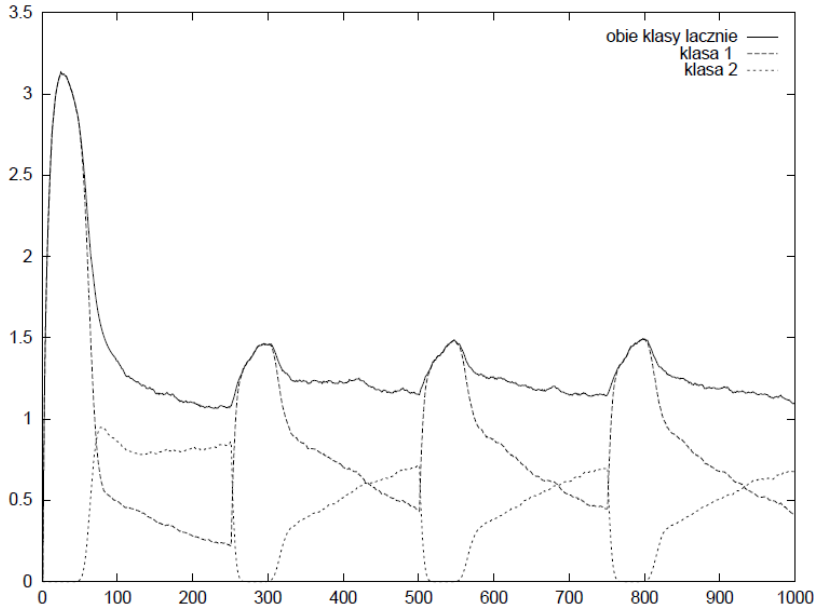


Figure 8.15: Mean value of packets in the queue controlled by jumping window mechanism (first and second class separately and together, simulation 10 000 repetitions)

type with the rate of $3\ Mbits/s$ *the multiplexer's service rate is* $2\ Mbits/s$*. The length of the transmitted packets follows an exponential distribution with a mean of 12 Kbits, so the average packet transmission time is 6 ms. We choose a time unit of this size, hence* $\lambda_{ON} = 1.5$*, and for the multiplexer* $\mu = 1$*. Let us assume that the ON periods have an average duration of 40 time units, and the OFF periods last an average of 120 time units, so* $\lambda_{eff} = 0.375$*.*

Figs. 8.17, 8.18 display the multiplexer's queue in steady state, illustrating the impact of the mechanism parameters.

■

8.2.4 The *sliding window* mechanism

The *sliding window* mechanism is more restrictive than the *jumping window* mechanism: it requires that, at every instant t, the number of packets arriving during the interval $[t - T_c, t]$ does not exceed a specified threshold.

As before, we assume that after the threshold B_c has been exceeded, an additional number B_e of packets may still be admitted during the interval T_c, but with reduced priority. When the total number of packets transmitted during this period exceeds $B_c + B_e$, all subsequent packets are discarded.

The operation of the mechanism can be represented by the queueing model shown in Fig. 8.19. The first station is of type $G/D/B_c/B_c$: it has B_c parallel service channels, no waiting room, and a constant service time equal to T_c. This station behaves similarly to the sliding-window mechanism: if, during the interval T_c preceding the arrival of a new packet, there have already been B_c accepted packets, all service channels are occupied. Since there is

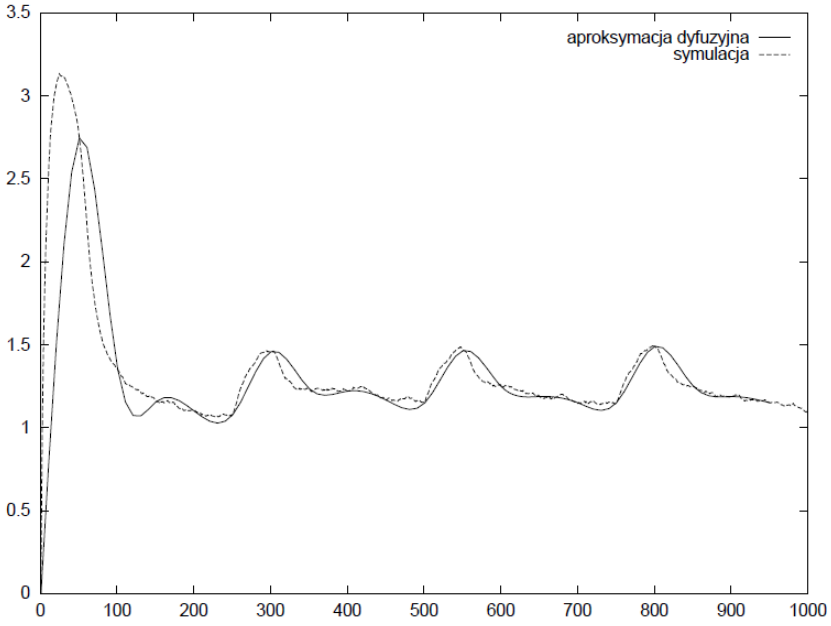


Figure 8.16

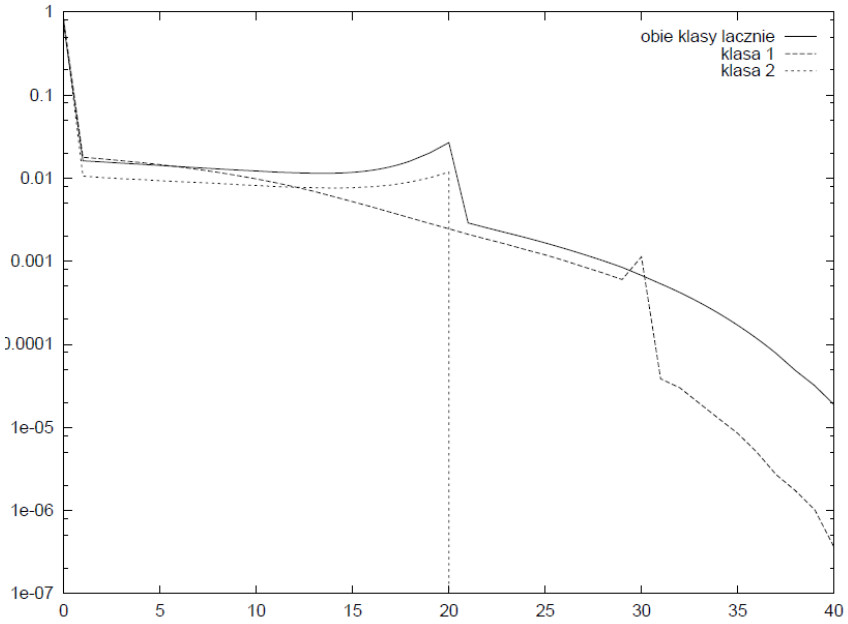


Figure 8.17: Jumping window mechanism, distribution of the queues length at muliplexer, for both classes and for each class separately, $N = 40$, $T_c = 80$, $B_c = 30$, $B_e = 60$.

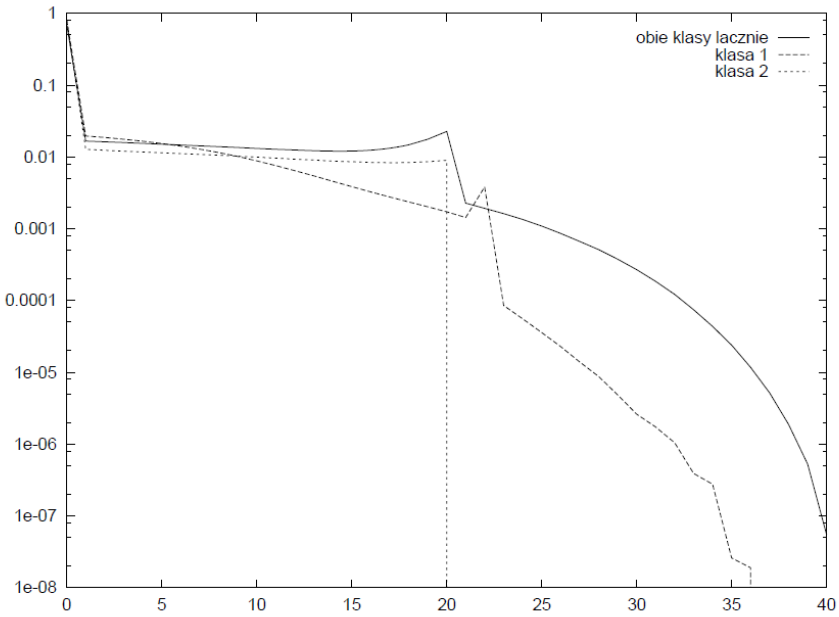


Figure 8.18: Jumping window mechanism, distribution of the queues length at muliplexer, for both classes and for each class separately, $N = 40$, $T_c = 60$, $B_c = 22$, $B_e = 44$.

no queue, the new packet cannot be accepted.

Rejected packets are then directed to a second station of type $G/D/B_e/B_e$, which allows at most B_e additional packets to pass during any interval of length T_c. Packets that cannot pass through this second station represent packets permanently rejected by the sliding-window mechanism.

Since no exact model is available for a $G/D/c/c$ station, we use the diffusion approximation described by Eqs. (7.56) and (7.60). The diffusion process is defined on the interval $(0, c)$, which is divided into c subintervals of unit length. Different diffusion parameters apply in different intervals: when i service channels are active, that is, for $x \in [i-1, i]$, we assume

$$\beta_i = \lambda - \frac{i}{T_c}, \qquad \alpha_i = \lambda C_A^2.$$

Since the service time is constant, its variance is zero, and therefore α_i depends only on the arrival process $A(x)$.

Using solution (7.56), we determine the distribution of the number of tasks in the station $G/D/B_c/B_c$, and in particular the probability p_{B_c} that the system is full.

With probability $1 - p_{B_c}$, packets are accepted by the first station. With probability p_{B_c}, packets are rejected by the first station and directed to the second station of type $G/D/B_e/B_e$.

If the original packet stream is described by the interarrival-time density $a(x)$, then the accepted stream at the first station is described by

$$a_1(x) = a(x)(1 - p_{B_c}) + a(x) * a(x)\, p_{B_c}(1 - p_{B_c}) + a(x) * a(x) * a(x)\, p_{B_c}^2(1 - p_{B_c}) + \cdots .$$

The same distribution also describes the output stream of the first station, that is, the stream of high-priority packets entering the network, because passing through the station introduces only a constant delay T_c.

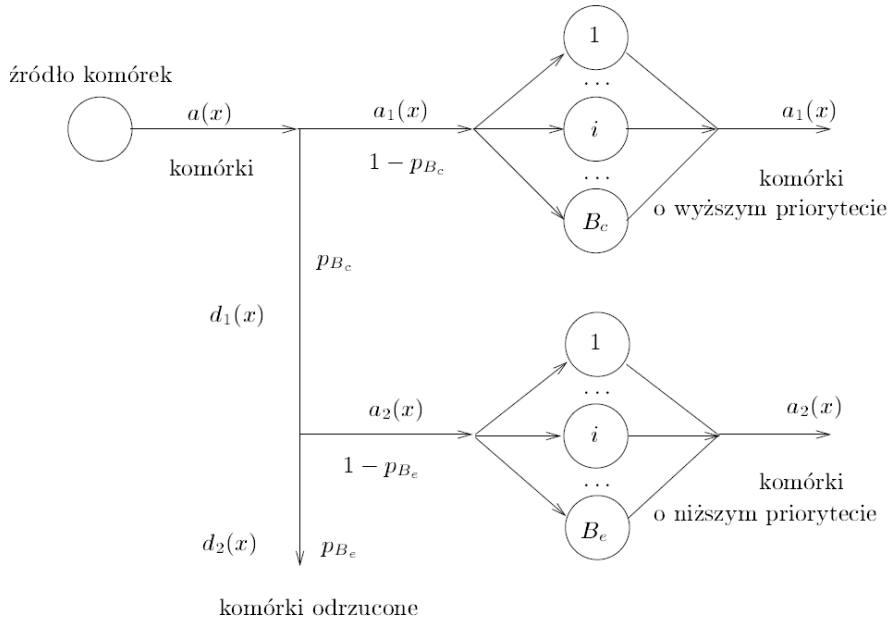


Figure 8.19: Queueing model of the sliding window mechanism

The stream of packets rejected by the first station and directed to the second station is described by

$$d_1(x) = a(x)p_{B_c} + a(x) * a(x)(1 - p_{B_c})p_{B_c} + a(x) * a(x) * a(x)(1 - p_{B_c})^2 p_{B_c} + \cdots .$$

Using the same approach as for routing probabilities in a queueing network, cf. Eq. (7.52), we can calculate the coefficient of variation of the stream arriving at the second station.

Using the diffusion model $G/D/B_e/B_e$, we determine the probability p_{B_e} that the second station is full. This probability provides an estimate of the packet-loss probability.

The stream of low-priority packets entering the network is described by the interarrival-time density

$$a_2(x) = a_1(x)(1 - p_{B_e}) + a_1(x) * a_1(x)\, p_{B_e}(1 - p_{B_e}) + a_1(x) * a_1(x) * a_1(x)\, p_{B_e}^2(1 - p_{B_e}) + \cdots .$$

Numerical examples and results for this model are presented in [5, 55].

8.3 Network switch models

8.3.1 Multiclass FIFO switch output streams dynamics

Consider a queueing network model representing a computer network. Customers (packets of fixed size) are grouped into classes. Each class represents a connection between two points of the network and its description includes the features of the source and the itinerary across the network. Customer queues at servers correspond to the queues of cells waiting at a node to be sent further. At a queue exit the class of a customer should be known to determine its routing. In steady state queuing models this probability, for a certain class k, is given by the ratio of

the throughput $\lambda^{(k)}$ of this class customers passing the node to the total througput λ of this node : $\lambda^{(k)}/\lambda$, where $\lambda = \sum_{k=1}^{K} \lambda^{(k)}$ and K is the number of classes passing across the node. In transient state, the throughputs are a function of time and the probability $\lambda^{(k)}(t)/\lambda(t)$ as well. Morover, the flows at the entrance and the exit are not the same: $\lambda^{(k)}_{i,\,in} \neq \lambda^{(k)}_{i,\,out}$. The composition of output flow reflects the previous compositions at the entrance delayed by the response time of the queue.

To solve this problem, we choose the constant service time as the time unit, divide the time axis into intervals of unitary length and assume that the flow parameters are constant during that interval, e.g. $\lambda^{(k)}(\theta)$ denotes the class k flow at station i during an interval θ, $\theta = 1, 2, \ldots$

The input stream $\sum_k \lambda^{(k)}_{in}(\theta)$ reaches the output of the queue with a delay corresponding to the queue length in the buffer at the arrival time. The part $p(n,\theta)\sum_k \lambda^{(k)}_{i,\,in}(\theta)$ of this submitted load cannot be served and the corresponding flow cannot appear at the output before the time $t = \theta + n + 1$. So, taking into account these delays, the unfinished work ready to be processed at the time t in the station which is initially empty can be expressed as:

$$U^{(k)}(t) = \sum_{n=1}^{N} \lambda^{(k)}_{in}(t-n) \cdot p(n-1, t-n) \quad \text{for} \quad k = 1, \ldots, K.$$

A similar formulation may be easily derived for initially nonempty queue knowing its composition at $t = 0$. Remark that some accumulation periods (of high or quickly increasing load) may yield that ready work at t exceeds the server capacity: $\sum_k U^{(k)}(t) > 1$. Such a phenomenon introduces some additional delay in the transfer of the input stream to the output.

To compute the output throughput $\lambda^{(k)}_{out}(t)$, we find first the *ready time* ϑ_1 of the cells at the head of the queue. This is equal to the smallest value of θ such that:

$$\sum_{\tau=0}^{\theta} \sum_k U^{(k)}(\tau) \geq \sum_{\tau=0}^{t-1} \sum_k \lambda^{(k)}_{out}(\tau). \tag{8.13}$$

Then, we determine the smallest ϑ_2 for which

$$\sum_{\tau=0}^{\vartheta_2} \sum_k U^{(k)}(\tau) - \sum_{\tau=0}^{t-1} \sum_k \lambda^{(k)}_{out}(\tau) \geq 1 - p(0,t).$$

The output throughput $\lambda^{(k)}_{out}(t)$ is obtained as

$$\lambda^{(k)}_{out}(t) = \sum_{\theta=\vartheta_1}^{\vartheta_2} w_\theta \cdot U^{(k)}(\theta) \tag{8.14}$$

where w_{ϑ_1} represents the percentage of $U^{(k)}(\vartheta_1)$ that has not been sent yet, $w_\theta = 1$ for $\theta \neq \vartheta_1$ and ϑ_2 , and w_{ϑ_2} is chosen such that $\sum_{\theta=\vartheta_1}^{\vartheta_2} w_\theta \cdot U(\theta) = 1 - p(0,t)$.

Example 8.5

Fig. 8.25 presents a model of a switch. Consider a switch having 3 input streams end one output being the sum of the three input streams. The output queue has the capacity of 100 packets and is analysed with the use of $G/D/1/100$ multiclass diffusion model and $M/D/1$ multiclass fluid approximation model. The service time is constant, $\mu = 1$. The input streams are Poisson with parameters $\lambda^{(k)}(t)$ chosen as:

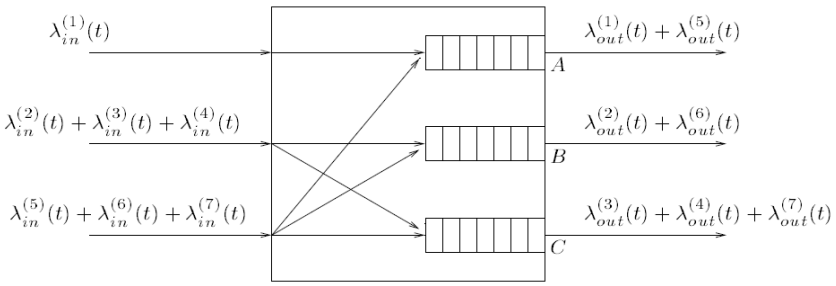


Figure 8.20: Switch model

time t	$0-10$	$10-20$	$20-30$	$30-40$	$40-80$	>80
$\lambda_{in}^{(1)}(t)$	0.1	0.8	0.8	0.1	0.1	0
$\lambda_{in}^{(2)}(t)$	0.2	0.2	0.2	0.2	0.2	0
$\lambda_{in}^{(3)}(t)$	0	0	0.5	0.5	0	0

Figures 8.21 – 8.26 present the resulting output stream (queue is empty at the beginning). These results, obtained with the use of diffusion approximation and fluid flow approximation, are compared with simulation results which represent the mean of 400 000 independent runs and practically may be considered as exact. The differences between input and output streams are clearly visible for the whole stream as well as for each class separately. They justify the need of the presented approach in the transient analysis of networks: the impact the switch queue on the flow dynamics is important and cannot be neglected. In all considered cases the results of diffusion approximation are very close to simulation and clearly better that those obtained with the fluid flow approximation.

The following model allows us to approximate the time-varying number of cells waiting for transmission in the buffers of a statistical ATM network multiplexer. On this basis, it is possible to estimate the time-dependent probability of cell loss caused by buffer overflow.

The irregularity of the cell stream and the resulting variation in the waiting time for transmission also introduce irregularity in the arrival times of cells at their destination, commonly referred to as *jitter*. The model also makes it possible to assess this effect by determining the distribution of transit times through the switch.

The analysis of ATM multiplexers is also important in the design of traffic-control mechanisms that may be incorporated into the network.

A number of models have already been proposed, e.g. [65, 44, 112, 86], but they do not address the transient behaviour of the buffer under time-varying load conditions.

8.3.2 Switch with push-out policy (*Push-Out*)

One of the most frequently considered queue-management strategies is the *push-out* policy. Let $n^{(1)}$ and $n^{(2)}$ denote the numbers of first-class and second-class packets in the queue, respectively, and let N denote the total buffer capacity.

As long as

$$n = n^{(1)} + n^{(2)} \leq N,$$

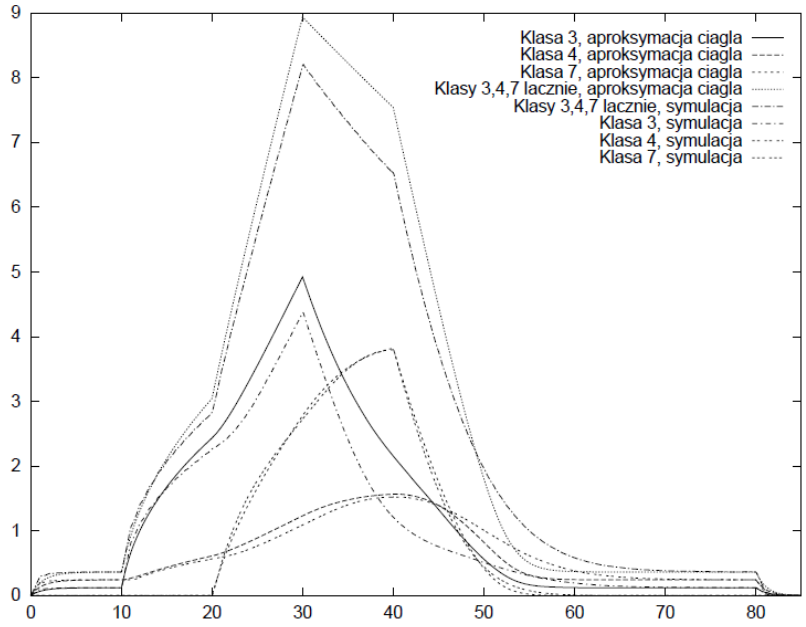


Figure 8.21: The mean queue length: global and per classes as a function of time – simulation and fluid approximation results

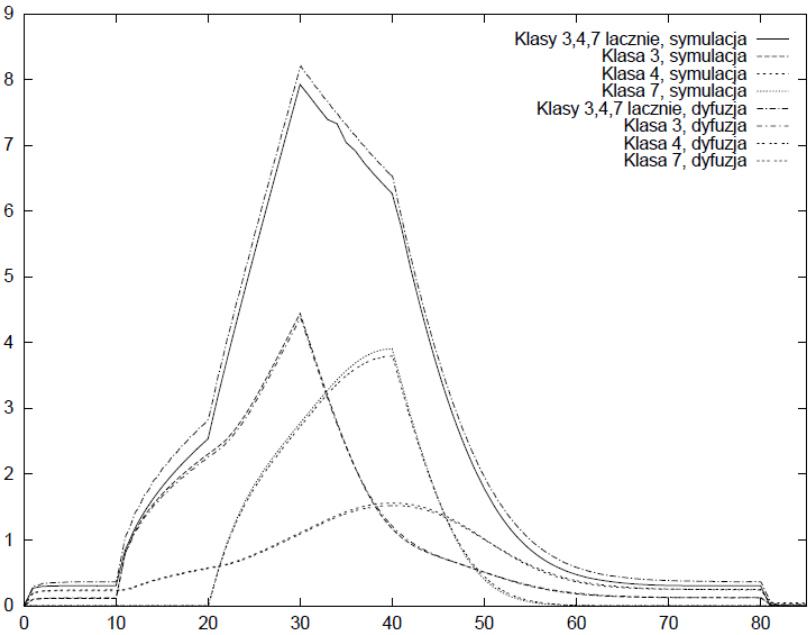


Figure 8.22: The mean queue length: global and per classes as a function of time – simulation and diffusion approximation results

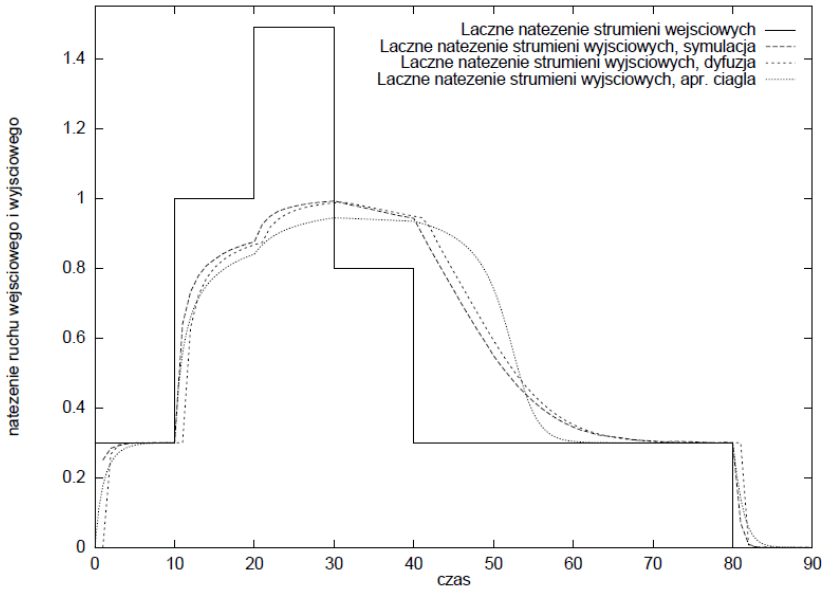


Figure 8.23: The intensities $\lambda_{in}(t)$, $\lambda_{out}(t)$ of global input and output flows, obtained by simulation, diffusion and fluid flow approximations

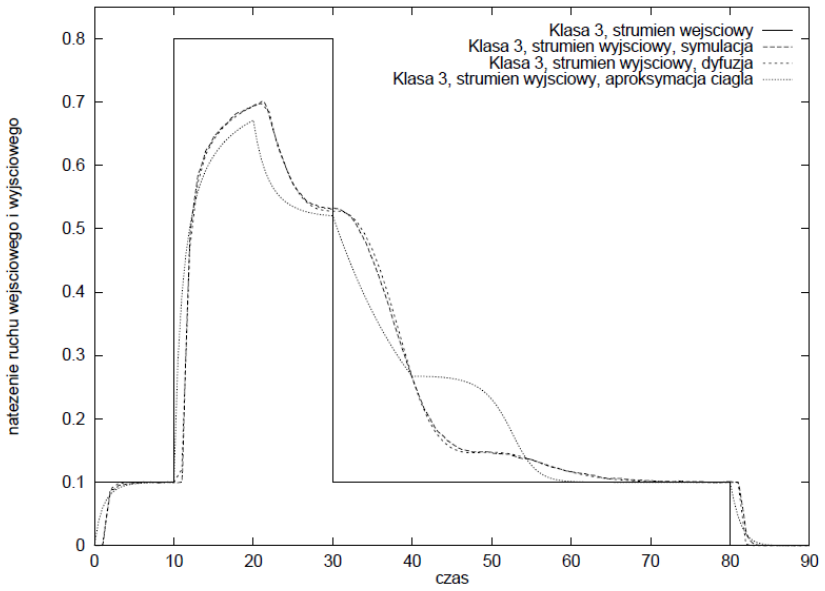


Figure 8.24: The intensities of $\lambda_{in}^{(1)}(t)$, $\lambda_{out}^{(1)}(t)$ input and output flows for class 1 traffic; the output flow is obtained by simulation, diffusion and fluid flow approximations

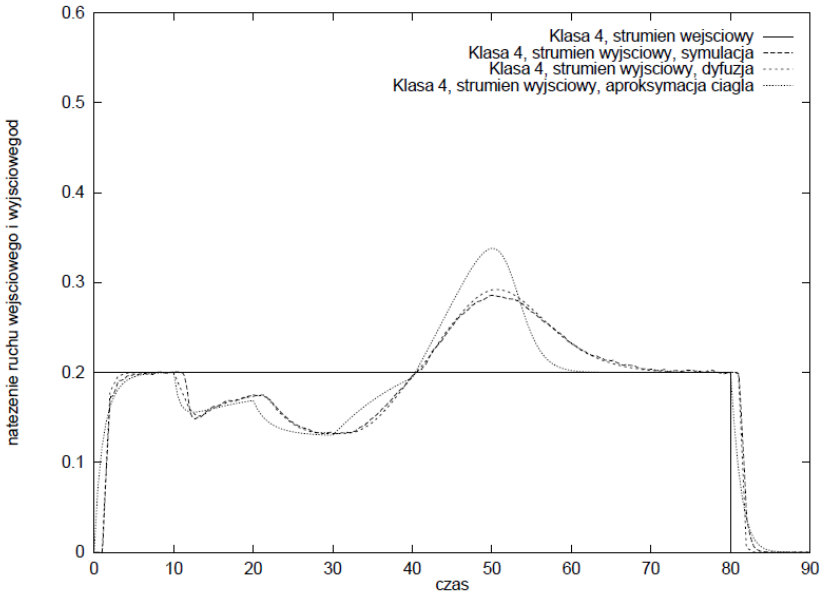


Figure 8.25: The intensities of $\lambda_{in}^{(2)}(t)$, $\lambda_{out}^{(2)}(t)$ input and output flows for class 2 traffic; the output flow is obtained by simulation, diffusion and fluid flow approximations

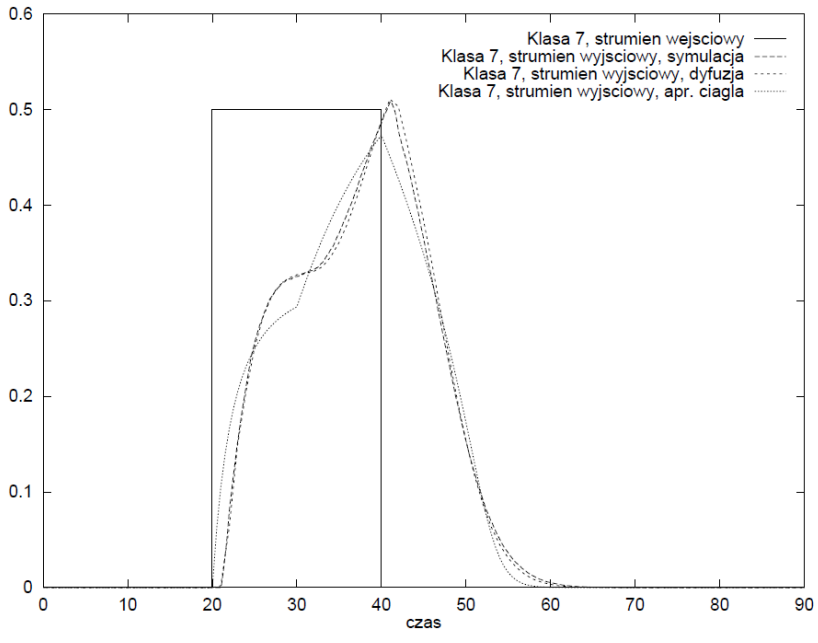


Figure 8.26: The intensities of $\lambda_{in}^{(3)}(t)$, $\lambda_{out}^{(3)}(t)$ input and output flows for class 2 traffic; the output flow is obtained by simulation, diffusion and fluid flow approximations

packets are served according to the natural queueing policy. However, once the buffer is full, newly arriving second-class packets are discarded immediately. If a first-class packet arrives while the buffer is full, it searches for a second-class packet already present in the buffer and replaces it. Only if the queue already contains exclusively first-class packets is the arriving first-class packet also lost.

The diffusion approximation is particularly useful in this context for two reasons:

- it allows arbitrary arrival-time distributions for both traffic classes, which is important because the interarrival-time distribution significantly affects queue behaviour and packet losses;
- it enables transient analysis, so that the queue can be studied during successive periods of light and heavy traffic.

During periods in which the queue is not full, the system may be described by a two-class $G/G/1/N$ model. Losses occur only during overflow periods.

To determine the effective throughput, the following iterative procedure may be used.

- Let $\lambda^{(k)[j]}$ denote the arrival intensity of class k in the j-th iteration. Initially,

$$\lambda^{(k)[0]} = \lambda^{(k)}, \qquad k = 1, 2.$$

- For operating periods without losses, the model parameters are determined from equations (7.12) and (7.13). Using the $G/G/1/N$ model and solution (7.32), we determine the functions $f(x)$, p_0, and p_N.

  The function $f(x)$ approximates the distribution $p(n)$ of the total number of packets of both classes combined. The conditional distribution

$$p(n \mid n < N)$$

  for both classes is then obtained from equation (7.14).

- Overflow occurs with probability $p(N)$. The conditional distribution of the number of packets of class k, neglecting the push-out mechanism, is approximated by

$$p^{(k)}(n^{(k)} \mid N) = \binom{N}{n^{(k)}} \left(\frac{\lambda^{(k)[j]}}{\lambda}\right)^{n^{(k)}} \left(1 - \frac{\lambda^{(k)[j]}}{\lambda}\right)^{N-n^{(k)}}, \qquad k = 1, 2. \tag{8.15}$$

- The residence time at the upper barrier $x = N$ has density

$$b^{[j]}(x) = \frac{\lambda^{(1)[j]}}{\lambda^{(1)[j]} + \lambda^{(2)[j]}}\, b^{(1)}(x) + \frac{\lambda^{(2)[j]}}{\lambda^{(1)[j]} + \lambda^{(2)[j]}}\, b^{(2)}(x).$$

- Assuming that arrivals of first-class packets during overflow periods form a Poisson process with parameter $\lambda^{(1)[j]}$, the probability of n first-class arrivals during a single overflow period is

$$p_{\text{arr}}(n) = \int_0^\infty \frac{(\lambda^{(1)[j]} x)^n}{n!} e^{-\lambda^{(1)[j]} x}\, b(x)\, dx, \qquad n = 0, 1, \ldots$$

If the service time is constant, i.e.

$$b(x) = \delta(x - 1/\mu),$$

then

$$p_{\text{arr}}(n) = \frac{(\lambda^{(1)[j]}/\mu)^n}{n!} e^{-\lambda^{(1)[j]}/\mu}, \qquad n = 0, 1, \ldots$$

- The probability that exactly n second-class packets are pushed out is then

$$p_{\text{push}}(n) = p_{\text{arr}}(n) \sum_{n_2=n+1}^{N} p^{(2)}(n^{(2)} \mid N) + p^{(2)}(n \mid N) \sum_{i=n}^{\infty} p_{\text{arr}}(i).$$

  The first term corresponds to the case in which exactly n first-class packets arrive and at least $n + 1$ second-class packets are available for removal. The second term corresponds to the case in which there are exactly n second-class packets in the queue and at least n first-class arrivals occur.

- If the first-class arrival process is not Poisson, the density $a^{(1)}(x)$ or cumulative distribution function $A^{(1)}(x)$ of the interarrival times is required. In that case, the probability of n first-class arrivals during the overflow period is

$$p_{\text{arr}}(n) = \int_0^{\infty} b(x) \int_0^x a^{(1)*n}(t) \left[1 - A^{(1)}(x - t)\right] dt\, dx,$$

  where $a^{(1)*n}$ denotes the n-fold convolution of the interarrival-time density.

In the $G/G/1/N$ system, tasks of both classes that arrive while the queue is full are lost, and the effective throughput of the station is

$$\lambda_{\text{eff}}^{(1)} = \lambda^{(1)}(1 - p_N), \qquad \lambda_{\text{eff}}^{(2)} = \lambda^{(2)}(1 - p_N)\,. \tag{8.16}$$

Under the push-out rule, $\lambda_{\text{eff}}^{(1)}$ increases, whereas $\lambda_{\text{eff}}^{(2)}$ decreases:

$$\lambda_{\text{eff}}^{(1)} = \lambda^{(1)}(1 - p_N) + p_N \lambda^{(1)} \varepsilon, \qquad \lambda_{\text{eff}}^{(2)} = \lambda^{(2)}(1 - p_N) - p_N \lambda^{(1)} \varepsilon, \tag{8.17}$$

where ε is the probability that a first-class customer arriving during an overflow period pushes a second-class customer out of the queue. It is defined as the ratio of the average number of customers pushed out to the average number of first-class arrivals during overflow:

$$\varepsilon = \frac{\displaystyle\sum_{k=1}^{N} p^{(2)}(k \mid N) \left[\sum_{i=0}^{k-1} i\, p_{\text{over}}(i) + k \sum_{i=k}^{\infty} p_{\text{over}}(i) \right]}{\displaystyle\sum_{k=1}^{\infty} k\, p_{\text{over}}(k)}\,. \tag{8.18}$$

In the case of a constant service time, the denominator of this expression is

$$\sum_{k=1}^{\infty} k\, p_{\text{over}}(k) = \frac{\lambda^{(1)}}{\mu} = \varrho^{(1)}\,.$$

We then calculate the updated effective arrival rates:

$$\lambda^{(1)[j+1]} = \lambda^{(1)} + \frac{p_N}{1 - p_N} \lambda^{(1)} \varepsilon, \qquad \lambda^{(2)[j+1]} = \lambda^{(2)} - \frac{p_N}{1 - p_N} \lambda^{(1)} \varepsilon\,.$$

$\lambda^{(1)}$	$\lambda^{(2)}$	method	$E[N]$	$E[N^{(1)}]$	$E[N^{(2)}]$
0.2	0.4	diffusion	0.831	0.277	0.554
		simulation	1.049 ± 0.007	0.350 ± 0.002	0.699 ± 0.005
0.2	0.6	diffusion	2.045	0.511	1.534
		simulation	2.391 ± 0.042	0.598 ± 0.010	1.793 ± 0.022
0.2	0.8	diffusion	10.00	2.056	7.944
		simulation	10.06 ± 0.16	2.104 ± 0.037	7.959 ± 0.123
0.2	1.2	diffusion	18.35	3.749	14.60
		simulation	18.60 ± 0.01	3.745 ± 0.012	14.86 ± 0.02

Table 8.2: Average numbers of packets in the queue for $C_{A1}^2 = C_{A2}^2 = 1$ and $N = 20$.

- We recalculate the values of $f(x)$, p_0, and p_N using the $G/G/1/N$ model, and iterate until a fixed point is reached, that is, until

$$\sum_{i=1}^{2} \left| \lambda^{(i)[j+1]} - \lambda^{(i)[j]} \right|$$

  becomes smaller than an arbitrarily chosen tolerance.

In practice, the algorithm converges after only a few iterations.
The relative losses of first- and second-class tasks are then

$$L^{(1)} = \frac{\lambda^{(1)} - \lambda_{\text{eff}}^{(1)}}{\lambda^{(1)}} = p_N(1-\varepsilon)\,, \qquad L^{(2)} = \frac{\lambda^{(2)} - \lambda_{\text{eff}}^{(2)}}{\lambda^{(2)}} = p_N\left(1 + \frac{\lambda^{(1)}}{\lambda^{(2)}}\varepsilon\right)\,. \tag{8.19}$$

Example 8.6

The accuracy of the approximate solutions must, of course, be assessed carefully. Table 8.2 compares the results of the diffusion model with simulation results. The model parameters were chosen so that the packet-loss probability was significantly higher than would be acceptable in practice, because only in that case was it possible to obtain simulation results within a reasonable amount of time and with a relative error, at the 95% confidence level, below 5% of the estimated value.

It can be seen that the queue occupancy predicted by the diffusion approximation is generally slightly lower than the actual value.

Fig. **??** presents the main result of the model calculations: the dependence of the loss probabilities of first- and second-class packets, $L^{(1)}$ and $L^{(2)}$, on the buffer capacity N, the arrival intensities $\lambda^{(1)}$, $\lambda^{(2)}$, and the variance of the input stream (parameter $C_A^{(1)^2}$). Note that the values of $L^{(1)}$ and $L^{(2)}$ are shown on a logarithmic scale. Service times are deterministic, with

$$\mu^{(1)} = \mu^{(2)} = 1.$$

In the analysis of transient states, we assume that the time axis is divided into intervals $T_1, T_2, \ldots, T_i, \ldots$, corresponding to different traffic regimes. The diffusion parameters α_i and β_i remain constant within each interval T_i and change abruptly when the system moves to the next interval.

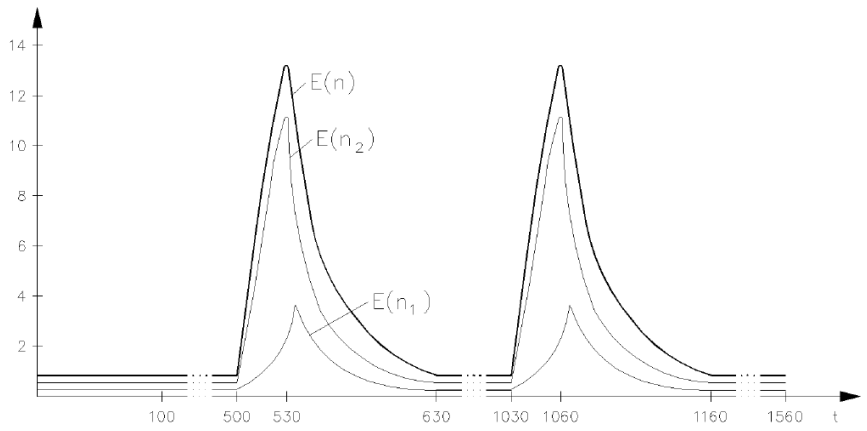


Figure 8.27: Queue occupancy during periods of low and high traffic intensity, $N = 20$.

In addition, time is divided into small intervals Δt, small relative to the lengths of the intervals T_i. Within each such segment, we use the transient-state solution for the $G/G/1/N$ system and, at the end of the segment, correct it using the iterative algorithm described above, which takes the push-out mechanism into account.

Fig. 8.27 shows an example of queue occupancy during alternating periods of low and high traffic intensity. Two phases, T_1 and T_2, occur successively. Phase T_1 corresponds to a low traffic intensity, whereas phase T_2 corresponds to a high traffic intensity. Their durations are 500 and 30 time units, respectively.

For both classes and in both phases, arrivals follow Poisson processes, and service times are constant and identical for both classes. The arrival rate of the first class remains unchanged in both phases:

$$\lambda_1^{(1)} = \lambda_2^{(1)} = 0.2.$$

The arrival rate of the second class changes from

$$\lambda_1^{(2)} = 0.4$$

during phase T_1 to

$$\lambda_2^{(2)} = 1.4$$

during phase T_2.

8.3.3 Threshold policy

In addition to push-out policies, one may also consider the so-called threshold policy. If the total number of packets n in the queue does not exceed a threshold value N_1, then packets of both classes are accepted. However, once the number of packets exceeds N_1, only high-priority packets are admitted, whereas lower-priority packets are discarded.

Let $\lambda^{(1)}$ and $\lambda^{(2)}$ denote the arrival rates of first- and second-class packets, respectively. The effective arrival rate entering the node is therefore

$$\lambda^{(1)} + \lambda^{(2)}$$

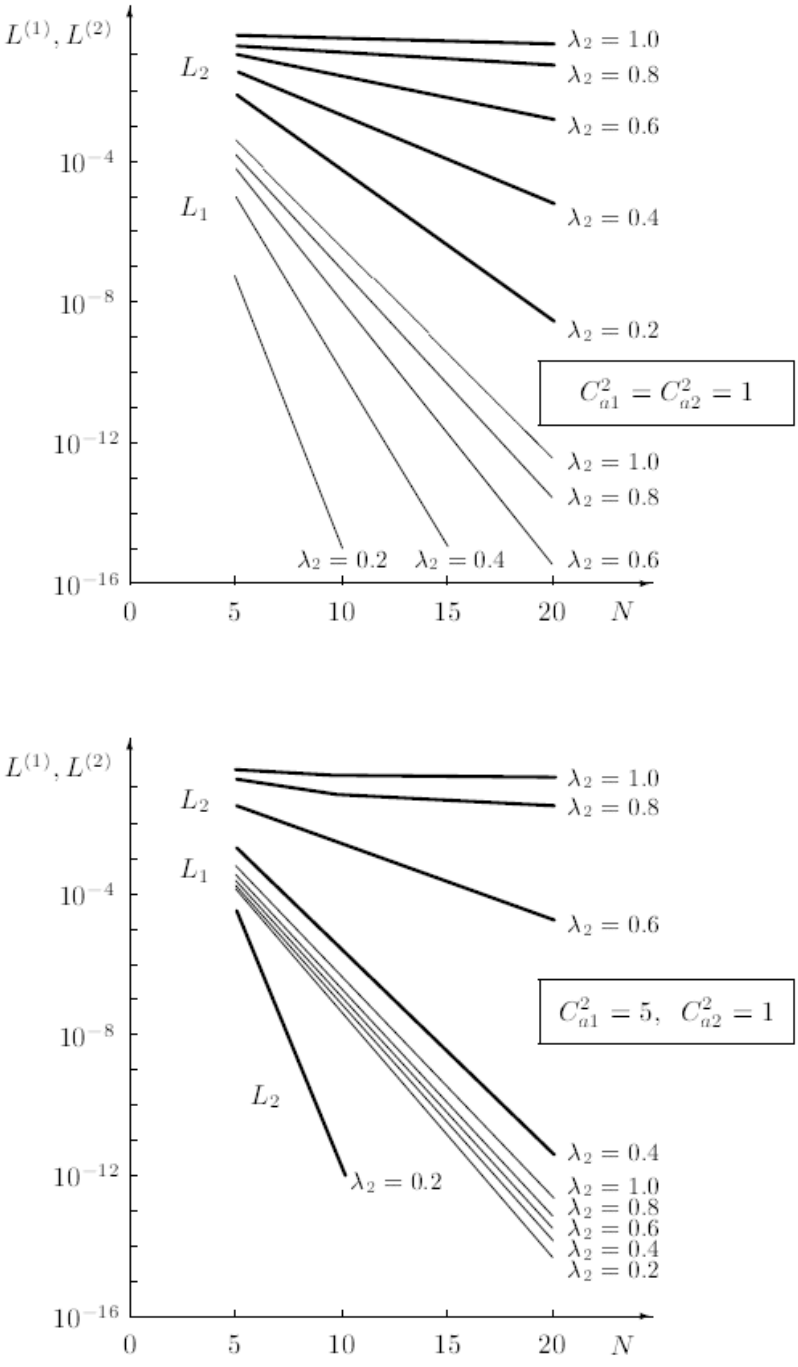


Figure 8.28: Packet-loss probabilities $L^{(1)}$ and $L^{(2)}$

when $n \le N_1$, and

$$\lambda^{(1)}$$

when $n > N_1$.

Since the effective arrival process depends on the queue occupancy, we use a diffusion model with state-dependent parameters $\alpha(x)$ and $\beta(x)$. This is the same model as that used for the finite-capacity $G/G/1/N$ queue, described by equations (7.55) and solution (7.56).

We assume

$$\beta(x) = \begin{cases} \beta_1 = \lambda^{(1)} + \lambda^{(2)} - \mu, & 0 < x \le N_1, \\ \beta_2 = \lambda^{(1)} - \mu, & N_1 < x < N, \end{cases} \tag{8.20}$$

and

$$\alpha(x) = \begin{cases} \alpha_1 = \lambda^{(1)} {C_A^{(1)}}^2 + \lambda^{(2)} {C_A^{(2)}}^2 + \mu C_B^2, & 0 < x \le N_1, \\ \alpha_2 = \lambda^{(1)} {C_A^{(1)}}^2 + \mu C_B^2, & N_1 < x < N. \end{cases} \tag{8.21}$$

Let $f_1(x)$ and $f_2(x)$ denote the density functions on the intervals $x \in (0, N_1]$ and $x \in [N_1, N)$, respectively. From solution (7.56) we obtain

$$f_1(x) = \begin{cases} \dfrac{\lambda_0 p_0}{-\beta_1} \left(1 - e^{z_1 x}\right), & 0 < x \le 1, \\ \dfrac{\lambda_0 p_0}{-\beta_1} \left(1 - e^{z_1}\right) e^{z_1 (x-1)}, & 1 \le x \le N_1, \end{cases} \tag{8.22}$$

and

$$f_2(x) = \begin{cases} f_1(N_1) e^{z_2 (x - N_1)}, & N_1 \le x \le N - 1, \\ \dfrac{\mu p_N}{-\beta_2} \left[1 - e^{z_2 (x-N)}\right], & N - 1 \le x < N. \end{cases} \tag{8.23}$$

where

$$z_n = \frac{2\beta_n}{\alpha_n}, \qquad n = 1, 2.$$

The probabilities p_0 and p_N, corresponding to the process remaining at the lower and upper barriers, are obtained from the normalisation condition.

The loss probabilities for high- and low-priority packets are then

$$L^{(1)} = p_N,$$

and

$$L^{(2)} = P[X > N_1] = \int_{N_1}^{N} f_2(x)\, dx + p_N.$$

Higher-priority cells are lost only when the entire buffer is full. Models describing this mechanism are usually based on Markov chains [43, 68, 81]. Below, we use a diffusion approximation to describe the case in which buffer partitioning is used as the queue-management rule.

The diffusion process is defined on the interval

$$x \in [0, N]$$

and represents the total number of cells of both classes present in the buffer.

The process parameters differ on the intervals

$$0 < x \leq N_1 \qquad \text{and} \qquad N_1 < x < N.$$

In the first interval, arrivals of both classes are taken into account, whereas in the second interval only arrivals of the higher-priority class are considered.

Thus,

$$\beta(x) = \begin{cases} \beta_1 = \lambda^{(1)} + \lambda^{(2)} - \mu, & 0 < x \leq N_1, \\ \beta_2 = \lambda^{(1)} - \mu, & N_1 < x < N, \end{cases} \tag{8.24}$$

and

$$\alpha(x) = \begin{cases} \alpha_1 = \lambda^{(1)} {C_A^{(1)}}^2 + \lambda^{(2)} {C_A^{(2)}}^2 + \mu C_B^2, & 0 < x \leq N_1, \\ \alpha_2 = \lambda^{(1)} {C_A^{(1)}}^2 + \mu C_B^2, & N_1 < x < N. \end{cases} \tag{8.25}$$

Since we are considering fixed-size cells with deterministic transmission times, we continue to assume that

$$C_B^2 = 0.$$

The residence time at the upper barrier $x = N$ is determined by the density given in equation (8.1); μ_N is the reciprocal of the mean residence time.

The model considered here is a special case of the model with piecewise-constant parameters used to describe the $G/G/1/N$ system with a finite customer population, cf. Section **??**. The steady-state solution is therefore given by equations (7.56), which in the present case become

$$f_1(x) = \begin{cases} \dfrac{[\lambda^{(1)} + \lambda^{(2)}]p_0}{-\beta_1} \left(1 - e^{z_1 x}\right), & 0 < x \leq 1, \\ \dfrac{[\lambda^{(1)} + \lambda^{(2)}]p_0}{-\beta_1} \left(1 - e^{z_1}\right) e^{z_1(x-1)}, & 1 \leq x \leq N_1, \end{cases} \tag{8.26}$$

and

$$f_2(x) = \begin{cases} f_1(N_1) e^{z_2(x-N_1)}, & N_1 \leq x \leq N-1, \\ \dfrac{\mu p_N}{-\beta_2} \left[1 - e^{z_2(x-N)}\right], & N-1 \leq x < N. \end{cases} \tag{8.27}$$

where

$$z_n = \frac{2\beta_n}{\alpha_n}, \qquad n = 1, 2.$$

The probabilities p_0 and p_N are determined from the normalisation condition.

The loss probability of a first-class cell, $L^{(1)}$, is equal to the probability that the entire buffer is full:

$$L^{(1)} = p_N.$$

The loss probability of a second-class cell, $L^{(2)}$, is equal to the probability that the number of cells in the buffer exceeds the threshold N_1:

$$L^{(2)} = P[X > N_1] = \int_{N_1}^{N} f_2(x)\, dx + p_N.$$

Example 8.7

Let us assume, for example, that the buffer size is $N = 100$ *and the threshold value is*

$$N_1 = 50.$$

The steady-state distribution of the total number of packets of both classes, as determined from solution (8.27) and from simulation, is shown for several input intensities on a linear scale in Fig. 8.29 and on a logarithmic scale in Fig. 8.30.

For some parameter values, the results were compared with simulation. This was possible only for relatively large probabilities; for lower probabilities, the time required for simulation exceeded reasonable limits.

Fig. 8.31 shows both the probability that the threshold is exceeded and the probability that the buffer is full. The former was compared with the second-class packet loss probability obtained from simulation. The loss probability for first-class packets was too small to be estimated reliably by simulation.

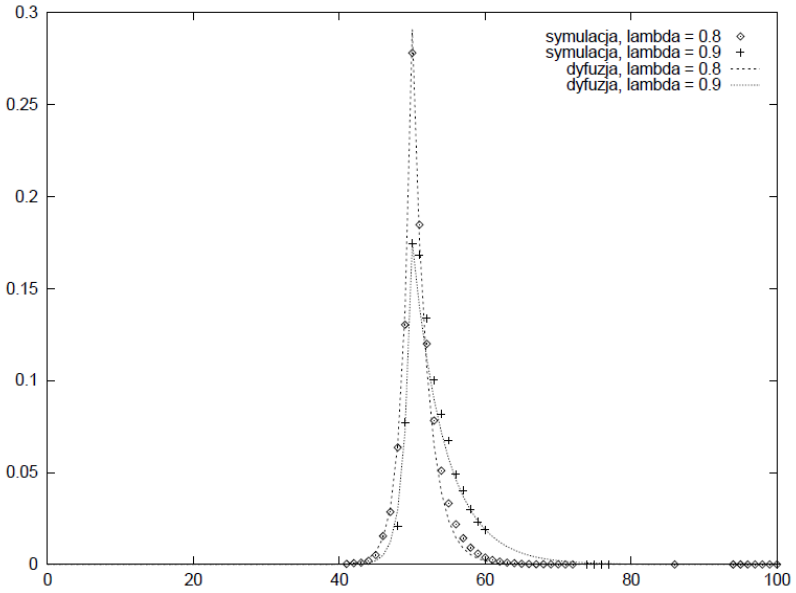


Figure 8.29: Steady-state distribution of the total number of packets for input traffic parameters $\lambda^{(1)} = \lambda^{(2)} = \lambda = 0.8, 0.9$; results of the diffusion model and simulation.

To describe the transient state, we use the same technique as in the $G/G/1/N$ model. Since there are now two diffusion regions with different parameters, there is a probability flow between them in the transient regime. This flow is balanced by introducing an additional barrier at the boundary between the two intervals, namely at $x = N_1$.

Let us consider two separate diffusion processes, $X_1(t)$ and $X_2(t)$, defined on the intervals

$$x \in (0, N_1) \qquad \text{and} \qquad x \in (N_1, N),$$

respectively.

At $x = 0$ there is a barrier with residence-time density $l_0(t)$, after which the process returns to $x = 1$. At $x = N_1$ there is an absorbing barrier for process $X_1(t)$. Let $\gamma^L_{N_1}(t)$ denote

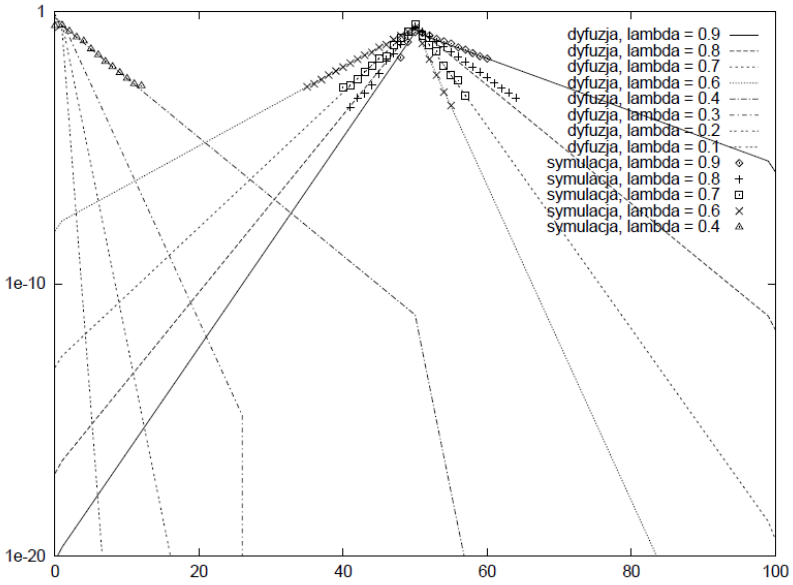


Figure 8.30: Steady-state distribution of the total number of packets for input intensities $\lambda^{(1)} = \lambda^{(2)} = \lambda = 0.1, \ldots, 0.9$; results of the diffusion model and simulation shown on a logarithmic scale.

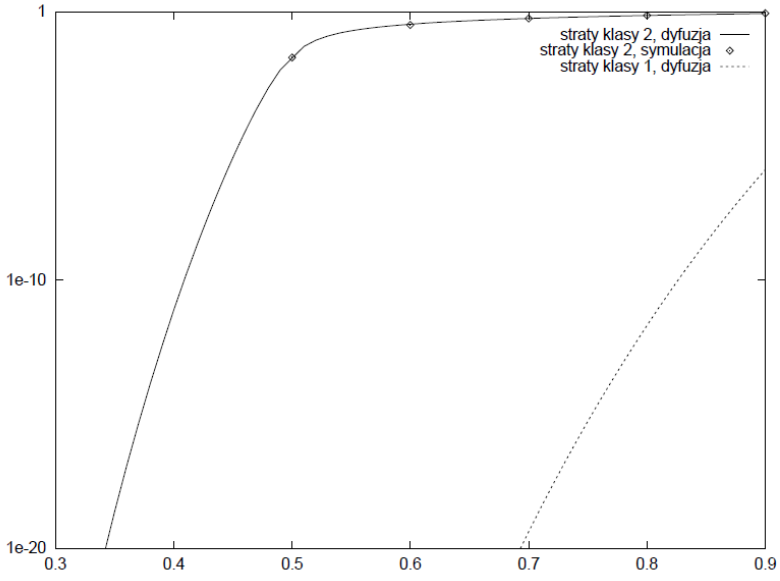


Figure 8.31: Probability of threshold exceedance and buffer overflow obtained from the diffusion model, together with the simulated probability of second-class packet loss, as a function of the input intensity $\lambda^{(1)} = \lambda^{(2)} = \lambda = 0.3, \ldots, 0.9$.

the probability density that process $X_1(t)$ reaches this barrier at time t. The process is then restarted at $x = N_1 - \varepsilon$ with intensity $g_{N_1-\varepsilon}(t)$.

The process $X_2(t)$ is bounded above by a barrier at $x = N$, with residence-time density $l_N(t)$, after which it returns to $x = N - 1$. Its lower boundary is the absorbing barrier at $x = N_1$. The process is restarted at $x = N_1 + \varepsilon$ with intensity $g_{N_1+\varepsilon}(t)$. Let $\gamma^R_{N_1}(t)$ denote the probability density that process $X_2(t)$ crosses the barrier at $x = N_1$.

The coupling between the two processes is described by

$$g_{N_1+\varepsilon}(t) = \gamma^L_{N_1}(t), \qquad g_{N_1-\varepsilon}(t) = \gamma^R_{N_1}(t),$$

that is, at every instant, the probability that one process terminates at the barrier $x = N_1$ is equal to the probability that the other process starts on the opposite side of the barrier, at a small distance $\varepsilon > 0$, introduced to prevent immediate absorption.

Similarly to equations (7.37), we define the density functions $f_1(x, t; \psi_1)$ and $f_2(x, t; \psi_2)$ as superpositions of the functions $\phi_1(x, t; x_0)$ and $\phi_2(x, t; x_0)$ for processes defined on the same intervals but with absorbing barriers at both ends:

$$\begin{aligned} f_1(x,t;\psi_1) &= \phi_1(x,t;\psi_1) + \int_0^t g_1(\tau)\phi_1(x,t-\tau;1)\,d\tau \\ &\quad + \int_0^t g_{N_1-\varepsilon}(\tau)\phi_1(x,t-\tau;N_1-\varepsilon)\,d\tau, \end{aligned} \tag{8.28}$$

$$\begin{aligned} f_2(x,t;\psi_2) &= \phi_2(x,t;\psi_2) + \int_0^t g_{N-1}(\tau)\phi_2(x,t-\tau;N-1)\,d\tau \\ &\quad + \int_0^t g_{N_1+\varepsilon}(\tau)\phi_2(x,t-\tau;N_1+\varepsilon)\,d\tau. \end{aligned} \tag{8.29}$$

To use these solutions, one must determine the densities

$$g_1(t), \qquad g_{N_1-\varepsilon}(t), \qquad g_{N_1+\varepsilon}(t), \qquad g_{N-1}(t).$$

The balance equations for the probability flows are

$$\begin{aligned}
\gamma_0(t) &= p_0(0)\delta(t) + \gamma_{\psi_1,0}(t) + \int_0^t g_1(\tau)\gamma_{1,0}(t-\tau)\,d\tau \\
&\quad + \int_0^t g_{N_1-\varepsilon}(\tau)\gamma_{N_1-\varepsilon,0}(t-\tau)\,d\tau, \\
\gamma^L_{N_1}(t) &= \gamma_{\psi_1,N_1}(t) + \int_0^t g_1(\tau)\gamma_{1,N_1}(t-\tau)\,d\tau \\
&\quad + \int_0^t g_{N_1-\varepsilon}(\tau)\gamma_{N_1-\varepsilon,N_1}(t-\tau)\,d\tau, \\
\gamma_N(t) &= p_N(0)\delta(t) + \gamma_{\psi_2,N}(t) + \int_0^t g_{N_1+\varepsilon}(\tau)\gamma_{N_1+\varepsilon,N}(t-\tau)\,d\tau \\
&\quad + \int_0^t g_{N-1}(\tau)\gamma_{N-1,N}(t-\tau)\,d\tau, \\
\gamma^R_{N_1}(t) &= \gamma_{\psi_2,N_1}(t) + \int_0^t g_{N_1+\varepsilon}(\tau)\gamma_{N_1+\varepsilon,N_1}(t-\tau)\,d\tau \\
&\quad + \int_0^t g_{N-1}(\tau)\gamma_{N-1,N_1}(t-\tau)\,d\tau. \qquad (8.30)
\end{aligned}$$

The restart intensities are then given by

$$\begin{aligned}
g_1(\tau) &= \int_0^\tau \gamma_0(t) l_0(\tau - t)\,dt, \qquad g_{N_1+\varepsilon}(t) = \gamma^L_{N_1}(t), \\
g_{N-1}(\tau) &= \int_0^\tau \gamma_N(t) l_N(\tau - t)\,dt, \qquad g_{N_1-\varepsilon}(t) = \gamma^R_{N_1}(t). \qquad (8.31)
\end{aligned}$$

Equations (8.30) and (8.31), analogous to equations (7.38) and (7.40) for the $G/G/1/N$ system, form a system of eight equations with eight unknown functions.

After applying the Laplace transform, products become ordinary products of transforms, and we obtain a linear system from which the unknown transforms can be determined:

$$\bar{g}_1(s), \qquad \bar{g}_{N_1-\varepsilon}(s), \qquad \bar{g}_{N_1+\varepsilon}(s), \qquad \bar{g}_{N-1}(s),$$

$$\bar{\gamma}_0(s), \qquad \bar{\gamma}_N(s), \qquad \bar{\gamma}^L_{N_1}(s), \qquad \bar{\gamma}^R_{N_1}(s).$$

These quantities are expressed in terms of the known transforms

$$\bar{\gamma}_{\psi_1,0}(s), \qquad \bar{\gamma}_{\psi_1,N_1}(s), \qquad \bar{\gamma}_{1,0}(s), \qquad \bar{\gamma}_{1,N_1}(s),$$

$$\bar{\gamma}_{N_1-\varepsilon,0}(s), \qquad \bar{\gamma}_{N_1-\varepsilon,N_1}(s), \qquad \bar{\gamma}_{\psi_2,N_1}(s), \qquad \bar{\gamma}_{\psi_2,N}(s),$$

$$\bar{\gamma}_{N_1+\varepsilon,N_1}(s), \qquad \bar{\gamma}_{N_1+\varepsilon,N}(s), \qquad [\bar{\gamma}_{N-1,N_1}(s), \qquad \bar{\gamma}_{N-1,N}(s),$$

which are defined by equations of the form (7.39).

In this way, the transforms

$$\bar{g}_1(s), \qquad \bar{g}_{N_1-\varepsilon}(s), \qquad \bar{g}_{N_1+\varepsilon}(s), \qquad \bar{g}_{N-1}(s)$$

are obtained and used in equations (8.28) and (8.29) to determine the transformed state densities. The original functions $f_1(x,t;\psi_1)$ and $f_2(x,t;\psi_2)$ are then recovered numerically [113, 119].

The state density of the complete process is

$$f(x,t;\psi) = \begin{cases} f_1(x,t;\psi_1), & 0 < x < N_1, \\ f_2(x,t;\psi_2), & N_1 < x < N. \end{cases}$$

The same approach can be extended to models with more than two regions having different diffusion parameters, separated by additional internal barriers.

To determine the number of cells in each class, we must account for the composition of the inflow and outflow streams. Let $p^{(i)}(t)$ denote the probability that a cell arriving at time t belongs to class i:

$$p^{(i)}(t) = \frac{\lambda_{\text{eff}}^{(i)}(t)}{\lambda_{\text{eff}}^{(1)}(t) + \lambda_{\text{eff}}^{(2)}(t)}. \tag{8.32}$$

The effective arrival rates are

$$\lambda_{\text{eff}}^{(1)}(t) = \lambda^{(1)}(t)\,[1 - p_N(t)], \qquad \lambda_{\text{eff}}^{(2)}(t) = \lambda^{(2)}(t)\,[1 - p_{n \geq N_1}(t)], \tag{8.33}$$

where $p_{n \geq N_1}(t)$ is the probability that the buffer is unavailable to second-class cells.

We know the distribution of the total number $n(t)$ of cells present in the buffer at time t. Let $\nu(t)$, with $0 \leq \nu(t) \leq n(t)$, denote the number of first-class cells at the tail of the queue behind the last second-class cell, as seen by the next arriving second-class cell.

Since the service time is one time unit, the effective service time for second-class cells is

$$1 + \nu(t).$$

Let $n^{(2)}(t)$ denote the number of second-class cells in the queue. Then

$$P\Big[\nu = i \,\Big|\, n(t) = n,\, n^{(2)}(t) = n^{(2)}\Big] = \frac{\binom{n-i-1}{n^{(2)}-1}}{\binom{n}{n^{(2)}}}, \qquad 0 \leq i \leq n - n^{(2)}. \tag{8.34}$$

If $n^{(2)} = 0$, then

$$P[\nu = i \mid n] = \delta_{i,n},$$

where $\delta_{i,n}$ is the Kronecker delta.

Hence,

$$\begin{aligned} P[\nu = i \mid n(t) = n] &= (1 - p^{(2)})^i \delta_{i,n} \\ &\quad + \sum_{n^{(2)}=1}^{\min\{n-i,N_1-1\}} \binom{n-i-1}{n^{(2)}-1} {p^{(2)}}^{n^{(2)}} (1-p^{(2)})^{n-n^{(2)}}. \end{aligned} \tag{8.35}$$

The unconditional distribution of ν is therefore

$$\begin{aligned} P[\nu = i] &= (1 - p^{(2)})^i P[n(t) = i] \\ &\quad + \sum_{l=0}^{N} P[n(t) = l] \sum_{m=1}^{\min\{l-i,N_1-1\}} \binom{l-i-1}{m-1} {p^{(2)}}^{m} (1-p^{(2)})^{l-m}. \end{aligned} \tag{8.36}$$

The mean and variance of the effective service time $B^{(2)}(t) = 1 + \nu(t)$ for second-class cells are then

$$E[B^{(2)}(t)] = 1 + E[\nu(t)], \tag{8.37}$$

$$C_B^{(2)^2}(t) = \frac{E[(B^{(2)}(t))^2]}{E[B^{(2)}(t)]^2} - 1 = \frac{E[\nu^2(t)] - E[\nu(t)]^2}{(E[\nu(t)] + 1)^2}. \tag{8.38}$$

The coefficient $C_B^{(2)^2}$ is given by equation (8.38); the coefficient $C_B^{(1)^2}$ can be determined similarly. The coefficients $C_A^{(1)^2}$ and $C_A^{(2)^2}$ are obtained from the corresponding input streams.

Changes in the arrival rates at time t influence the queue composition with a delay proportional to $n(t)$, and influence the effective service time of second-class cells with a delay of approximately $\mu^{(2)}(t + n(t) - \nu(t))$.

The composition of the end of the queue, represented by $\nu(t)$, depends not only on the instantaneous class probabilities $p(t)$, but also on their values over the interval since the arrival of the last second-class cell. Since this dependence is difficult to describe exactly, an approximation is used in which the relevant delay is taken to be $n(t)/2$.

The characteristics of the output streams are determined separately for each class using the approximation

$$d^{(i)}(x) = \rho^{(i)} b^{(i)}(x) + \left(1 - \rho^{(i)}\right) a^{(i)}(x) * b^{(i)}(x), \tag{8.39}$$

where $*$ denotes convolution.

This allows the coefficient of variation of the interdeparture times for class i to be estimated as

$$C_D^{(i)^2} = C_A^{(i)^2}(1 - \rho^{(i)}) + \rho^{(i)^2} C_B^{(i)^2} + \rho^{(i)}(1 - \rho^{(i)}). \tag{8.40}$$

The transient-state solution given by equations (8.28) and (8.29) assumes diffusion parameters that are constant over time. This assumption can be relaxed by dividing time into short intervals within which the parameters α and β are assumed constant, and by using the solution at the end of interval l as the initial condition for interval $l + 1$.

Example 8.8

Let us assume the following numerical values. During the interval $t \in [0, 100]$*, the stream of priority cells has intensity*

$$\lambda^{(1)} = 2$$

cells per unit of time, and the stream of lower-priority cells has intensity

$$\lambda^{(2)} = 1.$$

For $t > 100$*, the intensity of the priority stream decreases to*

$$\lambda^{(1)} = 0.6667,$$

whereas

$$\lambda^{(2)} = 1$$

remains unchanged.

The service time is constant and equal to one unit of time. The buffer size is $N = 100$*, and the threshold value varies from*

$$N_1 = 50 \qquad \textit{to} \qquad N_1 = 90.$$

The parameter ε in equations (8.28)–(8.30) was set to

$$\varepsilon = 0.1.$$

Figures 8.32 and 8.33 show the distribution of the number of cells in the buffer at selected time instants, as determined by simulation and by the diffusion approximation. To preserve clarity, the results of the two methods, which are very similar, are presented in separate plots.

At the end of the second time period under consideration ($t = 400, 500, 600$), the distribution has essentially stabilised. Figures 8.34 and 8.35 show the mean number of cells in the buffer as a function of time.

In the first period (the initial 100 time units), the system is heavily congested: the buffer fills rapidly. In the second period, congestion remains high. The probability of exceeding the threshold (and thus losing lower-priority cells) is approximately 0.7, but, in accordance with the buffer-partition policy, the probability of losing priority cells remains small; see Figs. 8.36 and **??**.

The threshold value N_1 appears as a parameter in the graphs. In the case considered here, as N_1 increases, the average number of lower-priority cells also increases: they are allocated more buffer space and, under heavy congestion, they occupy it. At the same time, the number of priority cells also increases slightly, because the larger total occupancy of the buffer causes priority cells to spend more time waiting in the queue.

Figure 8.38 shows the mean number of cells, broken down by class, calculated according to equations (8.32), (8.38), and the procedure described above. The corresponding simulation results are also shown.

As can be seen, in the final period, when the system approaches steady state, the mean number of second-class cells is underestimated by the diffusion model. This results from an overestimation of the loss probability for that class, as shown in Fig. 8.37. Nevertheless, the model correctly predicts the dynamics of the changes in the numbers of cells of both classes.

It can be observed that, in the initial period, the number of cells of both classes increases. Then, after the threshold is exceeded, the number of lower-priority cells begins to decrease, because they are no longer admitted to the buffer. Finally, when the offered load decreases and the queue starts to stabilise, the ratio between the numbers of cells of the two classes present in the buffer also approaches a stable value.

8.3.4 Switching with taking packet size into account

Packets may have different lengths: each packet consists of a certain number (from 1 to K) of blocks of fixed size. The buffer capacity N is defined as the number of blocks it can hold. Transmission, and thus freeing of space in the buffer, occurs block by block. When the buffer is close to full, only those packets are accepted whose number of blocks does not exceed the number of free buffer slots.

We assume that the diffusion process represents the number of blocks currently stored in the buffer. We assign a separate customer class to each packet size: class k, $k = 1, 2, \ldots, K$, corresponds to the arrival of a batch of k tasks.

In the case of batch arrivals from a single customer class, the diffusion parameters are chosen as

$$\beta = \lambda\nu_g - \mu, \qquad \alpha = \lambda\nu_g(C_A^2 + C_g^2) + \mu C_B^2,$$

where ν_g is the mean batch size and C_g^2 is the squared coefficient of variation of the batch-size distribution.

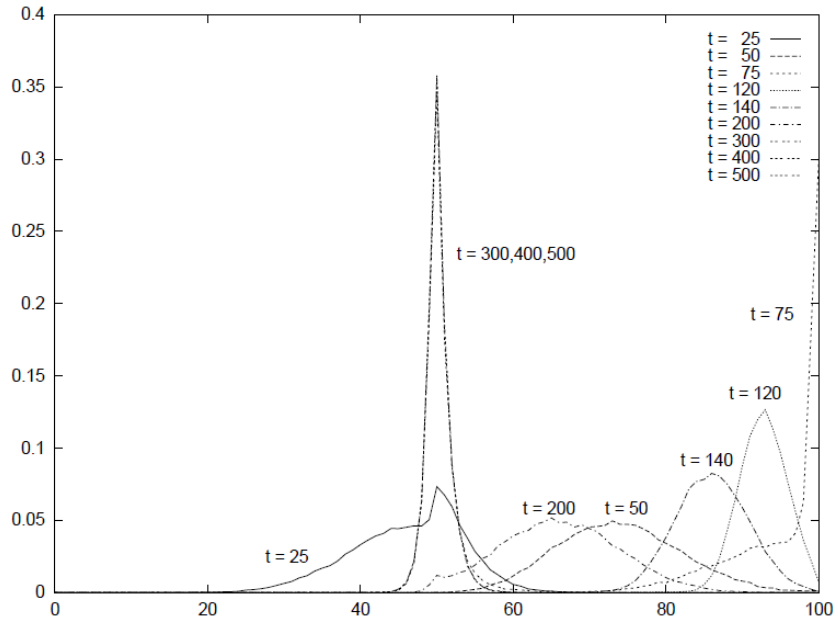


Figure 8.32: Distribution of the total number of cells of both classes in the buffer for selected times $t = 25, \ldots, 500$; buffer size $N = 100$, threshold value $N_1 = 50$; simulation results

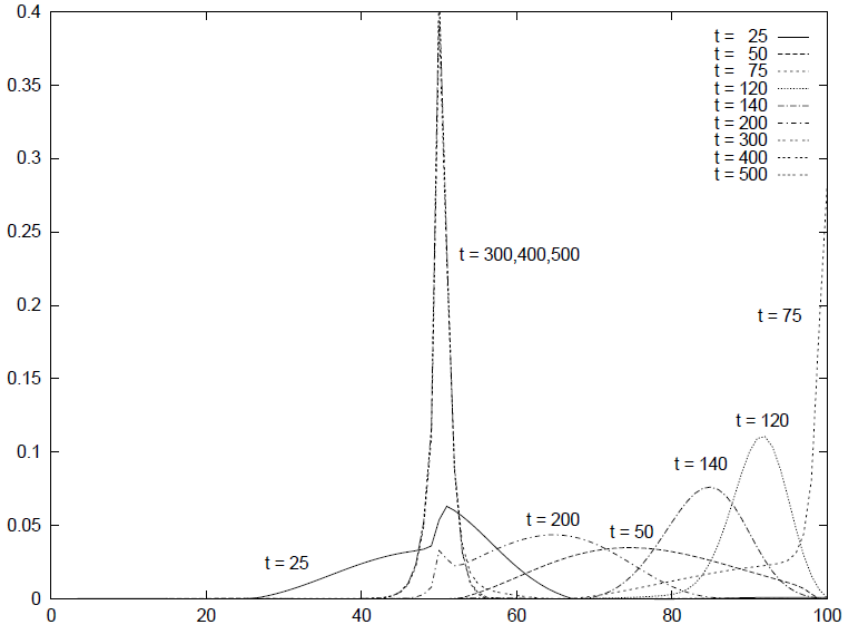


Figure 8.33: Distribution of the total number of cells of both classes in the buffer for selected times $t = 25, \ldots, 500$; buffer size $N = 100$, threshold value $N_1 = 50$; results of the diffusion approximation

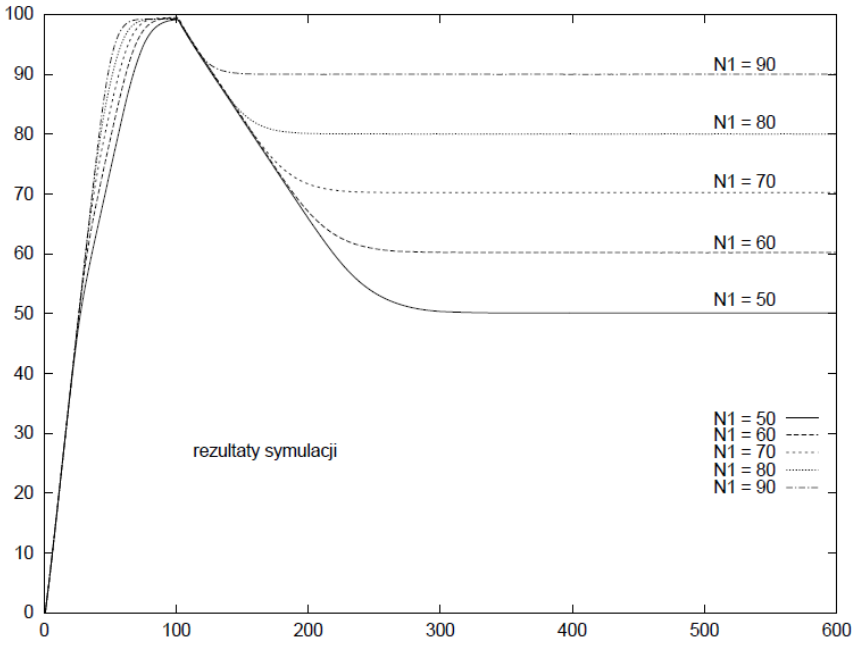


Figure 8.34: Mean number of cells as a function of time; the threshold value $N_1 = 50, 60, 70, 80, 90$ appears as a parameter; simulation results

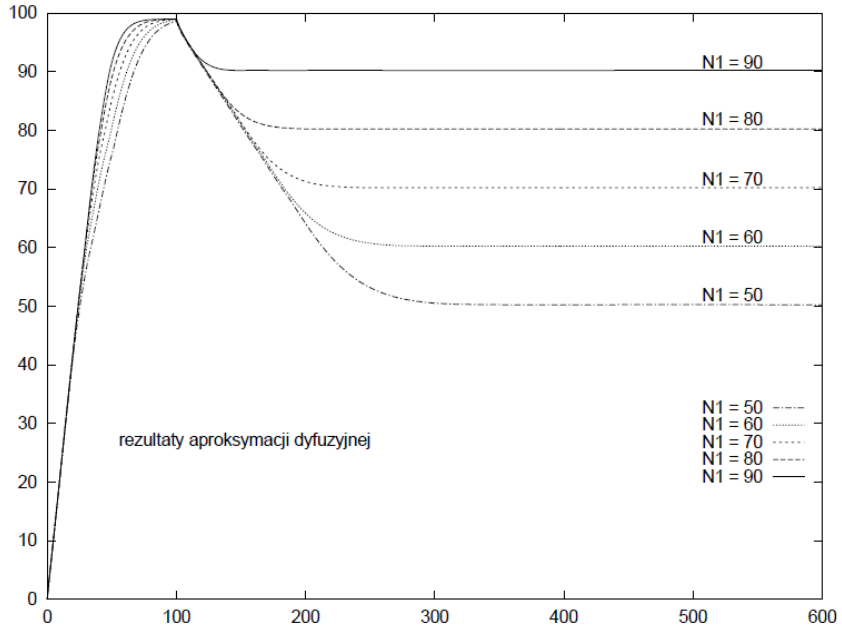


Figure 8.35: Mean number of cells as a function of time; the threshold value $N_1 = 50, 60, 70, 80, 90$ appears as a parameter; results of the diffusion approximation

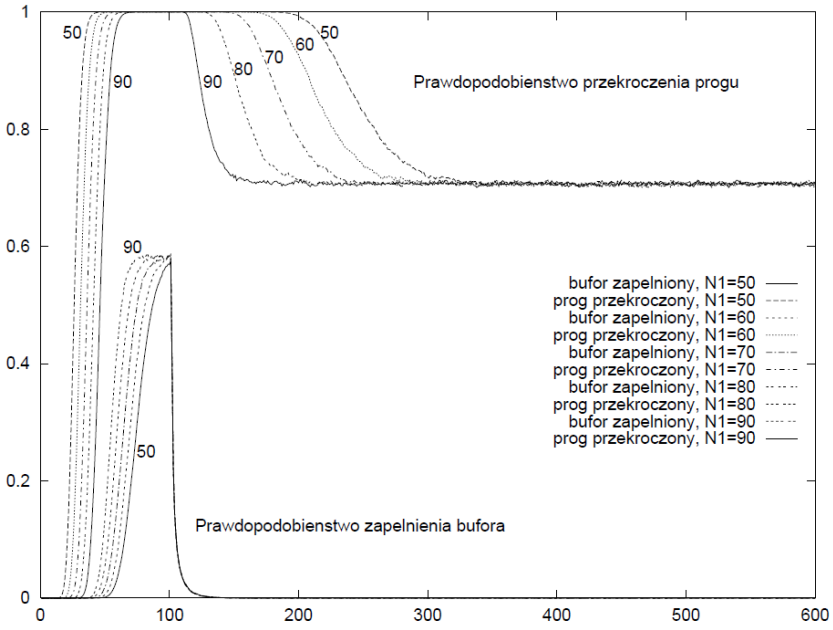


Figure 8.36: Probability that a buffer of size $N = 100$ is full, and probability that the threshold is exceeded, as functions of time; the threshold value $N_1 = 50, 60, 70, 80, 90$ appears as a parameter; simulation results

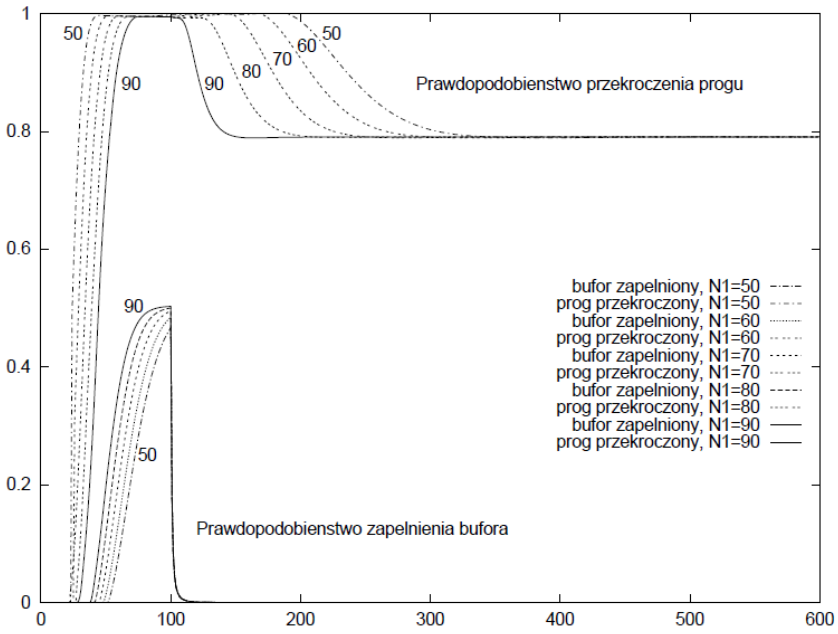


Figure 8.37: Probability that a buffer of size $N = 100$ is full, and probability that the threshold is exceeded, as functions of time; the threshold value $N_1 = 50, 60, 70, 80, 90$ appears as a parameter; results of the diffusion approximation

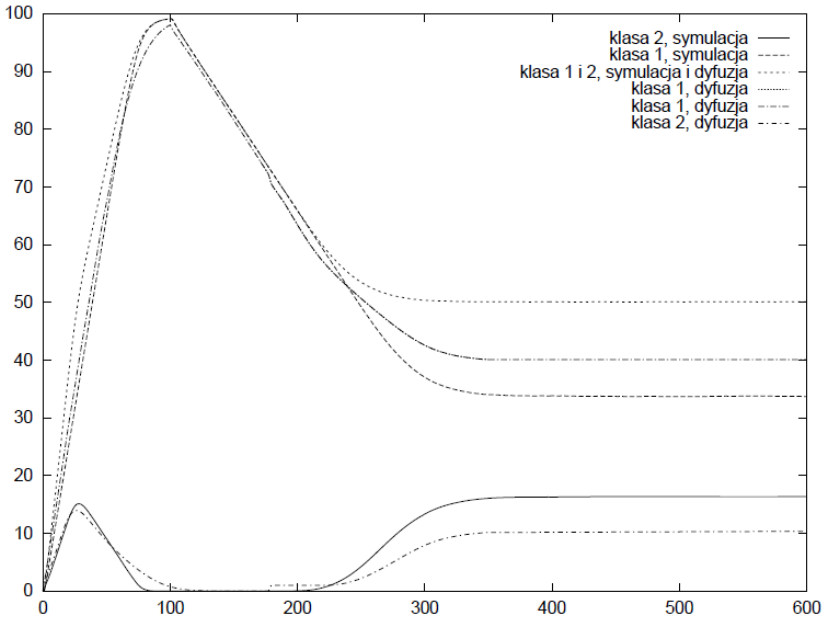


Figure 8.38: Mean numbers of first-class and second-class cells as functions of time; $N = 100$, $N_1 = 50$

In our case, for class k, the batch size is constant and equal to k, so that

$$\nu^{(k)} = k, \qquad {C_g^{(k)}}^2 = 0.$$

The parameters of the aggregate input stream are therefore, according to formulas (7.12) and (7.13) for a multi-class flow,

$$C_A^2 = \sum_{k=1}^{K} \frac{\lambda^{(k)[o]}}{\lambda^{[o]}}\, k\, {C_A^{(k)}}^2, \qquad \frac{1}{\mu} = \sum_{k=1}^{K} \frac{\lambda^{(k)[o]}}{\lambda^{[o]}} \frac{1}{\mu^{(k)}}, \tag{8.41}$$

where

$$\lambda^{[o]} = \sum_{k=1}^{K} \lambda^{(k)[o]}.$$

The intensity $\lambda^{(k)[o]}$ takes into account losses due to packet rejection when the packet no longer fits into the buffer; it is obtained as the steady-state value in the next iteration. We choose the initial values

$$\lambda^{(k)[1]} = \lambda^{(k)}.$$

We then calculate the parameters (8.41), determine the distribution of the number of blocks in the buffer using the $G/G/1/N$ model, and compute the rejection probability for 1-block packets as p_N. Accordingly, we set

$$\lambda^{(1)[2]} = \lambda^{(1)[1]}(1 - p_N).$$

The rejection probability for 2-block packets is $p_N + p(N-1)$, hence

$$\lambda^{(2)[2]} = \lambda^{(2)[1]}\,[1 - p_N - p(N-1)],$$

and, more generally,

$$\lambda^{(k)[2]} = \lambda^{(k)[1]}\big[1 - p_N - p(N-1) - \cdots - p(N-k+1)\big].$$

For the new values of the input intensities, we recalculate the distribution of the number of blocks using the model with state-dependent parameters, that is, solution (7.56), assuming that for $x < K$ all customer classes are included in the diffusion parameters, whereas for $N-k-1 < x < N-k$ only classes $1, \ldots, k$ are included.

8.3.5 Synchronous switch

In many networks, a switch does not transmit packets continuously, but synchronously, at fixed time intervals (time slots). Assume that packets are stored in a buffer of length N, and that transmissions occur every T time units. At most r packets may be transmitted at one such instant.

A diffusion process defined on the interval $x \in (0, N)$ and considered over the period $[0, T]$ allows us to determine the buffer occupancy. We proceed as in the description of the sliding window (cf. Section 8.2.3): we model the process describing the accumulation of arriving packets, with parameters

$$\beta = \lambda, \qquad \alpha = \lambda C_A^2.$$

At the point $x = N$ there is an absorbing barrier: when the process reaches $x = N$, it remains there until the end of the period $[0, T]$. Since the process has positive drift ($\beta > 0$), we neglect the boundary at $x = 0$ for simplicity.

If the process starts at $x = 0$, then its density function $p(x, t; 0, N)$, where we explicitly indicate both the starting point and the location of the absorbing barrier, has the form (cf. equation (8.6))

$$p(x,t;0,N) = \frac{1}{\sqrt{2\alpha\Pi t}}\left[\exp\left(-\frac{(x-\beta t)^2}{2\alpha t}\right) - \exp\left(\frac{2\beta N}{\alpha} - \frac{(x-2N-\beta t)^2}{2\alpha t}\right)\right]. \tag{8.42}$$

Let $g(t)$ denote the probability density of first passage to the barrier at $x = N$, cf. equation (8.7). Then

$$g(t) = -\frac{d}{dt}\int_{-\infty}^{N} p(x,t;0,N)\,dx = \frac{N}{\sqrt{2\alpha\pi t^3}}\exp\left(-\frac{(N-\beta t)^2}{2\alpha t}\right). \tag{8.43}$$

The probability that at the end of the interval T the process is at N is

$$p(N,T) = \int_0^T g(t)\,dt. \tag{8.44}$$

Let us now describe the behaviour of the multiplexer over successive time intervals T_i, $i = 1, 2, \ldots$. At the start of each interval, we reset the process time to $t = 0$.

If the number of packets at the end of interval T_i is $n_i = n \geq r$, then at the start of the next interval the number of packets is $n - r$. Otherwise, if $n_i < r$, the buffer becomes empty.

In the diffusion model, this is represented by taking the initial conditions $\psi^{i+1}(x)$ in interval T_{i+1} as

$$\begin{aligned} p_0^{i+1} &= \int_0^r f^i(x,T;\psi^i,N)\,dx, \\ \psi^{i+1}(y) &= \begin{cases} f^i(y+r,T;\psi^i,N), & 0<y<N-r, \\ 0, & N-r<y<N. \end{cases} \end{aligned}$$

Thus, if at the end of interval T_i the process value is $x_f^i > r$, then at the start of interval T_{i+1} the process is reset to

$$x_0^{i+1} = x_f^i - r,$$

whereas if $x_f^i \le r$, it is reset to zero.

Equations (8.42) and (8.43) can now be rewritten for interval T_{i+1}, taking the new initial conditions into account. The density of the process is

$$f^{i+1}(x,t;\psi^{i+1},N) = \int_0^N p(x-\xi,t;0,N-\xi)\,\psi^{i+1}(\xi)\,d\xi. \tag{8.45}$$

The probability density that a process starting at $x=\xi$ at time $t=0$ reaches the barrier at $x=N$ at time t is

$$g^{i+1}(t;\xi) = \frac{N-\xi}{\sqrt{2\alpha\pi t^3}} \exp\left(-\frac{(N-\xi-\beta t)^2}{2\alpha t}\right), \tag{8.46}$$

and, for the whole process with distributed initial condition,

$$g^{i+1}(t) = \int_0^N g^{i+1}(t;\xi)\,\psi^{i+1}(\xi)\,d\xi. \tag{8.47}$$

The probability that at the end of the interval the process is at N, that is, that the buffer is full, is calculated from equation (8.44).

When analysing transient states, calculations are performed over the time horizon of interest, divided into segments of length T. The process parameters must remain constant within each such segment, or else the segment must be subdivided further.

If the process parameters do not depend on time and we are interested in the steady state, then the calculations are continued until convergence, that is until

$$f^{i+1}(x,T;\psi^{i+1},N) \approx f^i(x,T;\psi^i,N) \qquad \text{as } i\to\infty.$$

8.3.6 Multimedia switch

Let us consider a model of a statistical switch in an integrated-services network providing two types of service: wideband (*wideband*, *WB*) and narrowband (*narrowband*, *NB*). A *WB* connection requires the allocation of b channels to each customer, whereas an *NB* connection requires only one channel.

There are $r_1 b$ channels allocated to *WB* traffic and r_2 channels allocated to *NB* traffic. Thus, the total number of channels is $r_1 b + r_2$.

Customers of both types are stored in two separate buffers: one of capacity N_1 for *WB* customers and one of capacity N_2 for *NB* customers. Customers arriving to a full buffer are

lost. If, at the end of a time slot, some channels reserved for *WB* traffic remain unused, they may be allocated temporarily to *NB* customers.

An analysis of the switch described above, based on multidimensional Markov chains, is presented in [15], while an adaptation of the diffusion model $G/D/c/N$ to describe the operation of this switch is given in [62].

Here we use the synchronous-switch model introduced in the previous subsection. Let N_1 denote the buffer size for first-class customers (*WB*), and N_2 the buffer size for second-class customers (*NB*). The maximum numbers of customers served in a single time slot are r_1 and r_2, respectively.

Let us assume two diffusion processes, $X_1(t)$ and $X_2(t)$. The process $X_1(t)$ is the same as in the previous model, except that each first-class customer requires b service channels. Therefore, we assume

$$\beta_1 = b\lambda_1, \qquad \alpha_1 = b\lambda_1 C_{A1}^2.$$

The process $X_2(t)$ depends on $X_1(t)$. This dependence can be expressed through the initial condition of the process $X_2(t)$: for the period $(i+1)$, the initial density function ψ_2^{i+1} takes the form

$$\begin{aligned} p_{2,0}^{i+1} &= \int_0^{r_2} f_2^i(x, T; \psi_2^i, N_2)\, dx \\ &\quad + \sum_{j=1}^{r_1-1} \left[1 - \sum_{n=0}^{j} p_1(r_1 - n)\right] \int_{r_2+j}^{r_2+j+1} f_2^i(x, T; \psi_2^i, N_2)\, dx \end{aligned}$$

and

$$\psi_2^{i+1}(y) = \begin{cases} p_{2,0}^{i+1}, & \text{for } y = 0, \\ f_2^i(y + r_1 + r_2, T; \psi_2^i, N_2) \left[1 - \sum_{n=0}^{r_1-1} p_1(r_1 - n)\right], & \text{for } 0 < y < N_2 - (r_1 + r_2), \\ \int_0^{r_1-1} f_2^i(y + r_2 + j, T; \psi_2^i, N_2) \left[1 - \sum_{n=0}^{j} p_1(r_1 - n)\right] dj, & \text{for } N_2 - (r_1 + r_2) < y < N_2 - r_2, \\ 0, & \text{for } N_2 - r_2 < y < N_2. \end{cases}$$

8.4 Correlation in the input stream

In integrated-services networks, strong variability is observed in the traffic intensity generated by users, as well as autocorrelation in the transmitted packet stream. These properties are illustrated by the signals shown in Figs. 8.39 and 8.40.

Figure 8.39, based on the measurements reported in [105, 104], shows an example of the time-varying transmission rate of moving images encoded in the MPEG standard, whereas Fig. 8.40 shows the autocorrelation function calculated for the same signal.

The observed nature of such data streams has a significant impact on the behaviour of packet queues waiting for transmission at network nodes, and thus on the quality of service offered by the network, expressed through the packet-loss probability and the variance of packet-transmission times along the entire virtual path connecting two network nodes.

Queueing models used in computer-network analysis usually do not account for time-varying transmission intensity; instead, they rely mainly on average traffic rates and analyse the network

under steady-state conditions. They also generally ignore autocorrelation in the traffic streams, assuming that successive interarrival times are independent random variables.

One of the few approaches that allows transient behaviour to be taken into account, and thus permits the description of time-varying information flow, is the diffusion approximation. Below we show how this method can be adapted to describe correlated information streams. The results and numerical examples are taken from [25, 26].

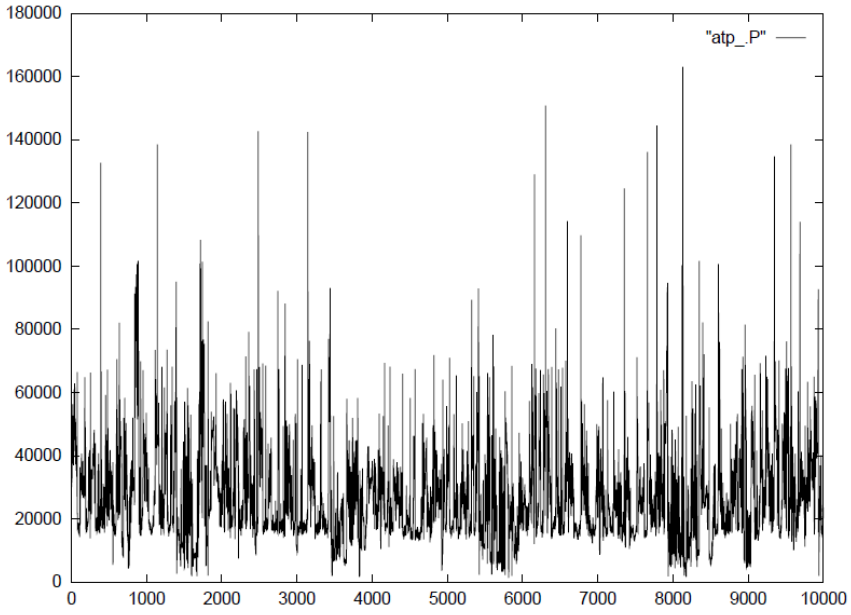


Figure 8.39: Example of a measured traffic-intensity sequence during transmission of encoded moving images; the horizontal axis shows the frame number, and the vertical axis shows the frame size in bits

8.4.1 The influence of correlation and time-varying stream parameters on the coefficients of the diffusion model

The traffic stream generated by network users is most often described by means of an MMPP (*Markov Modulated Poisson Process*). In this model, the time-varying statistical properties of the traffic are determined by the transient states of an underlying Markov chain.

Let us assume that the modulating Markov chain has M states. When the chain is in state i, the parameter of the corresponding Poisson process is λ_i, which means that packet arrivals occur at intervals described by an exponential distribution with mean λ_i^{-1}.

In the special case $M = 2$, one obtains an *ON–OFF* process. Markov models of systems described in this way have often been used to analyse both steady-state and transient phenomena associated with packet traffic, for example in [7, 69, 73, 72].

However, the size of the state space and numerical difficulties limit the practical use of exact Markov models mainly to systems consisting of a single service station.

Using the diffusion approximation, we may generally assume that a Markov chain defines phases in which the input stream, not necessarily Poisson, is characterised by the first two

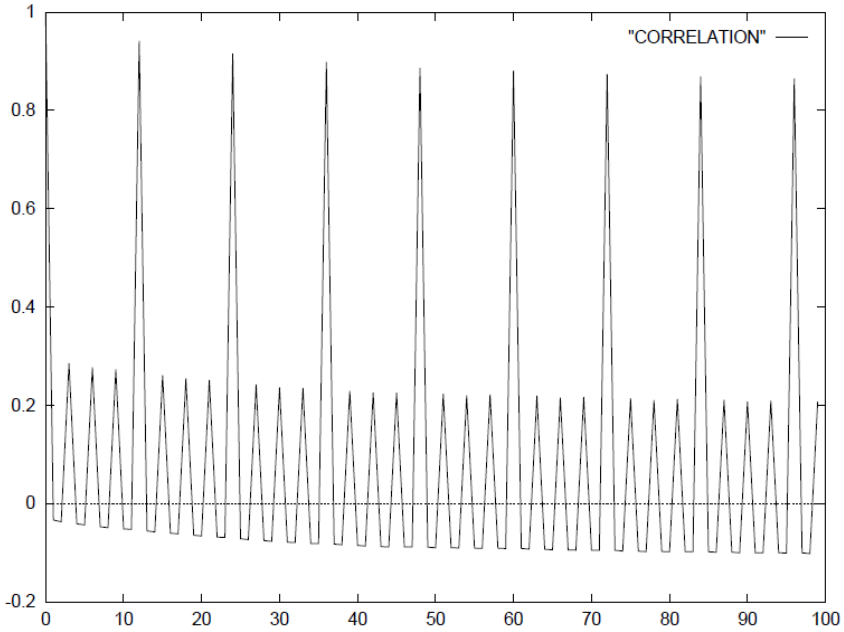


Figure 8.40: Autocorrelation of the traffic intensity (frame size) shown in Fig. 8.39, as a function of frame-number lag

moments of the distribution of interarrival times, $E[A_i] = \lambda_i^{-1}$ and $E[A_i^2]$.

Thus, if $\pi_i(t)$ denotes the probability that at time t the Markov process modulating the packet stream is in state i, then the first two moments of the distribution of interarrival times are

$$E[A(t)] = \frac{\sum_{i=1}^{M} \pi_i(t)\lambda_i E[A_i]}{\sum_{i=1}^{M} \pi_i(t)\lambda_i} = \frac{\sum_{i=1}^{M} \pi_i(t)}{\sum_{i=1}^{M} \pi_i(t)\lambda_i} = \frac{1}{\sum_{i=1}^{M} \pi_i(t)\lambda_i} = \frac{1}{\lambda(t)}, \tag{8.48}$$

$$E[A(t)^2] = \frac{\sum_{i=1}^{M} \pi_i(t)\lambda_i E[A_i^2]}{\sum_{i=1}^{M} \pi_i(t)\lambda_i}. \tag{8.49}$$

If $\lambda_i = 0$, i.e. the system is silent in phase i, then the term $E[A_i^2]$ in equation (8.49) is replaced by the second moment of the duration of phase i, while $\lambda_i = 0$ is replaced by the reciprocal of the mean duration of that phase.

The solution of the system of equations

$$\frac{d\pi_i(t)}{dt} = \sum_k \pi_k(t) q_{ki},$$

where q_{ki} are the transition rates from state k to state i, gives the state probabilities of the modulating Markov chain.

We then use $\pi_i(t)$ to determine the time-dependent diffusion coefficients:

$$\beta(t) = \lambda(t) - \mu, \qquad \alpha(t) = \lambda(t) C_A^2(t) + \mu C_B^2,$$

where

$$\lambda(t) = \frac{1}{E[A(t)]}, \qquad C_A^2(t) = \frac{E[A(t)^2]}{E[A(t)]^2} - 1.$$

In the case of a two-state Markov chain (an *ON–OFF* source), if r_j denotes the transition rate out of state j, $j = 1, 2$, then the probability of being in state 1 at time t is given by

$$\begin{aligned} \pi_1(t \mid \pi_1(0) = 1) &= \frac{r_1}{r_1 + r_2} \exp[-(r_1 + r_2)t] + \frac{r_2}{r_1 + r_2}, \\ \pi_1(t \mid \pi_1(0) = 0) &= \frac{r_2}{r_1 + r_2} \left[1 - \exp[-(r_1 + r_2)t]\right]. \end{aligned} \tag{8.50}$$

Here, $\pi_i(t \mid \pi_i(0) = 1)$ denotes the probability that the chain is in state i at time t, given that it started in the same state at time $t = 0$. Similarly,

$$\pi_2(t \mid \pi_2(0) = 1) = 1 - \pi_1(t \mid \pi_1(0) = 0),$$

and

$$\pi_2(t \mid \pi_2(0) = 0) = 1 - \pi_1(t \mid \pi_1(0) = 1).$$

The time scale over which correlation remains significant is determined by the autocorrelation function

$$R(t, \tau) = E[A(t)A(t + \tau)] - E[A(t)]^2,$$

which takes the form

$$R(t, \tau) = \sum_{i=1}^{M} \pi_i(t) \left[\pi_i(\tau \mid \pi_i(0) = 1)\lambda_i^2 + \sum_{\substack{j=1 \\ j \neq i}}^{M} \pi_j(\tau \mid \pi_i(0) = 1)\lambda_i \lambda_j \right]. \tag{8.51}$$

We may also consider a superposition of K independent *ON–OFF* models or, more generally, streams modulated by different Markov chains.

Let $\lambda^{(k)}(t)$ and ${C_A^{(k)}}^2(t)$ denote the parameters of the k-th *ON–OFF* source. The resulting input intensity $\lambda(t)$ is simply the sum of all $\lambda^{(k)}(t)$.

Since the number of customers arriving from the k-th source during a time interval of length t is approximately normally distributed with variance $\lambda^{(k)}(t){C_A^{(k)}}^2(t)$, and because the sum of independent normally distributed random variables is itself normally distributed with variance equal to the sum of the component variances, we obtain

$$C_A^2(t) = \sum_{k=1}^{K} \frac{\lambda^{(k)}(t)}{\lambda(t)} {C_A^{(k)}}^2(t). \tag{8.52}$$

The solution of the model with time-varying parameters is obtained, as before, by dividing the time axis into short intervals and determining the model parameters within each interval. After solving the diffusion equation for the transient state over a given interval, the solution at the end of that interval serves as the initial condition (the function ψ) for the next interval, in which different model parameters apply. Fig. 8.41 presents some numerical results indicating good performance of diffusion approximation.

Example 8.9

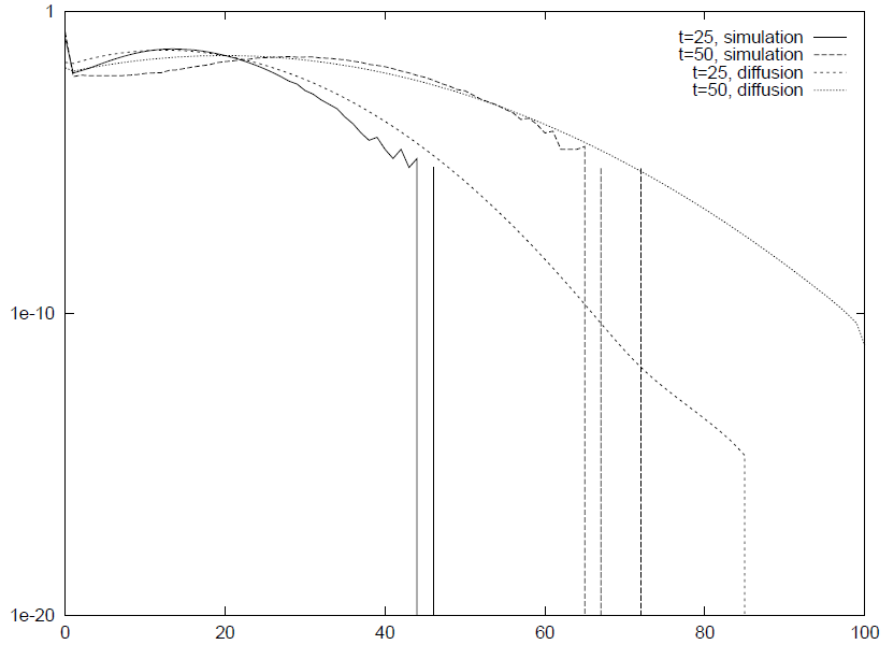


Figure 8.41: Queue distribution at $t = 25$ and $t = 50$ – comparison of diffusion approximation and simulation (150 000 repetitions); the source has two phases with $\lambda_1 = 1.6$ and $\lambda_2 = 0.4$; at $t = 0$ the queue is empty; computation each $\Delta = 1$

Let us take as the unit of time the time required to transmit a single cell. For example, for a channel with a throughput of 155 Mbps, this unit is equal to $2.7 \cdot 10^{-6}$ *s. All cells are assumed to have the same priority.*

Cells arrive at a switch represented by a standard FIFO queue with finite capacity equal to 100 cells. The service time is constant. Consider the following two cases.

(a) The source alternates between two phases: a high-activity phase and a low-activity phase. The durations of both phases are exponentially distributed, with average duration equal to 100 time units: $r_1 = r_2 = 0.01$, *cf. equation (8.50). The generated cell stream is Poisson with intensity* $\lambda_1 = 1.6$ *cells per unit time when the system is in the first phase, and* $\lambda_2 = 0.4$ *cells per unit time (case 1) or* $\lambda_2 = 0$ *(case 2) when the system is in the second phase. At the initial time* $t = 0$, *the system is in the first phase.*

(b) Source with multiple phases: we consider a commonly used model of cell traffic generated by moving-image transmission, introduced in [77] and later used, for example, in [72, 7], to model statistically multiplexed video sources.

The aggregate process is represented by a continuous-time Markov chain with 31 states, shown in Fig. 8.44. The transition rate for leaving state j *is*

$$r_j = 1.03077(31 - j) + 2.5923j,$$

and the transition probabilities are

$$p_{j,j+1} = \frac{1.03077(31-j)}{1.03077(31-j) + 2.5923j}, \qquad \begin{array}{l} p_{j,j-1} = 1 - p_{j,j+1}, \\ p_{1,2} = p_{31,30} = 1. \end{array} \qquad 2 \le j \le 30,$$

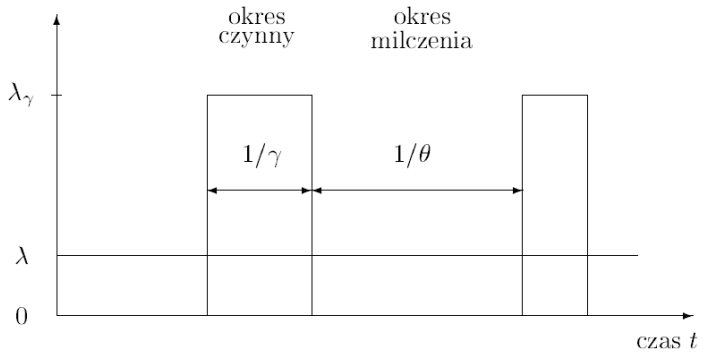


Figure 8.42: Active and silent periods of cell generation

When the Markov chain is in state j, the traffic intensity generated by the source is

$$\lambda_j = 2.391(j-1).$$

In the original formulas, taken from [72], the unit of time is one millisecond. In the present example, the coefficient values have been rescaled to match the previously selected unit of time.

Results for case (a).

Figure 8.45 shows, as a function of time, the average queue length predicted by the diffusion model, compared with the corresponding simulation results.

In the simulation model, the operation of the system, with the same structure and the same probability distributions, was represented using discrete-event simulation. The queue-length trajectories were generated 10,000 times and averaged over the same time horizon.

The observed discrepancies are most likely due to the accumulation of numerical errors, since calculations in a given time interval depend on the results obtained in the previous interval. These discrepancies could be reduced by selecting shorter intervals over which the diffusion-process parameters are assumed constant.

It can be seen that an *ON–OFF* system with exponentially distributed phase durations produces correlated traffic only over a time horizon comparable to a single *ON–OFF* cycle.

Results for case (b).

Figure 8.46 illustrates the changes in queue length as a function of time and of the initial state j_0 of the Markov chain.

In Fig. 8.47, the results for constant service times are compared with the results obtained for exponentially distributed service times. This comparison confirms the conclusions of [72], namely that the effect of the service-time distribution is negligible compared with the impact of a variable and correlated input source.

Example 8.10

Cell-stream model

We assume that each source alternates between active and inactive periods, both exponentially distributed with parameters γ and θ, respectively, see Fig. 8.42.

During active periods, the source emits cells with intensity λ_γ; during inactive periods, it remains silent. The activity of the source is therefore controlled by a two-state Markov process

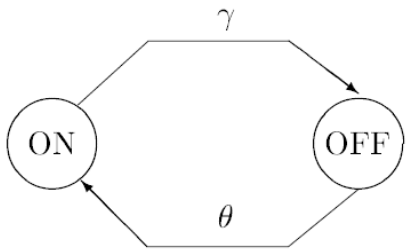


Figure 8.43: Markov process controlling $ON - OFF$ source activity

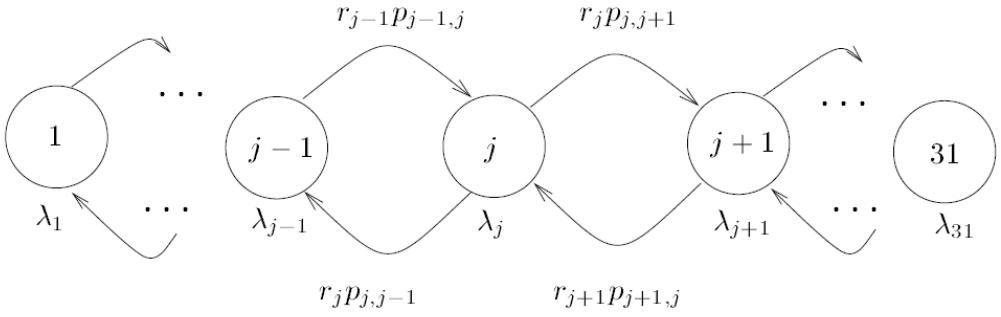


Figure 8.44: Markov chain used in the example to model a correlated cell stream

that alternates between the active and inactive states, see Fig. 8.43.

Taking both active and inactive periods into account, the distribution of the interarrival times between consecutive cells takes the form [44]

$$F_X(x) = \left[\left(1 - \frac{\gamma}{\lambda_\gamma}\right) + \frac{\gamma}{\lambda_\gamma}\left(1 - e^{-\theta(x-1/\lambda_\gamma)}\right)\right] \mathbf{1}(x - 1/\lambda_\gamma), \tag{8.53}$$

where $\mathbf{1}(x)$ denotes the unit-step function.

From this distribution, one can determine the average arrival rate of cells

$$\lambda = E[X]^{-1} = \frac{\theta}{\gamma + \theta}\lambda_\gamma \tag{8.54}$$

and the squared coefficient of variation of the interarrival-time distribution

$$C_X^2 = \frac{\mathrm{Var}[X]}{E[X]^2} = \frac{1 - (1 - \gamma\lambda_\gamma)^2}{(\gamma\lambda_\gamma + \theta\lambda_\gamma)^2}. \tag{8.55}$$

The stream of cells arriving at the buffer is a superposition of many such independent streams. We assume that there are K streams with parameters $\lambda^{(k)}$ and ${C^{(k)}}^2$.

Let us assume that the link bandwidth is 150 Mbit/s. The transmission time of a 53-byte ATM cell is therefore

$$D^{-1} = \frac{53 \times 8}{150 \times 10^6}$$

seconds, which corresponds to a service rate

$$D = 0.3537 \times 10^6$$

cells per second.

We further assume that the traffic is generated by a superposition of two types of sources:

- Type A sources have active periods with an average duration of 10 ms and silent periods with an average duration of 90 ms. During an active period, the source emits cells at the maximum link rate,

  $$\lambda_A = D = 0.3537 \times 10^6$$

  cells per second.

  The average intensity of traffic generated by a type A source is

  $$\lambda^A = \frac{0.01}{0.01 + 0.09}\lambda_A = 0.1\lambda_A.$$

  Using equation (8.55), we obtain

  $${C_A^{(A)}}^2 = 5731.67.$$

- Type B sources have active periods with an average duration of 10 ms and silent periods with an average duration of 40 ms. During an active period, the source also emits cells at the maximum rate

  $$\lambda_B = D = 0.3537 \times 10^6$$

cells per second.

The average intensity of traffic generated by a type B source is

$$\lambda^B = \frac{0.01}{0.01 + 0.04}\lambda_B = 0.2\lambda_B.$$

The corresponding squared coefficient of variation is

$$C_A^{(B)^2} = 530.10.$$

Initially, the multiplexer receives traffic from three type A sources and one type B source. At time $t = t_1$, the offered load increases, and the traffic originates from four type A sources and two type B sources.

For $t < t_1$, the diffusion-process parameters are

$$\alpha_1 = 3\lambda^A C_A^{(A)^2} + \lambda^B C_A^{(B)^2} = 645\,687\,456,$$

$$\beta_1 = 3\lambda^A + \lambda^B - D = -360\,150.$$

For $t \geq t_1$, the parameters become

$$\alpha_2 = 4\lambda^A C_A^{(A)^2} + 2\lambda^B C_A^{(B)^2} = 885\,916\,289,$$

$$\beta_2 = 4\lambda^A + 2\lambda^B - D = -70\,740.$$

The coefficients α_1, β_1 and α_2, β_2 determine the density function of the corresponding diffusion process. For example, one may assume that before t_1 the system operates in steady state, and that the steady-state solution (7.32) with parameters α_1, β_1 serves as the initial condition $\psi(x)$ for the transient solution (7.37) after the change at $t = t_1$.

The transformed probability $\bar{p}_N(s)$, whose numerical inversion gives $p_N(t)$, can then be used to estimate the time-varying cell-loss probability $L(t)$ for $t > t_1$, according to equation (7.47).

8.5 Traffic within the network

8.5.1 Changes in packet flow along the virtual path

The model presented below is described in more detail in [24].

Figure 8.48 illustrates the model under consideration, consisting of M stations connected in series. Each station represents a network node forming part of a virtual path, through which packets are transmitted from the source point A to the destination point B.

Each station receives two traffic streams: the cells (packets) of the virtual circuit itself and the local traffic generated at the node. Each of these streams consists of two classes of tasks, corresponding to priority cells (class 1) and ordinary cells (class 2).

Let $\lambda_i^{(vc,k)}$ denote the intensity (measured as the number of packets transmitted per unit time) of class-k packets, $k = 1, 2$, belonging to the virtual circuit and entering station i, $i = 1, \ldots, M$. Similarly, let $\lambda_i^{(l,k)}$ denote the intensity of local traffic of class k at station i.

The superscripts in parentheses denote packet classes, whereas the subscripts indicate the station number.

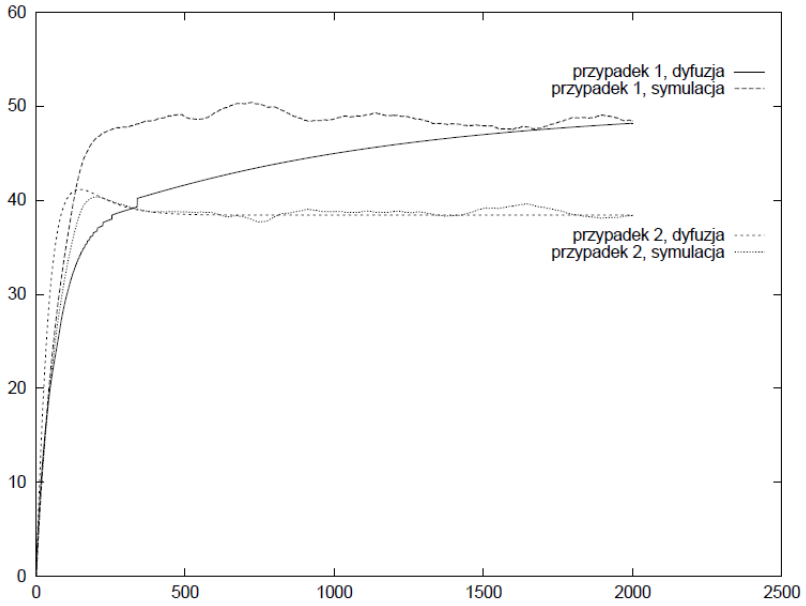


Figure 8.45: Time-dependent average queue length in a network switch; *ON–OFF* source with phase durations defined by the parameter $1/\alpha = 100$; $\lambda_1 = 1.6$, $\lambda_2 = 0.4$ (average $\lambda = 1.0$, case 1) or $\lambda_2 = 0$ (average $\lambda = 0.8$, case 2); constant service time; results of simulation and diffusion approximation

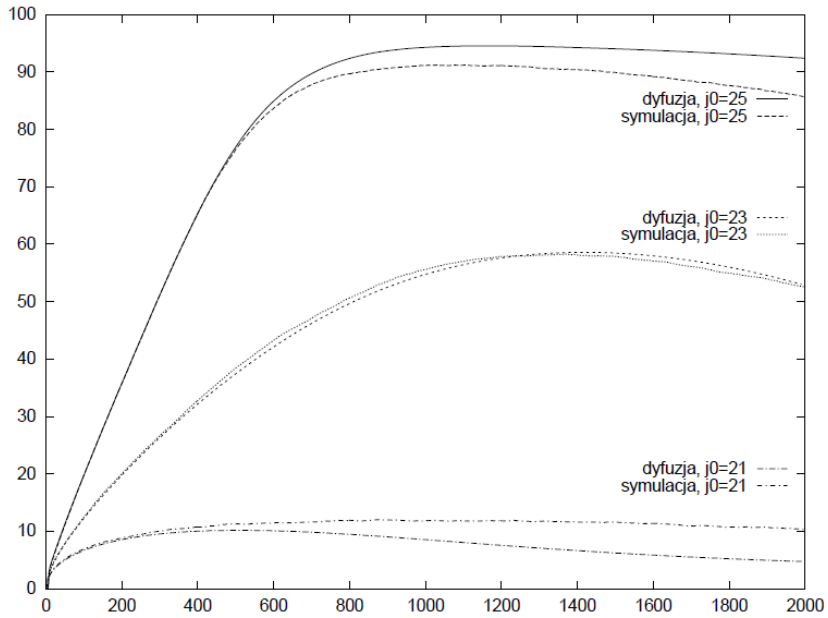


Figure 8.46: Time-dependent average queue length in the switch; the source is controlled by a 31-phase Markov chain, with initial phase $j_0 = 21, 23, 25$; constant service time; results of the diffusion approximation and simulation

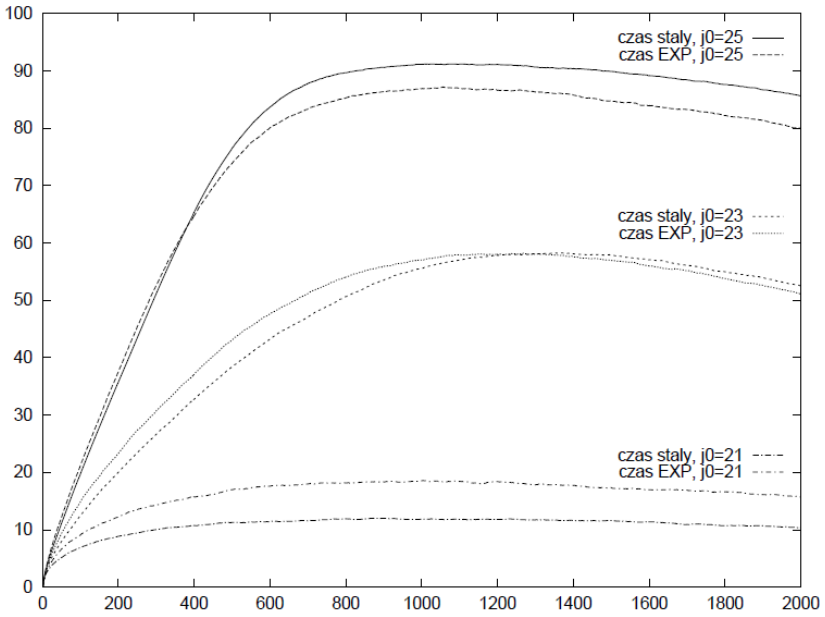


Figure 8.47: Time-dependent average queue length in the switch; the source is controlled by a 31-phase Markov chain, with initial phase $j_0 = 21, 23, 25$; comparison of exponential and constant service times

At the first station, the values of $\lambda_1^{(vc,k)}$ and ${C_{A1}^{(vc,k)}}^2$ are given. For subsequent stations, these values must be calculated. The local-traffic parameters $\lambda_i^{(l,k)}$ and ${C_{Ai}^{(l,k)}}^2$ are assumed to be known for every station.

The coefficient C_A^2 denotes the squared coefficient of variation of the distribution of inter-arrival times. Service times are constant and identical for all tasks in the queue:

$$\frac{1}{\mu_i^{(vc,k)}} = \frac{1}{\mu_i^{(l,k)}} = \frac{1}{\mu}, \tag{8.56}$$

$${C_{Bi}^{(vc,k)}}^2 = {C_{Bi}^{(l,k)}}^2 = 0, \qquad k = 1, 2, \qquad i = 1, \ldots, M. \tag{8.57}$$

For the virtual-path problem, we assume that

$${C_{Ai}^{(vc,k)}}^2 = {C_{D(i-1)}^{(vc,k)}}^2, \qquad i = 2, \ldots, M.$$

The symbol A refers to arrivals, B to service, and D to departures.

In the steady state, the virtual-circuit flow rate is the same at all stations:

$$\lambda_i^{(vc,k)} = \lambda_1^{(vc,k)}, \qquad i = 2, \ldots, M.$$

In the transient state, the parameters λ_i, C_{Ai}^2, and C_{Di}^2 vary with time. We must therefore distinguish between the input intensity $\lambda_{i,\text{in}}(t)$ and the output intensity $\lambda_{i,\text{out}}(t)$ of station i.

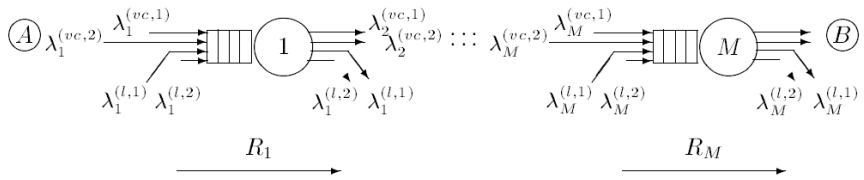


Figure 8.48: Virtual-path model

The interdeparture times from node i are described by a distribution with density $d_i(t)$, for which the output-stream parameters satisfy

$$E[d_i(t)] = \frac{1}{\lambda_{i,\text{out}}(t)} = \frac{1}{\lambda_{i,\text{in}}(t)}[1 - \varrho_i(t)] + \frac{1}{\mu}, \tag{8.58}$$

$$C^2_{Di}(t) = C^2_{Ai}(t)[1 - \varrho_i(t)] + \varrho_i(t)[1 - \varrho_i(t)], \tag{8.59}$$

where

$$\varrho_i(t) = 1 - p_{i0}(t),$$

and $C^2_{Di}(t)$ is the squared coefficient of variation of the interdeparture-time distribution $d_i(t)$.

At the moment when the input parameters change, the composition of the queue is still determined by the previous parameters. Thus, for some time, the probability that a departing customer belongs to class k depends on the old values of $\lambda_i^{(k)}$ and λ_i, combined with the new value of ϱ_i.

When the service time is constant and equal to $1/\mu$, this delay is x/μ, where x is the number of customers in the queue. The density function of the corresponding delay distribution has the same form as the queue-length density f at the time $t = t_{ch}$ when the input parameters change:

$$q(t) = f(x, t_{ch}; x_0)\delta\left(t - \frac{x}{\mu}\right) = \frac{1}{\mu} f(t\mu, t_{ch}; x_0). \tag{8.60}$$

Our algorithm assumes that the parameters change only at the beginning of each successive interval Δt, while remaining constant within the interval.

Let the value of a parameter, for example λ_i, during the j-th time interval, i.e. for

$$t \in [j\Delta t, (j+1)\Delta t),$$

be denoted by $\lambda_i(j)$.

Let $\theta^{(k)}_{i,\text{in}}(j)$ denote the proportion of class-k customers in the input stream to station i during the j-th time interval:

$$\theta^{(k)}_{i,\text{in}}(j) = \frac{\lambda^{(k)}_{i,\text{in}}(j)}{\lambda_{i,\text{in}}(j)}.$$

The composition of the output stream of station i during interval j can then be expressed

as

$$\theta_{i,\text{out}}^{(k)}(j) = \sum_{l=2}^{h} P[n_i > (l-1)\mu\Delta t; j-l+1]\,\theta_{i,\text{in}}^{(k)}(j-l) \prod_{m=l}^{h-1} P[n_i \le m\mu\Delta t; m] + \theta_{i,\text{in}}^{(k)}(j-1) \prod_{m=l}^{h-1} P[n_i \le m\mu\Delta t; m], \tag{8.61}$$

where

$$\prod_{m=a}^{b}(\cdot) \equiv 1, \qquad \text{if } b < a,$$

$$h = \frac{N}{\mu\Delta t},$$

N is the queue capacity, and $\mu\Delta t$ is the number of customers that can be served during the interval Δt.

For simplicity, we assume that N is an integer multiple of $\mu\Delta t$.

Example 8.11

Let us assume the following numerical parameters. The virtual link consists of $M = 4$ nodes, and the transmission time of a cell is equal to one unit of time, i.e. $\frac{1}{\mu} = 1$.

We assume that the local traffic parameters are constant and identical for all stations:

$$\lambda_i^{(l,1)} = \lambda_i^{(l,2)} = 0.25, \qquad {C_{Ai}^{(l,1)}}^2 = {C_{Ai}^{(l,2)}}^2 = 1.$$

The flow of priority cells entering the first link of the virtual path is time-dependent: during periods of low activity,

$$\lambda_1^{(vc,1)} = 0.05, \qquad {C_{Ai}^{(vc,1)}}^2 = 1,$$

and during periods of high activity,

$$\lambda_1^{(vc,1)} = 0.50, \qquad {C_{Ai}^{(vc,1)}}^2 = 0.50.$$

The high-activity period lasts 20 time units, while each low-activity period lasts 80 time units.

The stream of lower-priority cells in the virtual path has parameters independent of time:

$$\lambda_1^{(vc,2)} = 0.05, \qquad {C_{Ai}^{(vc,2)}}^2 = 1.$$

Figure 8.49 shows the traffic intensity $\lambda_{1,\text{in}}(t)$ at the input of the first station and the output flows $\lambda_{i,\text{out}}(t)$, $i = 1, \ldots, 4$, for successive stations.

We assume that at $t = 0$ the system is in a steady state corresponding to the low-activity period. The duration of intervals in which the model parameters are constant is taken as $\Delta t = 5$ time units. Propagation delays are neglected.

During the initial phase of high activity, differences between the output intensities $\lambda_{i,\text{out}}(t)$ are visible, but these differences gradually disappear over time.

Figure **??** presents the evolution of the average queue lengths $E[N_i(t)]$ corresponding to the assumed dynamics of the input streams.

Figure 8.50 shows the coefficients of variation of the response times for individual stations. It can be observed that the first two stations ($i = 1, 2$) exhibit different characteristics, in which the influence of the input flow of the virtual path is clearly visible. For subsequent stations, this influence becomes negligible, and the behaviour of the stations becomes practically identical, provided that their local traffic is the same.

For m stations connected in series and having identical coefficients C_R^2, the coefficient of variation of the total response time is given by

$$C^2_{R_1 \ldots R_m} = \frac{1}{m} C_R^2.$$

Figures 8.49–**??** refer to the total traffic flow.

Figure 8.52 presents the output flows $\lambda_{i,\text{out}}^{(vc,1)}(t)$, $\lambda_{i,\text{out}}^{(vc,2)}(t)$ of priority and ordinary cells in the virtual path, as well as $\lambda_{i,\text{out}}^{(l,1)}(t)$, $\lambda_{i,\text{out}}^{(l,2)}(t)$ for local streams, compared with the total intensity $\lambda_i(t)$.

Figure 8.53 shows, in an enlarged view, the evolution of the flow $\lambda_{i,\text{out}}^{(vc,2)}(t)$.

At the beginning of a high-activity period, the flow $\lambda_{i,\text{out}}^{(vc,1)}(t)$ increases proportionally to $\varrho_i(t)$ (i.e. to $\lambda_i(t)$), because the composition of serviced cells is still determined by the previous (low-activity) parameters.

Later, the region with an increased proportion of priority cells moves towards the head of the queue, and the output stream $\lambda_{i,\text{out}}^{(vc,1)}(t)$ increases significantly.

Figure 8.54 shows the loss factors $L^{(1)}(t)$ and $L^{(2)}(t)$, defined as the ratio of the number of lost cells per unit time to the arrival intensity of cells of a given class, assuming the push-out policy and a buffer size of $N = 15$.

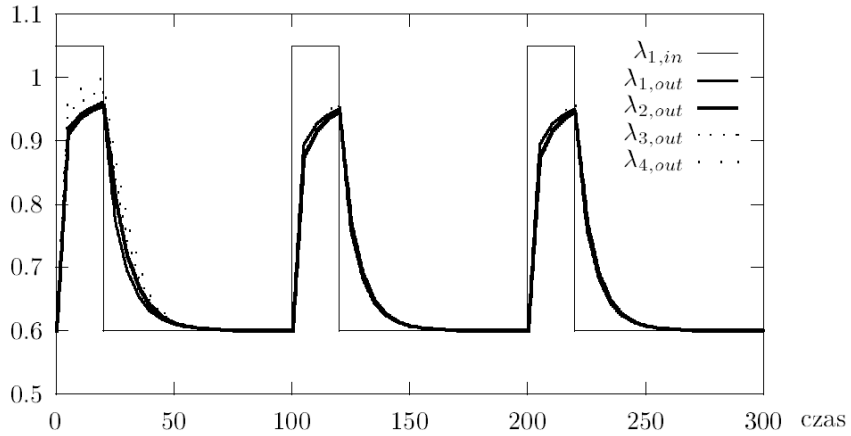


Figure 8.49: Intensity of the input stream at the first node and the output intensities at subsequent nodes of the virtual path

■

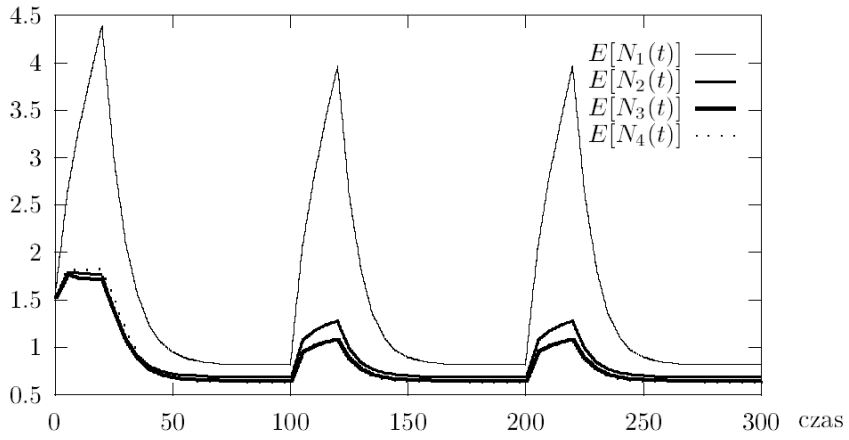


Figure 8.50: Average number of cells $E[N_i(t)]$ in nodes i of the virtual path

8.5.2 Irregularities in packet transmission times through the network

Packets are generated by their source at regular time intervals. However, while traversing the network, their rhythm becomes disturbed due to random waiting times at intermediate nodes. This phenomenon is referred to as *jitter*: packets sometimes arrive too close together and sometimes too far apart.

Solving the diffusion equation and obtaining the function $f_i(x)$ for station i yields not only an approximation of the number of customers present at the station. Since the service time is constant, it also provides an approximation of the response time R_i, i.e. the time spent by a cell at station i. Its density function $r_i(y)$ is given by

$$r_i(y,t;\psi) = \mu f_i[\mu(y-1),t;\psi]. \tag{8.62}$$

The argument of the function is shifted by 1 in order to include the service time of the tagged customer.

The total transit time through m consecutive stations of the virtual path may be estimated as an m-fold convolution of the functions $r_i(y,t;\psi)$:

$$r_{1\ldots m}(y) = r_1(y) * \cdots * r_m(y), \qquad m = 2,\ldots,M. \tag{8.63}$$

Thus, the response-time density depends directly on the queue-length density through

$$r_i(y,t;\psi) = \mu f_i(\mu y - 1,t;\psi).$$

The density function $r_{1\ldots m}(y,t;\psi)$ of the total transit time through m stations can therefore be determined in the same way as in (8.63), namely by repeated convolution of the functions $r_i(y,t;\psi)$.

Example 8.12

Let us assume that $M = 5$ stations form a virtual network. The buffer size at every station is $N = 80$. The service time is constant and equal to $1/\mu = 1$.

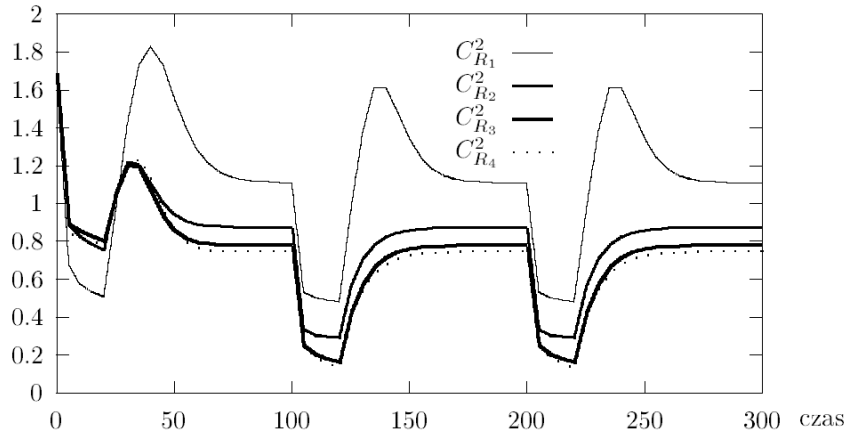


Figure 8.51: Coefficient of variation of response times for individual nodes i in the virtual path

The results determining the transmission delay, in the form of the distribution of the time required to pass through all stations in the network, calculated according to (8.62), are shown in Figs. 8.55 and 8.56.

The function $r(y)$ shown in Fig. 8.55 is calculated for the following parameters of the traffic streams at the five stations forming the virtual path. Parameters marked with the superscript (v) refer to the virtual-path traffic, while parameters marked with the superscript (l) refer to local traffic, transmitted together with the virtual-path traffic through only a single node, whose number is indicated by the subscript:

$$\lambda_1^{(v)} = 0.5, \qquad {C_{A1}^{(v)}}^2 = 1.0, \qquad \lambda_1^{(l)} = 0.3, \qquad {C_{A1}^{(l)}}^2 = 1.5,$$

$$\lambda_2^{(l)} = 0.3, \qquad {C_{A2}^{(l)}}^2 = 1.0, \qquad \lambda_3^{(l)} = 0.4, \qquad {C_{A3}^{(l)}}^2 = 0.5,$$

$$\lambda_4^{(l)} = 0.45, \qquad {C_{A4}^{(l)}}^2 = 1.5, \qquad \lambda_5^{(l)} = 0.2, \qquad {C_{A5}^{(l)}}^2 = 2.0.$$

The simulation results shown in the same figure represent the histogram of transit times for 200 000 cells.

Figure 8.56 illustrates the influence of the coefficient of variation ${C_{A1}^{(v)}}^2$ on the transit time. The three curves shown correspond to

$${C_{A1}^{(v)}}^2 = 1.0, \qquad {C_{A1}^{(v)}}^2 = 5.0, \qquad {C_{A1}^{(v)}}^2 = 10.0.$$

Compensation for traffic irregularities

Irregularities in the arrival times of cells at their destination may be reduced by introducing an additional station at the end of the path, immediately before the receiver. This station collects cells and then transmits them onward at regular intervals.

The operation of this station consists of two alternating periods. During the period T_1, cells are accumulated in the queue. During the period T_2, they are transmitted onward at regular intervals. The station buffer has finite capacity and can store at most N packets.

Using the diffusion approximation, we wish to determine:

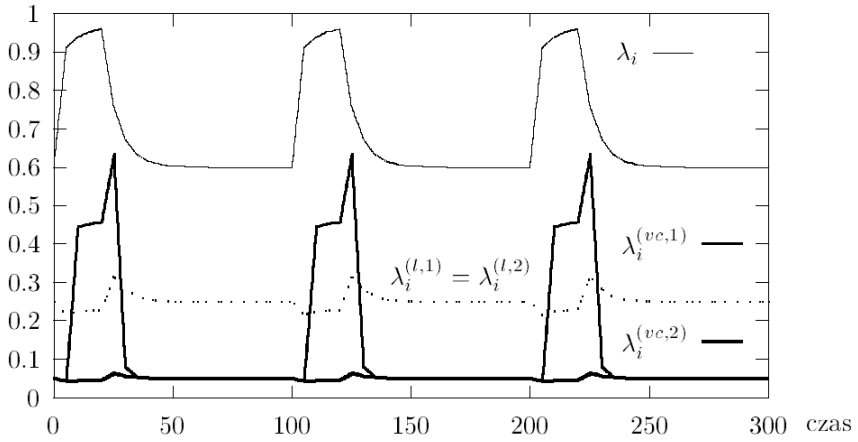


Figure 8.52: Output intensities $\lambda_{i,\text{out}}^{(vc,1)}(t)$ and $\lambda_{i,\text{out}}^{(vc,2)}(t)$ for priority and ordinary cells in the virtual path, compared with the total intensity $\lambda_i(t)$

- the probability that the buffer capacity is exceeded during T_1, resulting in packet loss,
- the probability that the number of packets drops to zero during T_2, thus interrupting the regular output flow.

Let λ and C_A^2 denote the parameters of the input stream.

During T_1, cells accumulate in the buffer, so the distribution of the number of packets $N(t)$ is approximated by a diffusion process $X_1(t)$ with parameters

$$\beta = \lambda, \qquad \alpha = \lambda C_A^2,$$

starting at $t = 0$ from $x = 0$, with an absorbing barrier at $x = N$. Once the process reaches $x = N$, it remains there until the end of period T_1, just as a queue remains full after overflow.

The density function of the process is [18]

$$p_1(x,t) = \frac{1}{\sqrt{2\alpha\Pi t}} \left[\exp\left(-\frac{(x-\beta t)^2}{2\alpha t} \right) - \exp\left(\frac{2\beta N}{\alpha} - \frac{(x-2N-\beta t)^2}{2\alpha t} \right) \right], \qquad 0 \le x < N.$$

At the end of period T_1, the probability that the queue is full is

$$p_1(N, T_1) = \int_0^{T_1} g(t)\, dt, \tag{8.64}$$

where $g(t)$ is the density of the first-passage time from $x_0 = 0$ to $x = N$:

$$g(t) = -\frac{d}{dt} \int_{-\infty}^{N} p_1(x,t)\, dx = \frac{N}{\sqrt{2\alpha\Pi t^3}} \exp\left(-\frac{(N-\beta t)^2}{2\alpha t} \right). \tag{8.65}$$

During the period T_2, the station continues to accept new cells arriving from the network with the same parameters as before, while transmitting them at fixed intervals Δ.

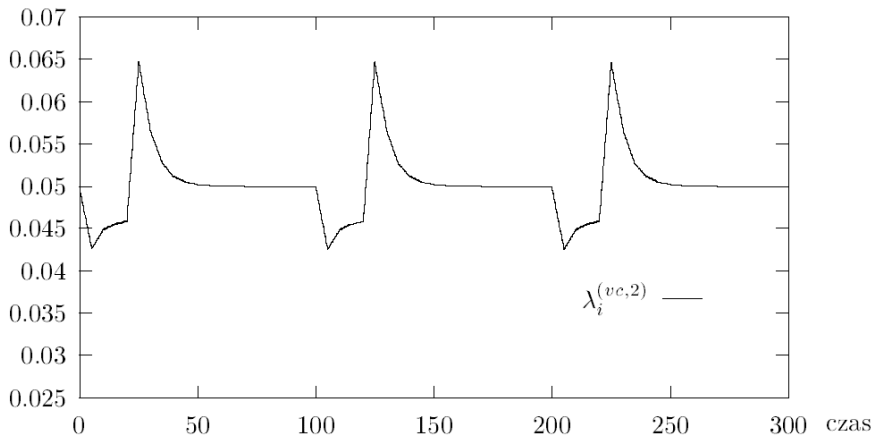


Figure 8.53: Output intensity $\lambda_{i,\text{out}}^{(vc,2)}(t)$ for lower-priority cells in the virtual path

We approximate the number of cells during T_2 using the diffusion model of a $G/G/1/N$ queue, with the initial condition $\psi(x)$ defined by

$$\psi(x) = \begin{cases} p_1(x, T_1) & \text{for} \quad 0 \le x < N, \\ p_1(N, T_1) & \text{for} \quad x = N. \end{cases}$$

When selecting the diffusion parameters, the constant service time must be taken into account:

$$\frac{1}{\mu} = \Delta, \qquad C_B^2 = 0.$$

To obtain the probability density that the process reaches the lower barrier for the first time at time t, we consider a diffusion process with an upper barrier at $x = N$ and returns from $x = N$ to $x = N - 1$.

The equations for diffusion with return processes are modified to

$$f(x,t;\psi) = \phi(x,t;\psi) + \int_0^t g_{N-1}(\tau)\phi(x, t-\tau; N-1)\, d\tau, \tag{8.66}$$

$$\begin{aligned} \gamma_N(t) &= p_1(N,0)\delta(t) + [1 - p_1(0,0) - p_1(N,0)]\gamma_{\psi,N}(t) \\ &\quad + \int_0^t g_{N-1}(\tau)\gamma_{N-1,N}(t-\tau)\, d\tau, \end{aligned} \tag{8.67}$$

$$\begin{aligned} \gamma_0(t) &= p_1(0,0)\delta(t) + [1 - p_1(0,0) - p_1(N,0)]\gamma_{\psi,0}(t) \\ &\quad + \int_0^t g_{N-1}(\tau)\gamma_{N-1,0}(t-\tau)\, d\tau. \end{aligned} \tag{8.68}$$

Using these equations, we determine the probability $p_2(0,t)$ that by time t the buffer has become empty at least once, thus interrupting the regular output rhythm:

$$p_2(0,t) = \int_0^t \gamma_0(\tau)\, d\tau. \tag{8.69}$$

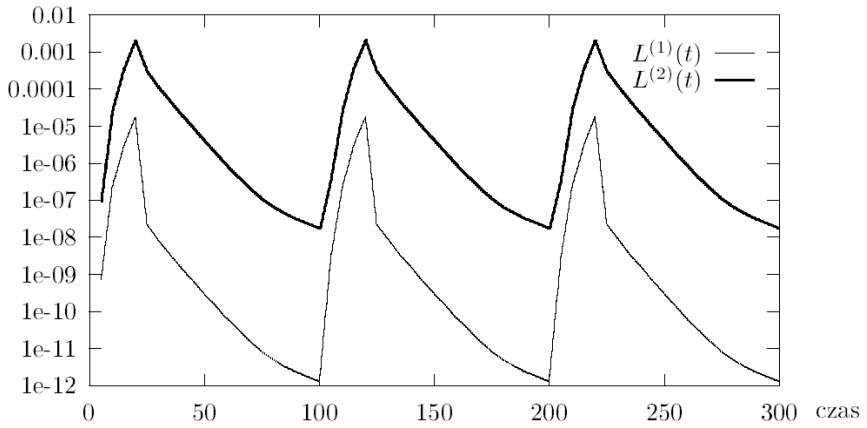


Figure 8.54: Loss factors $L^{(1)}(t)$ and $L^{(2)}(t)$ for both classes in the first section of the virtual path, during periods of low and high traffic intensity

The packet loss probabilities associated with the finite buffer size are

$$L_i = p_i(N, T_i), \qquad i = 1, 2.$$

Here, L_1 corresponds to losses during T_1, while L_2 corresponds to losses during T_2.

The probability that the system never becomes empty during T_2 is therefore

$$1 - p_2(0, T_2),$$

obtained from (8.69).

Example 8.13

Let us assume that the size of the compensating buffer is $N = 30$, and that the two alternating periods of its operation described above have lengths $T_1 = 100$ and $T_2 = 500$ time units, respectively. During the period T_2, packets leave the buffer at regular intervals $\Delta = 1$ unit of time. We wish to analyse the behaviour of an initially empty buffer over a single cycle consisting of the periods T_1 and T_2. The parameters of the input stream are given in the captions to the figures below.

Figs. 8.57, 8.58, and **??** refer to the period T_1.

Fig. 8.57 shows the queue-length distribution $p_1(x, t)$ as a function of time. The results of the diffusion model are compared with simulation results, obtained by averaging 50,000 independent transient simulation runs.

Fig. 8.58 shows the influence of the coefficient of variation C_A^2 of the input stream on the queue-length distribution at a selected instant within the period T_1.

Fig. **??** shows the probability $p_1(N, t)$ that the buffer becomes full by time $t \in [0, T_1]$, as a function of time t and the coefficient C_A^2. The curve corresponding to $C_A^2 = 1$ is compared with simulation results.

Fig. 8.60 presents probability L_2 of cell loss as a function of time $t \in T_2$ for $\lambda = 0.9$; results of the diffusion approximation and simulation.

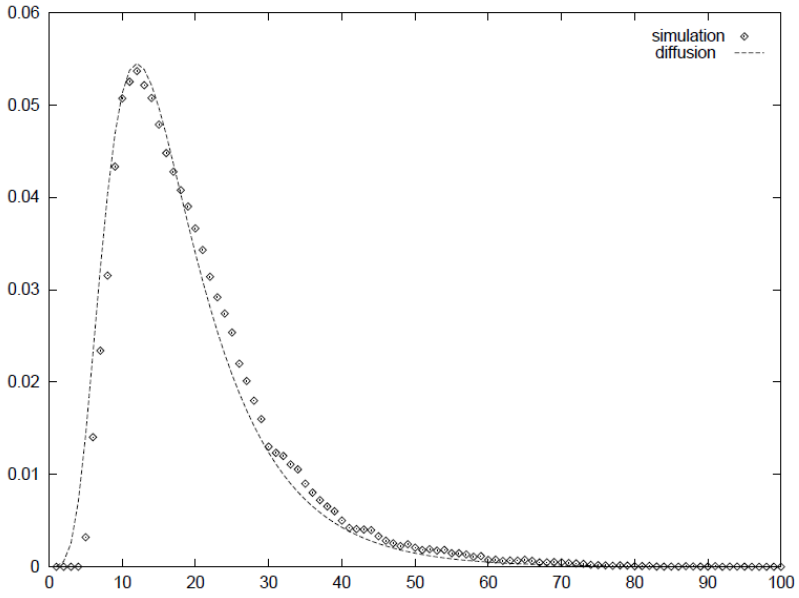


Figure 8.55: Distribution $r(y)$ of transmission time through the network; results of the diffusion approximation and simulation

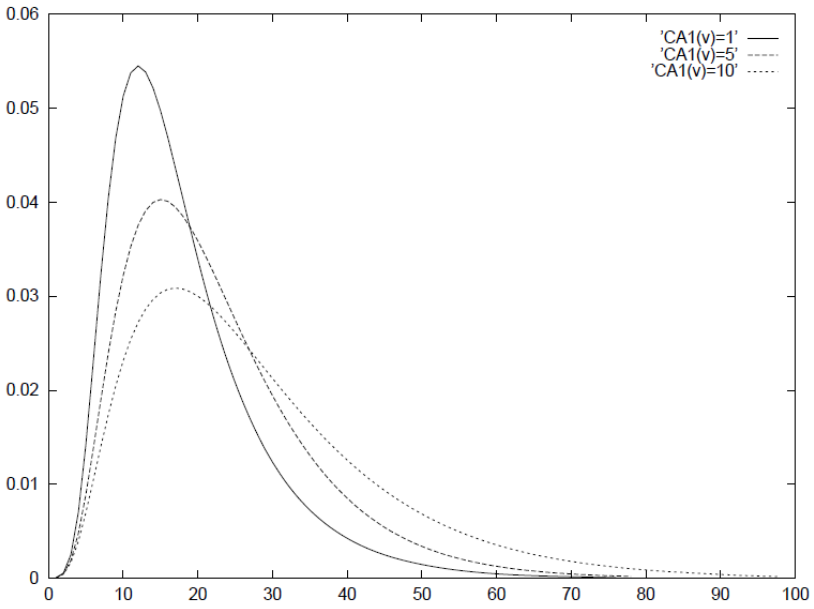


Figure 8.56: Distribution $R(y)$ of transmission time through the network as a function of $C_{A1}^{(v)^2}$

Fig. 8.61 shows the probability $p_2(0,t)$ that the buffer becomes empty for the first time during the period T_2.

The parameters chosen in this example are extreme: during the period T_1, the queue overflows, whereas during the period T_2 it almost empties completely.

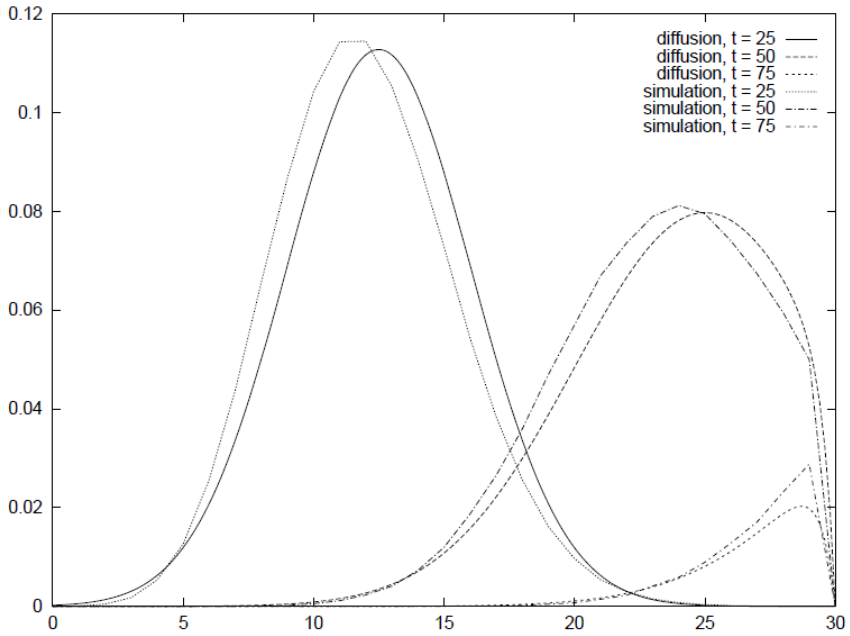


Figure 8.57: Queue-length distribution $p_1(x,t)$ during the period T_1 as a function of time; $\lambda = 0.5$, $C_A^2 = 1$, $N = 30$

■

8.6 Reactive control models

The *ECN* (*Explicit Congestion Notification*) mechanism, included among reactive traffic-control functions, operates according to the principle of feedback: when a network node reaches a prescribed congestion level (for example, a given queue length or a given number of packet losses over a specified time interval), the node sends a congestion notification to the source, which is then required to reduce its transmission intensity [118].

After a period during which no such notifications are generated, the source is again allowed to increase the rate at which it transmits information.

The technical implementation of this signalling may take one of two forms: either special control packets are sent directly from the congested node, or the node modifies a bit in the headers of packets that continue downstream (the so-called FECN, *forward ECN* bit). This serves as a message to the destination node, which, when acknowledging the received packets, informs the source that congestion has occurred along the path. The second method is simpler, but introduces a longer signalling delay.

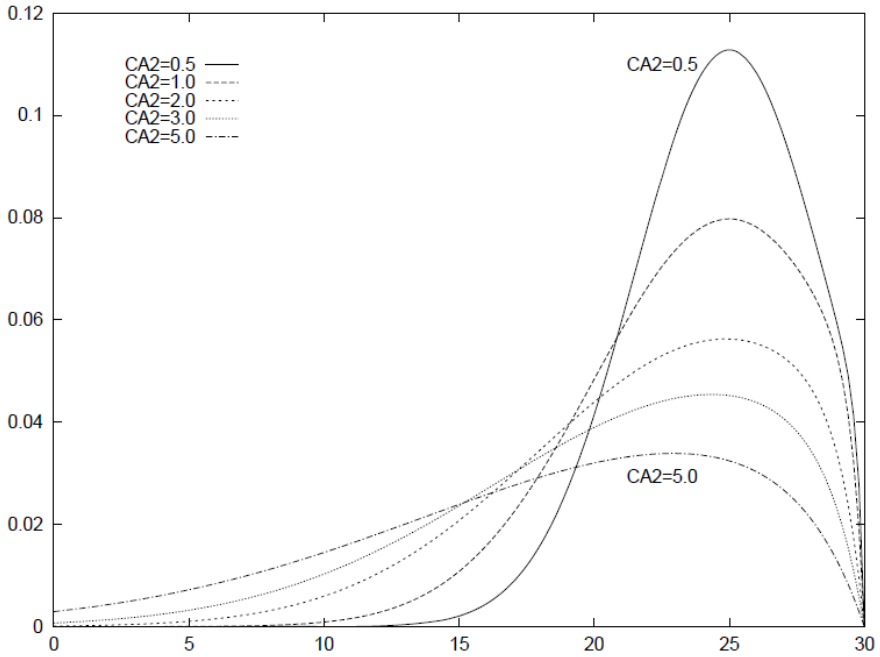


Figure 8.58: Queue-length distribution $p_1(x,t)$ for $t = 50$ time units as a function of C_A^2; $\lambda = 0.5$, $N = 30$

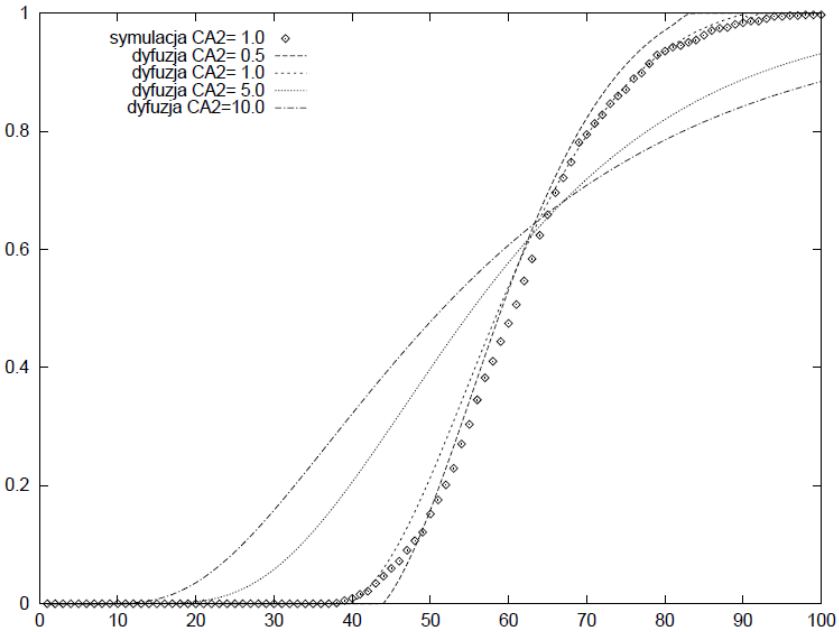


Figure 8.59: Probability $p_1(N,t)$ that the queue becomes full by time $t \in [0, T_1]$, as a function of time t and the coefficient of variation C_A^2 of the input flow; $\lambda = 0.5$, $N = 30$

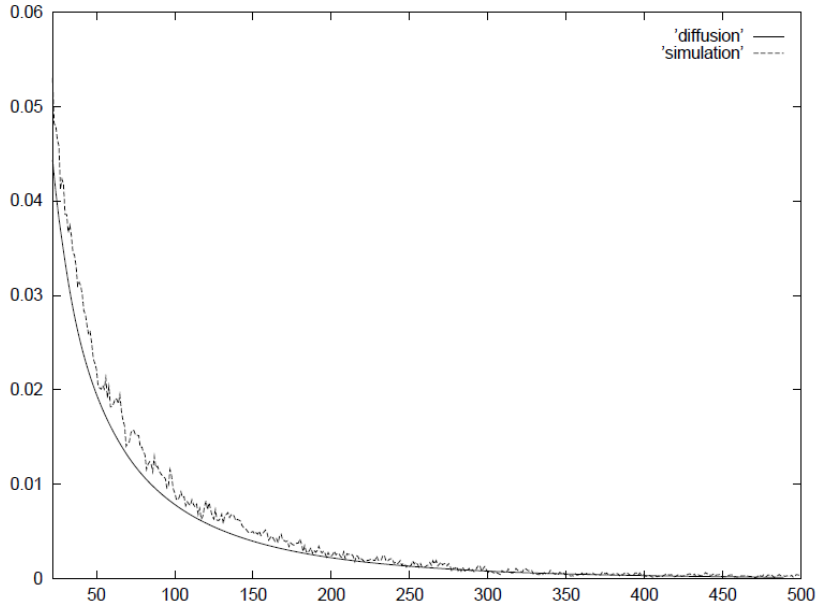


Figure 8.60: Probability L_2 of cell loss as a function of time $t \in T_2$ for $\lambda = 0.9$; results of the diffusion approximation and simulation

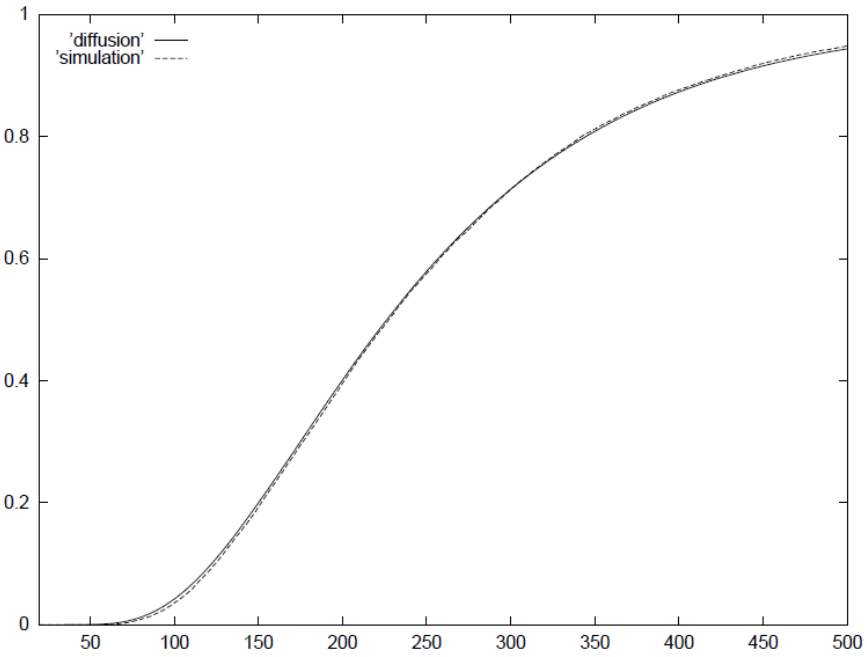


Figure 8.61: Probability $p_2(0,t)$ that the queue becomes empty at least once by time $t \in [0, T_2]$, as a function of time t; $\lambda = 0.9$, $N = 30$, $C_A^2 = 1$

The ECN mechanism must therefore be taken into account in the diffusion model.

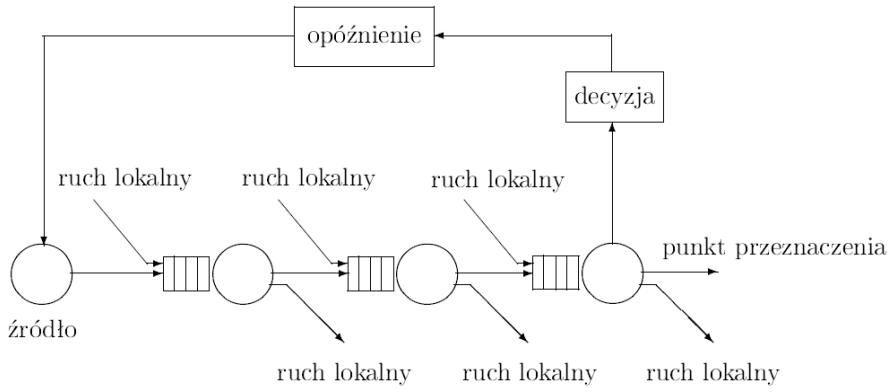


Figure 8.62: Queueing model with congestion notification of types FN and BN

A virtual-queue model with a feedback mechanism is shown in Fig. 8.62. The number of service stations is arbitrary. For simplicity, in the following model we omit the cell classes and the queueing disciplines considered in the previous examples. The control delay is determined by the propagation time, the decision time, and the response time.

Example 8.14

Let us assume that this delay is relatively short, $\Delta = 100$ *time units, and that the active period is relatively long, so that the effect of control is visible within a single active period. Assume that 300 cells are transmitted during each active period. Outside the active periods, which recur every 1000 time units, the source remains silent.*

The local traffic intensity is constant:

$$\lambda^{(l)} = 0.5$$

cells per unit of time. The buffer can hold 150 packets.

At the beginning of each active period, the virtual link starts transmitting at the maximum rate

$$\lambda^{(cv)} = 1$$

cell per unit of time.

When the queue length exceeds the threshold level

$$S_o = 20$$

cells, a command is sent to the source to reduce its transmission rate according to

$$\lambda^{(cv)} \leftarrow 0.667\,\lambda^{(cv)}.$$

This command reaches the source and is executed after Δ *time units.*

After a period of $\Delta_1 = 140$ *time units, that is, 40 time units after the change in traffic intensity, the queue is checked again: if its length still exceeds the threshold* S_o*, another command is sent to reduce the source activity.*

If the queue length is less than

$$S_i = 10,$$

a command is sent permitting an increase in activity, according to

$$\lambda^{(cv)} \leftarrow 1.5\,\lambda^{(cv)}.$$

We do not analyse the effectiveness of this example protocol here; our aim is only to present it in the form of a model. Figure 8.63 compares the simulation results with those of the diffusion model (diffusion1 and diffusion2, defined earlier on p. **??***).*

In the diffusion model, control decisions are made on the basis of the instantaneous value of the mean queue length predicted by the model. One may also use, as a congestion indicator, the value $p_N(t)$ predicted by the model.

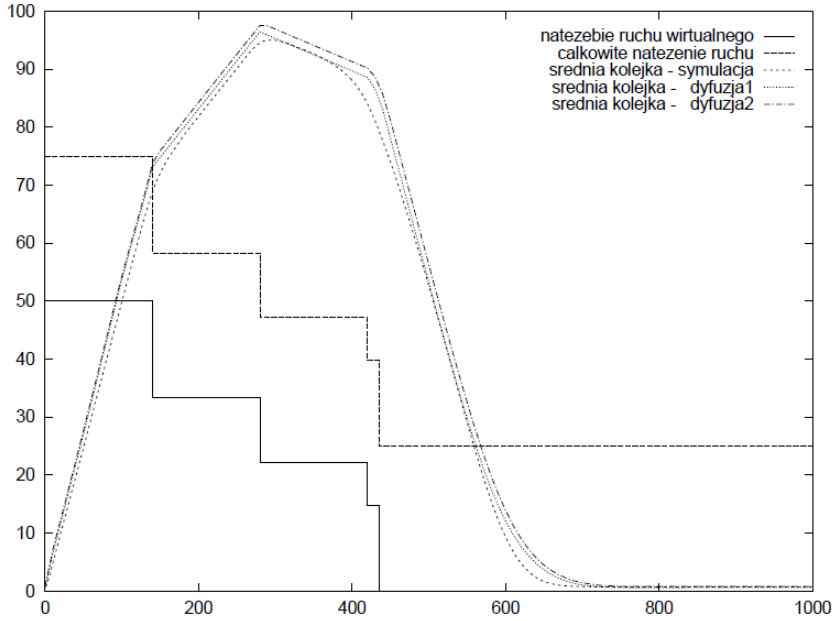


Figure 8.63: Control during an active period covering the transmission of 300 cells. Control decisions are visible as abrupt changes in the input-stream intensity; simulation and diffusion-model results

Let us now assume that the source activity periods are short (25 cells transmitted in each one) and frequent (every 150 time units). The source response time remains as before, $\Delta = 100$ time units, so the control algorithm cannot affect the source during the same activity period in which congestion arises.

Local traffic (which represents the aggregate of flows belonging to other virtual networks) changes slowly and non-periodically. The thresholds $S_o = 20$ and $S_i = 10$, as well as the control period $\Delta_1 = 140$, remain the same as before. The control algorithm reduces the permissible peak activity according to

$$\lambda^{(cv)} \leftarrow 0.5\,\lambda^{(cv)}$$

or increases it according to

$$\lambda^{(cv)} \leftarrow 2\,\lambda^{(cv)}.$$

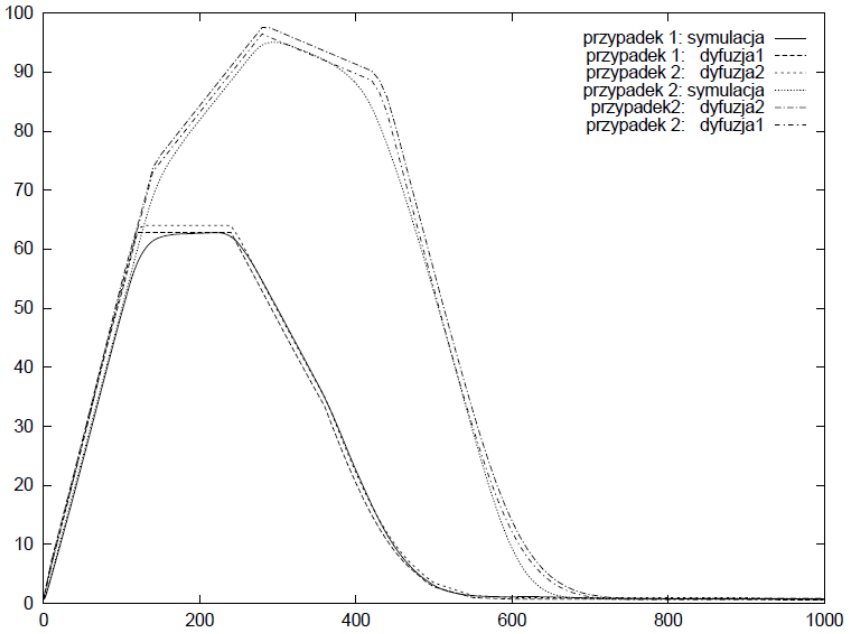


Figure 8.64: Control of the flow generated by the source during an active period comprising 300 cells, with control step $\Delta = 100$ (case 1), and during an active period of 250 cells with $\Delta = 80$ (case 2); mean queue length; simulation and diffusion-model results

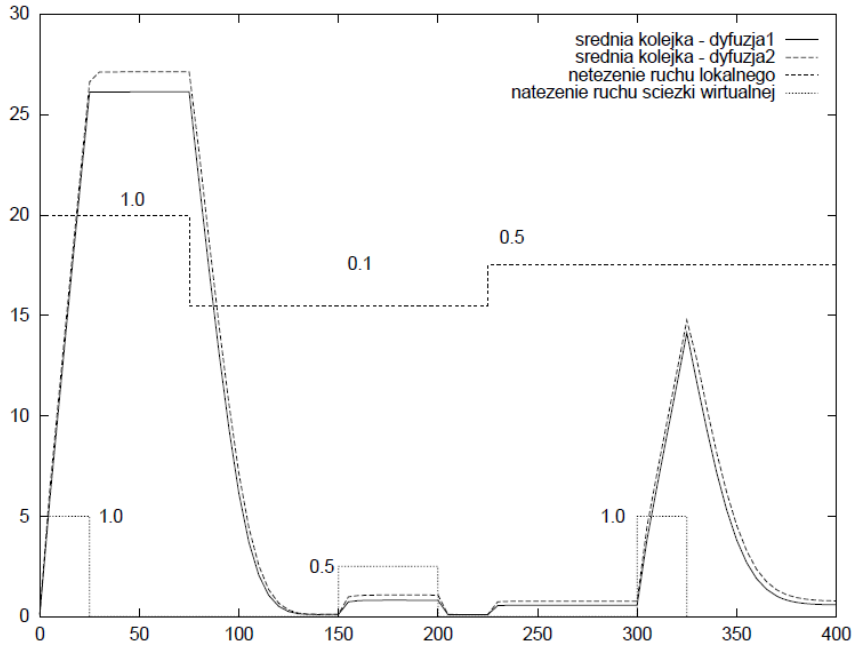


Figure 8.65: Control of the flow generated by the source over three consecutive active periods of 25 cells each

Figure 8.65 illustrates an example of this mechanism: the first active period, accompanied by heavy local traffic, causes the source activity in the next active period to be reduced by half (and thus that period lasts twice as long); subsequently the queue decreases, and the algorithm allows full source activity during the third period.

■

8.6.1 Feedback control: a simplified model

As shown in Figs. 7.4 and 7.5, the average queue length at a station, following a step change in load, evolves along a curve similar to an exponential function. Its parameter depends on the coefficient of variation of the interarrival-time distribution, the service-time distribution, and the degree of system utilisation; see Fig. 7.6.

Therefore, if we treat the traffic intensity as the input signal and the average queue length as the output signal, then the service station may be regarded as a first-order inertial system, that is, as an object that transforms the unit-step function $\mathbf{1}(t)$ ($\mathbf{1}(t) = 0$ for $t < 0$, $\mathbf{1}(t) = 1$ for $t \geq 0$) into the function

$$k\left(1 - e^{-t/T}\right),$$

where the constant k denotes the system gain and the constant T characterises the speed of change of the output signal.

The dynamics of the output will be similar if, instead of the average queue length, we consider as the output signal the loss factor $L^{(1)}(t)$ or $L^{(2)}(t)$, although the gain coefficient will, of course, be different.

Let $\bar{X}_i(s)$ denote the Laplace transform of the time-varying input signal at input i of the network, and let $\bar{Y}_i(s)$ denote the Laplace transform of the output signal (the average number of packets in the queue or the loss factor). If the transfer function of the system is

$$\bar{K}_i(s) = \frac{k_i}{1 + sT_i},$$

then the output signal is given by

$$\bar{Y}_i(s) = \bar{K}_i(s)\bar{X}_i(s) = \frac{k_i}{1 + sT_i}\bar{X}_i(s). \tag{8.70}$$

The parameters k_i and T_i for station i depend nonlinearly on the system utilisation and on the first two moments of the interarrival-time and service-time distributions:

$$k_i = k_i(\varrho_i, C_{A_i}^2, C_{B_i}^2), \qquad T_i = T_i(\varrho_i, C_{A_i}^2, C_{B_i}^2).$$

The attenuation of the signal transmitted back to the source is modelled by an element whose transfer function has the form

$$k_{2i}e^{-sD_i}.$$

The complete model is shown in Fig. 8.66, and an example time course of the signals is presented in Fig. 8.67.

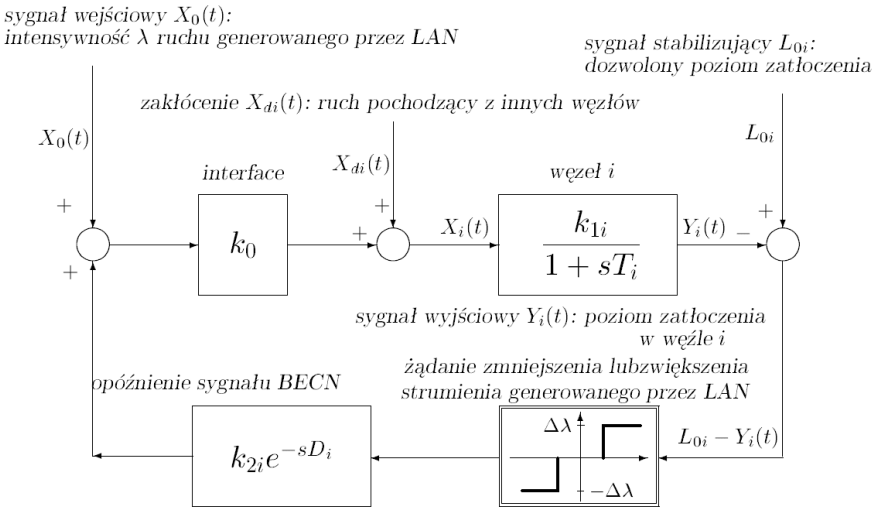


Figure 8.66: Block diagram of the ECN traffic-control system

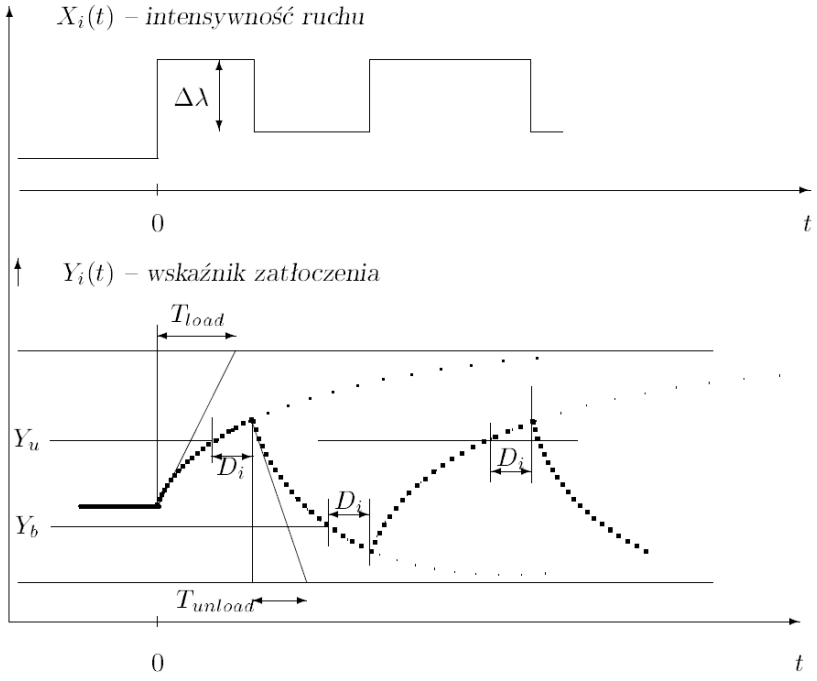


Figure 8.67: Example of changes in traffic intensity and congestion at a network node when the ECN traffic-control mechanism is used

8.7 Conclusions

The examples presented in this chapter illustrate several ways in which diffusion approximations can be used to analyse phenomena occurring in modern wide-area integrated-services networks. These applications include methods for controlling the traffic admitted to the network, ensuring packet-priority differentiation, modelling packet-queueing algorithms in network switches, predicting the behaviour of data streams traversing the network, describing irregularities in packet arrivals at the destination, and analysing feedback-based reactions to congestion arising in the network. In all of these applications, diffusion approximation proves to be a useful tool.

Its main advantages include the flexibility of adapting the method to different situations and queueing algorithms, the possibility of constructing models in the form of arbitrarily configured networks of stations, the ability to include multiple customer classes and arbitrary input streams and service-time distributions, and, above all, the ability to analyse transient states. From the models presented in the examples, a more general network model can also be constructed.

The disadvantages of the method should also be emphasised. In some cases, the approximation errors are not negligible. Moreover, in the analysis of transient states, the numerical inversion of Laplace transforms requires considerable effort in developing suitable software. This implementation must be carried out carefully in order to avoid numerical errors associated with manipulating very large and very small values.

Nevertheless, the computational effort required is significantly lower than for comparable Markov or simulation models, while the accuracy is higher than that obtained from fluid approximations. It therefore appears that the diffusion-approximation method is well suited to the analysis of packet (cell) traffic in modern B-ISDN networks.

The large volume of information generated by variable-rate sources, combined with a policy aimed at high link utilisation, leads to transient congestion in network buffers. The traffic-control algorithms considered here therefore require an analysis of transient states.

The diffusion-approximation method naturally provides such a description. It is more accurate than the so-called fluid approximation, which treats information flows as the flow of a fluid through a network of pipes and takes into account only average transmission values. At the same time, it is less complex and requires fewer computational resources than Markov models, which typically lead to very large systems of differential equations that must be solved numerically.

Chapter 9

Fluid flow approximation

Queuing models usually refer to single nodes or small topologies. However, it is crucial to assess the impact of various decisions, such as introducing new protocols or devices, on a much larger scale corresponding to the size of the Internet.

The **fluid-flow approximation** can be used for this purpose. This method may be regarded as an extension of Mean Value Analysis adapted to the study of transient states.

The number of customers in the system $n(t)$ may be compared to the volume of water in a reservoir, while the input and output streams of customers correspond to the flows of water entering and leaving the reservoir. The evolution of the queue length is thus described by a simple differential equation, where $\lambda(t)$ is the time-dependent input rate and μ is the constant output rate:

$$\frac{dn(t)}{dt} = \begin{cases} \lambda(t) - \mu, & n(t) > 0, \\ \max\{0, \lambda(t) - \mu\}, & n(t) = 0. \end{cases} \tag{9.1}$$

This simplified approach was introduced by Moran [84] to study water reservoirs and storage systems, and was later adopted in computer science. It has been used for the analysis of computer networks in a series of papers by Towsley and his collaborators, e.g., [76, 82], and subsequently by other authors, such as Sakumoto et al. [108].

The fluid-flow approximation is a first-order model: time-dependent flows and queue lengths are represented only by their mean values and are linked by first-order differential equations. Consequently, it is less accurate than the diffusion approximation. A numerical comparison of diffusion and fluid-flow methods for modelling transient states was presented in [91].

The simplicity of the fluid-flow approximation makes it possible to construct models of much larger networks and to analyse systems of a scale that would be infeasible using more detailed stochastic approaches. In particular, it enables the study of traffic control mechanisms applied in the TCP/IP stack.

The TCP policy controls the sending rate using the *congestion window*, which determines the number of packets that may be transmitted without waiting for acknowledgments. In the classical algorithm, this window is increased (by one or a smaller value) upon the arrival of each acknowledgment and sharply decreased (typically halved) when a packet loss is detected.

We assume that packet loss is caused by congestion in a bottleneck router located somewhere along the path between the sender and the receiver. This behaviour can be described by the following equations:

$$\frac{dW(t)}{dt} = \frac{1}{R(t)} - \frac{W(t)}{2} \cdot \frac{W(t-R(t))}{R(t-R(t))} \cdot p(t-R(t)), \tag{9.2}$$

$$\frac{dq(t)}{dt} = \begin{cases} -C + \dfrac{K(t)}{R(t)} W(t), & q(t) > 0, \\ \max\left\{0, -C + \dfrac{K(t)}{R(t)} W(t)\right\}, & q(t) = 0, \end{cases} \tag{9.3}$$

where:

- $W(t)$ is the TCP congestion window size [packets],
- $q(t)$ is the average queue length at the bottleneck router [packets],
- $R(t)$ is the round-trip time [s],
- C is the link capacity [packets/s],
- $p(t)$ is the probability of packet dropping or marking,
- $K(t)$ is the number of active TCP sessions.

Equation (9.2) can be extended to include the slow-start phase and timeout losses, see [82].

The fluid-flow approximation can be used to analyse the stability of TCP/IP traffic control. The system of equations describing TCP flow dynamics is linearized around an operating point, and the resulting model is expressed in terms of transfer functions. Both the queue and the RED mechanism are represented as first-order inertial systems.

The resulting control loop can then be analysed using standard tools of linear control theory, and stability can be assessed, for example, using the Nyquist criterion.

To incorporate unresponsive flows into the model, Eq. (9.3) is modified as follows [47]. Let $l(t)$ denote the traffic of responsive flows and $u(t)$ that of unresponsive flows.

Assuming that $u(t)$ is a stationary process with mean value u_0 and small fluctuations on a finer time scale, it may be approximated as constant. This effectively reduces the bandwidth available to responsive flows:

$$\dot{q}(t) = \begin{cases} -(C - u_0) + \dfrac{K(t)}{R(t)} W(t), & q(t) > 0, \\ \max\left\{0, -(C - u_0) + \dfrac{K(t)}{R(t)} W(t)\right\}, & q(t) = 0, \end{cases} \tag{9.4}$$

Linearization around the operating point (W_0, p_0, q_0, u_0) leads to the following transfer functions (expressed in the Laplace domain):

$$W(s) = -P_{win}(s) e^{-sR_0} p(s), \qquad l(s) = \frac{K}{R_0} W(s), \tag{9.5}$$

$$q(s) = P_{que}(s) \left[l(s) + u(s) \right], \qquad p(s) = q(s)\, C_{aqm}(s), \tag{9.6}$$

where

$$P_{win}(s) = \frac{\frac{R_0 C_{eff}^2}{2K^2}}{s + \frac{2K}{R_0^2 C_{eff}}}, \qquad P_{que}(s) = \frac{1}{s + \frac{C_{eff}}{C}\frac{1}{R_0}}, \qquad C_{eff} = C - u_0,$$

and $C_{aqm}(s)$ is the transfer function of the AQM controller.

These equations are complemented by relations defining the packet dropping or marking probability as a function of the average queue size. In the case of the classical RED algorithm, the dropping probability is given by:

$$p_d = \begin{cases} 0, & \text{if } avg \le min_{th}, \\ \dfrac{avg - min_{th}}{max_{th} - min_{th}} \cdot max_p, & \text{if } min_{th} < avg \le max_{th}, \\ 1, & \text{if } avg > max_{th}. \end{cases} \tag{9.7}$$

The moving average avg is computed upon each packet arrival and acts as a low-pass filter for the instantaneous queue length:

$$avg = \begin{cases} (1-w)\, avg + w\, n, & \text{if the queue is not empty}, \\ (1-w)^{\mu/\lambda}\, avg, & \text{if the queue is empty}, \end{cases}$$

where w is a small constant parameter and n is the instantaneous queue size.

The averaging process (denoted by x) may be modelled as:

$$\frac{dx}{dt} = \frac{\ln(1-w)}{\Delta}\left[q(t) - x(t)\right], \tag{9.8}$$

where Δ is the interarrival time of packets, typically approximated as $\Delta = 1/C$.

Thus, the transfer function of the RED mechanism, relating changes in queue length δq to changes in packet loss probability δp, takes the form:

$$C_{aqm}(s) = L_{red} \cdot \frac{k}{k+s},$$

where

$$k = -C\ln(1-w), \qquad L_{red} = \frac{max_p}{max_{th} - min_{th}}.$$

If a different AQM algorithm is used, the corresponding transfer function $C_{aqm}(s)$ must be modified accordingly.

The round-trip time is computed as

$$R_i(\boldsymbol{q}_i(t)) = \sum_{j \in \boldsymbol{V}_i} \frac{q_j(t)}{C_j} + \sum_{j \in \boldsymbol{L}_i} T_j. \tag{9.9}$$

where

- $\boldsymbol{q}_i$ is the vector of mean queue lengths along connection i,

- $\boldsymbol{V}_i$ is the set of nodes belonging to connection i,
- T_j is the propagation delay between nodes j and $j+1$,
- $\boldsymbol{L}_i$ is the set of links along connection i.

Following TCP principles, the intensity of each flow i is determined by a time-varying congestion window size W_i maintained by the sender. Equation (9.10) describes the evolution of the congestion window according to the basic Reno algorithm. Although it is no longer widely used, it serves as a reference model in the performance evaluation of newer TCP variants [3, 4, 107, 83, 56].

Following the original fluid-flow approximation studies, we focus on the congestion avoidance phase of TCP Reno. At the same time, the model can be extended to include slow start and fast retransmit/fast recovery phases, or to replace Reno with other protocols.

The window size increases by one per round-trip time in the absence of packet losses (first term on the right-hand side of Equation (9.10)) and decreases by half its current value, with a delay τ (since the sender is not immediately aware of the loss), when packet losses occur along the flow path (second term).

The loss intensity for connection i is computed as the packet sending rate

$$\frac{W_i(t-\tau)}{R_i(\boldsymbol{q}_i(t-\tau))}$$

multiplied by the probability that a packet is lost along its path. Therefore, the evolution of the congestion window is given by:

$$\frac{dW_i(t)}{dt} = \frac{1}{R_i(\boldsymbol{q}(t))} - \frac{W_i(t)}{2} \cdot \frac{W_i(t-\tau)}{R_i(\boldsymbol{q}_i(t-\tau))} \cdot \left(1 - \prod_{j \in \boldsymbol{V}_i} (1 - p_{ij})\right), \tag{9.10}$$

where p_{ij} is the packet loss probability for flow i at node j.

In AQM mechanisms used to control TCP/IP connections, packets may be dropped even if buffer space is still available. The drop probability $p(x_k)$ for the k-th packet is defined in Equation (9.12) according to the Random Early Detection (RED) mechanism [8, 33]. It depends on the weighted moving average of the queue length:

$$x_k = w x_{k-1} + (1-w) q_k, \tag{9.11}$$

where $0 < w < 1$ determines the balance between historical and current queue states.

This average acts as a low-pass filter, smoothing short-term fluctuations and emphasizing long-term trends. The choice of w (as well as thresholds $t_{\min}$ and $t_{\max}$) depends on traffic characteristics [90, 1]. In this model, we use a typical value $w = 0.95$. Unlike real networks where updates occur at packet arrivals, the model updates x_k at fixed intervals $\Delta = 0.1\,\text{s}$.

The drop probability is defined as:

$$p(x_k) = \begin{cases} 0, & 0 \le x_k < t_{\min}, \\ \dfrac{x_k - t_{\min}}{t_{\max} - t_{\min}} p_{\max}, & t_{\min} \le x_k \le t_{\max}, \\ 1, & x_k > t_{\max}. \end{cases} \tag{9.12}$$

9.0.1 Fluid-Flow Approximation of an Energy-Efficient Node

We assume that a router operates either in normal mode or in energy-saving mode. In the latter, the router does not serve packets but continues to buffer them.

Two queue-length thresholds define the operating mode:

1. When the queue length q decreases to the lower threshold T_A, service is switched off.
2. When the queue length increases to the upper threshold $T_B > T_A$, service is switched on.
3. Otherwise, the operating mode remains unchanged.

The hysteresis $T_B - T_A$ prevents frequent switching between modes. The queue dynamics are then described by:

$$\frac{dq_j(t)}{dt} = \begin{cases} \sum_{i=1}^{K} \frac{W_i(t)}{R_i(\boldsymbol{q}(t))} N_i - \mathcal{H}(q_j(t) > 0)\, C_j, & q_j \geq T_B, \\ \sum_{i=1}^{K} \frac{W_i(t)}{R_i(\boldsymbol{q}(t))} N_i, & q_j \leq T_A, \\ \sum_{i=1}^{K} \frac{W_i(t)}{R_i(\boldsymbol{q}(t))} N_i - \mathcal{H}(q_j(t) > 0)\, C_j, & T_A < q_j < T_B, \text{ (normal mode)}, \\ \sum_{i=1}^{K} \frac{W_i(t)}{R_i(\boldsymbol{q}(t))} N_i, & T_A < q_j < T_B, \text{ (energy-saving mode)}. \end{cases} \tag{9.13}$$

We compare two networks with identical topology and traffic: one composed of standard routers and the other of energy-aware routers.

To compare the performance of both networks, an exemplary topology consisting of 134,000 nodes and 50,000 flows was considered.

In the basic network, only 3.52% of routers were idle for less than 25% of the time. Most routers (94.17%) were idle between 25% and 50% of the time, while only 2.31% of nodes were idle up to 75% of the time.

In the energy-saving mode, the results change significantly. A negligible fraction of nodes (0.0024%) were in a non-transmitting state for less than 25% of the time. Only 0.84% of routers were idle up to 50% of the time, whereas as many as 99.15% of routers did not transmit for up to 75% of the time. A very small fraction (0.006%) remained inactive for the entire observation period.

The second set of results concerns the congestion rate. We introduce two indicators: CR_{1i} and CR_{2i}. The first is defined for each node i as the time-average of the ratio between the queue lengths in the energy-aware network, $q_{es_{i,j}}$, and in the basic network, $q_{n_{i,j}}$, according to:

$$CR_{1i} = \frac{1}{T} \sum_{j=1}^{T} \frac{q_{es_{i,j}}}{q_{n_{i,j}}}, \quad \text{for } q_{n_{i,j}} > 0, \tag{9.14}$$

where T denotes the number of observation intervals.

The second indicator is defined as the ratio of average queue lengths:

$$CR_{2i} = \frac{\frac{1}{T} \sum_{j=1}^{T} q_{es_{i,j}}}{\frac{1}{T} \sum_{j=1}^{T} q_{n_{i,j}}}. \tag{9.15}$$

The values of CR_{1i} range from 0.1876 to 34,774.9 and are grouped into six categories:

(a) $CR_{1i} \geq 10{,}000$ approximately 0.0048% of nodes,

(b) $1{,}000 < CR_{1i} \leq 10{,}000$ approximately 0.054% of nodes,

(c) $100 < CR_{1i} \leq 1{,}000$ approximately 0.38% of nodes,

(d) $10 < CR_{1i} \leq 100$ approximately 4.9% of nodes,

(e) $1 < CR_{1i} \leq 10$ approximately 93.33% of nodes,

(f) $CR_{1i} \leq 1$ approximately 1.33% of nodes.

Most values indicate that congestion in the energy-saving mode is typically between 1 and 10 times higher than in the normal mode (category (e), see Figure **??**).

Further analysis reveals that nodes in category (a) correspond to routers with very small buffers (e.g., 18, 63, and 309 packets). In the normal mode, queue lengths at these nodes are nearly zero (on the order of 10^{-6} for 18-packet buffers and 10^{-4} for 309-packet buffers), whereas in the energy-saving mode they exceed the threshold T_B (25%).

The indicator CR_{1i} is highly sensitive to transient spikes in the ratio $\frac{q_{es_{i,j}}}{q_{n_{i,j}}}$. Even a single large value can significantly inflate the metric. To mitigate this effect, we introduce CR_{2i}, which provides a more stable comparison.

The indicator CR_{2i} represents how many times the average queue length in the energy-saving mode exceeds that in the normal mode. Its values range from 0.5714 to 3.3297. For most nodes (99.7%), the values lie within the interval $[0.75, 2.0]$, indicating moderate differences.

The third set of results concerns the identification of overload periods in network nodes. The energy-saving mode reduces the occurrence of congestion periods and results in smoother system behavior. However, it significantly decreases the average throughput: 4.49 packets/s in the energy-saving mode compared to 8.31 packets/s in the normal mode.

This trade-off reflects the fundamental effect of energy-saving mechanisms: buffering packets and limiting transmission reduces bandwidth. Such a reduction may be acceptable for non-priority traffic or traffic without strict QoS requirements, as it lowers energy consumption. However, for QoS-sensitive traffic, the performance degradation is substantial, and parameter tuning (e.g., threshold selection) is necessary.

The model demonstrates how the introduction of energy-saving on–off nodes affects overall network performance. While it reduces congestion and energy consumption, it also significantly lowers throughput. The proposed approach enables efficient analysis of large-scale network behavior.

9.0.2 Conclusions

In the fluid-flow model, packet traffic is treated as a continuous flow rather than as discrete random events. The queue length $Q(t)$ evolves according to a deterministic balance equation (9.1), capturing the average system behavior without explicitly modeling stochastic variability.

A key advantage of the fluid-flow approach is its excellent scalability. It is particularly suitable for:

- large-scale communication networks (hundreds of thousands of nodes),

- analysis of congestion control mechanisms (e.g., TCP),
- investigation of long-term system behavior.

However, the fluid-flow approximation has important limitations:

- it ignores traffic variability and burstiness,
- it does not capture stochastic effects such as queue fluctuations or jitter,
- it is less accurate for highly variable or correlated traffic (e.g., ON–OFF sources),
- it cannot represent short-term dynamics precisely.

For detailed performance evaluation (e.g., delay distributions, losses, jitter), more advanced models such as diffusion approximations are required. In practice, fluid models are often complemented by stochastic approaches to obtain a comprehensive description of network behavior.

Conclusions and Perspectives

The development of queueing theory has always followed the evolution of the systems it seeks to describe. From the early telephone networks studied by Erlang to today's vast, heterogeneous, and highly dynamic communication infrastructures, modelling techniques have continuously adapted to increasing scale, complexity, and performance demands. This book has traced that evolution—from exact stochastic models to large-scale approximations—highlighting both the power and the limitations of existing approaches.

A central message emerging from this work is that the traditional boundaries of queueing theory are no longer sufficient. Modern systems rarely operate in steady state; instead, they evolve continuously under changing loads, adaptive protocols, and control mechanisms. In such environments, transient behaviour is not a secondary phenomenon—it is the system. Congestion, instability, and recovery occur over time scales that directly determine performance, user experience, and operational efficiency.

This shift calls for a rethinking of modelling paradigms. Diffusion and fluid-flow approximations, as presented in this book, offer a step in this direction. They move beyond static descriptions and enable the analysis of time-dependent dynamics, bridging the gap between microscopic stochastic behaviour and macroscopic system evolution. In doing so, they provide not only analytical tools but also a conceptual framework for understanding large-scale systems as evolving processes rather than equilibrium objects.

The implications of this perspective are far-reaching. As networks become increasingly programmable, autonomous, and data-driven, the distinction between system behaviour and control logic becomes blurred. Future performance models will likely need to incorporate learning mechanisms, feedback loops, and real-time adaptation as intrinsic components rather than external factors. Queueing systems will no longer be passive objects of analysis but active entities interacting with their environment.

At the same time, the scale of modern infrastructures challenges the very feasibility of traditional modelling techniques. Systems composed of hundreds of thousands—or even millions—of interacting elements cannot be understood solely through detailed stochastic descriptions. This reality suggests a growing role for multi-scale models, where fluid approximations capture global behaviour while stochastic refinements describe local fluctuations. The integration of these levels remains an open and compelling research problem.

Another emerging dimension is the increasing importance of energy efficiency and sustainability. As demonstrated in this book, introducing energy-aware mechanisms fundamentally alters system dynamics, creating new trade-offs between performance, stability, and resource consumption. These trade-offs are not merely technical—they reflect broader constraints that will shape the design of future networks and computing systems.

Looking ahead, several challenges stand out:

- developing unified frameworks that seamlessly combine stochastic, diffusion, and fluid

descriptions,

- extending models to capture non-stationary, adaptive, and learning-driven behaviour,
- designing scalable analytical tools capable of handling extreme system sizes,
- and incorporating new performance criteria, including energy consumption and robustness.

In conclusion, queueing theory continues to evolve in response to the challenges posed by modern technological systems. While its classical foundations remain indispensable, new methods and perspectives are required to address the scale, variability, and dynamics of contemporary applications.

Ultimately, the future of queueing theory lies not only in refining existing models but in redefining the questions we ask. Rather than focusing solely on equilibrium metrics, we must understand how systems evolve, adapt, and respond to changing conditions. This shift—from static analysis to dynamic understanding—may well represent the next major transformation of the field.

Artificial Intelligence is transforming the role and applicability of queueing models by shifting them from static, assumption-driven analytical tools into adaptive, data-informed components of modern performance engineering frameworks. Instead of relying on fixed distributions and steady-state assumptions, AI enables queueing models to be continuously calibrated using real system observations, capturing non-stationary workloads, complex dependencies, and environment-specific behaviors that classical formulations cannot easily represent. Through techniques such as parameter inference, residual correction, and surrogate modelling, AI enhances the predictive accuracy of queue-based analyses while preserving their interpretability and structural insight. Moreover, when integrated with reinforcement learning and control mechanisms, queueing models evolve from passive evaluation tools into active elements of decision-making systems, guiding scheduling, routing, and resource allocation in dynamic environments. As a result, the combined AI–queueing paradigm extends the relevance of queueing theory to contemporary large-scale, heterogeneous, and adaptive computing systems, making it a central component in areas such as cloud computing, network optimization, and AIOps.

It is our hope that this book contributes to that transition by providing both the classical foundations and the modern tools needed to explore it. If it also encourages readers to challenge established assumptions and to develop new modelling paradigms, then it will have fulfilled its broader purpose.

Appendix A

Service time, waiting time in the queue, interarrival times between successive customers, and the number of customers at a service station are random variables in the models described in this handbook. Let us recall the basic concepts and definitions related to random variables.

The mathematical definition of a random variable is as follows. Let us consider a set

$$\Omega = \{\omega_1, \omega_2, \ldots\}$$

whose elements are called *elementary events*. In the set Ω, we distinguish subsets A, called *events*, forming a family (more precisely, a *Borel σ-algebra* $\mathcal{B}$ of subsets of Ω). On $\mathcal{B}$ there is defined a function $P(\cdot)$ which is: non-negative ($P(A) \geq 0$), normalised ($P(\Omega) = 1$), and countably additive,

$$P\left(\bigcup_{k=1}^{\infty} A_k\right) = \sum_{k=1}^{\infty} P(A_k), \qquad \text{if } A_j \cap A_k = 0 \text{ for } j \neq k.$$

The triple $(\Omega, \mathcal{B}, P)$ is called a *probability space*.

A *random variable* X is a real-valued function $X(\omega)$ defined on the set of elementary events Ω and measurable with respect to $\mathcal{B}$, i.e. for every real x we have

$$\{\omega : X(\omega) < x\} \in \mathcal{B}.$$

The function

$$F_X(x) = P\{\omega : X(\omega) < x\} = P(X < x)$$

is called the *distribution function* of the random variable X.

Less formally, we may say that the variable X takes random values, and its behaviour is described by the distribution function $F_X(x) = P(X < x)$, which determines the probability that the value of X is less than any given real number x. The function $F_X(x)$ is non-decreasing, left-continuous, and satisfies

$$F_X(-\infty) = 0, \qquad F_X(\infty) = 1.$$

Since $P(X < a) = F_X(a)$ and $P(X < b) = F_X(b)$, it follows that

$$P(a \leq X < b) = F_X(b) - F_X(a).$$

If a random variable can take only specific values $x_1, x_2, \ldots$ (a finite or countable set) with probabilities $p_1, p_2, \ldots$, we speak of a *discrete random variable*. The distribution function of such a variable is a step function; at the point $x = x_i$ it increases by the amount p_i.

If the variable X can take values from an uncountable set, we call it a *continuous random variable*. Then there exists a non-negative function $f_X(x)$, called the *density function*, such that for any values a and b,

$$P(a \le X < b) = \int_a^b f_X(x)\,dx,$$

i.e.

$$F_X(x) = \int_{-\infty}^{x} f_X(\xi)\,d\xi,$$

and, of course,

$$\int_{-\infty}^{\infty} f_X(x)\,dx = 1.$$

The distribution function or density function contains all the information about the distribution of a random variable. This information is also carried by the moments of the distribution. The moment of order m is

$$E[X^m] = \begin{cases} \sum_i x_i^m p_i, & \text{for a discrete variable } X, \\ \int_{-\infty}^{\infty} x^m f_X(x)\,dx, & \text{for a continuous variable } X. \end{cases}$$

The first-order moment $E[X]$ is the mean value. The central moment of order m is

$$E\{[X - E[X]]^m\}.$$

The second central moment, i.e. the variance, is used especially often:

$$\mathrm{Var}[X] = \sigma_X^2 = \begin{cases} \sum_i [x_i - E[X]]^2 p_i, & \text{for a discrete variable } X, \\ \int_{-\infty}^{\infty} [x - E[X]]^2 f_X(x)\,dx, & \text{for a continuous variable } X. \end{cases}$$

Expanding the square $[X - E[X]]^2$, we obtain the relation

$$\mathrm{Var}[X] = E[X^2] - E[X]^2.$$

The ratio of the variance to the square of the mean is called the squared coefficient of variation:

$$C_X^2 = \frac{\sigma_X^2}{E[X]^2}.$$

For the exponential distribution, $C_X^2 = 1$. Since many queueing models give exact results only for exponential distributions, and the quality of approximate models deteriorates as the coefficients of variation of the assumed distributions deviate from unity, the value of C_X^2 is of great importance in modelling.

Instead of listing the probabilities $p_0, p_1, p_2, \ldots$ of a discrete distribution, we may specify its *generating function*

$$G(z) = \sum_{i=0}^{\infty} p_i z^i.$$

This allows the probabilities p_i to be recovered:

$$\begin{aligned} \left.\frac{dG(z)}{dz}\right|_{z=0} &= \left.\sum_{i=1}^{\infty} i p_i z^{i-1}\right|_{z=0} = p_1, \\ \left.\frac{d^k G(z)}{dz^k}\right|_{z=0} &= k!\, p_k. \end{aligned}$$

It also allows the moments of the distribution to be determined, for example

$$\left.\frac{dG(z)}{dz}\right|_{z=1} = \sum_{i=0}^{\infty} i p_i = E[X].$$

If $X = X_1 + X_2$, where X_1 and X_2 are independent discrete random variables, then

$$G_X(z) = G_{X_1}(z)\, G_{X_2}(z).$$

If a random variable is the sum of independent continuous random variables $X = X_1 + X_2$, then its density is the convolution of the densities of the component variables:

$$f_X(x) = \int_0^x f_{X_1}(\xi)\, f_{X_2}(x-\xi)\, d\xi = f_{X_1}(x) * f_{X_2}(x),$$

and its mean and variance are

$$E[X] = E[X_1] + E[X_2], \qquad \sigma_X^2 = \sigma_{X_1}^2 + \sigma_{X_2}^2.$$

Distributions of discrete random variables used in the textbook

- *Single-point distribution (deterministic, degenerate)*
  The variable takes only one value x_1; $p_1 = 1$. Hence

  $$E[X] = x_1, \qquad E[X^2] = x_1^2, \qquad \mathrm{Var}[X] = 0.$$

- *Bernoulli distribution (two-point)*
  The variable X takes two values: 0 and 1. Let us call 1 a success and 0 a failure, and denote

  $$P(X = 0) = q, \qquad P(X = 1) = p = 1 - q.$$

  Then

  $$\begin{aligned} E[X] &= 0 \cdot q + 1 \cdot p = p, \\ E[X^2] &= 0^2 \cdot q + 1^2 \cdot p = p, \\ \mathrm{Var}[X] &= E[X^2] - E[X]^2 = p - p^2 = p(1-p) = pq. \end{aligned}$$

- *Binomial distribution*
  Consider a sequence of Bernoulli trials. Let X_n denote the number of successes in n trials. The probability that there are exactly k successes is

  $$P(X_n = k) = p_k = \binom{n}{k} p^k (1-p)^{n-k}, \qquad 0 \le k \le n.$$

Example
If the probability of correct packet transmission in a computer network is p, then the binomial distribution determines the probability that, during the transmission of n packets, all are transmitted correctly, or that there are exactly i transmission errors.

- *Geometric distribution*
  Consider again a sequence of Bernoulli trials. Let X_i be the number of trials until the first success (including the successful trial). Then
  $$p_i = q^{i-1}p = (1-p)^{i-1}p, \qquad i = 1, 2, \ldots$$

- *Modified geometric distribution*
  The random variable is the number of trials before the first success in a sequence of Bernoulli trials. Then
  $$p_i = q^i p = (1-p)^i p, \qquad i = 0, 1, 2, \ldots$$

  Example
  The generating function of the modified geometric distribution is
  $$G(z) = \sum_{k=0}^{\infty} p(1-p)^k z^k = \frac{p}{1-(1-p)z}.$$

- *Poisson distribution*

  $$p_n = \frac{\alpha^n}{n!}e^{-\alpha}, \qquad n = 0, 1, 2, \ldots$$

  Example
  If events occur at intervals determined by an exponential distribution with parameter λ, then the number of events occurring during time interval t has a Poisson distribution with $\alpha = \lambda t$.

  Example
  The generating function of the Poisson distribution is
  $$G(z) = \sum_{k=0}^{\infty} \frac{\alpha^k}{k!}e^{-\alpha}z^k = e^{-\alpha}\sum_{k=0}^{\infty}\frac{(\alpha z)^k}{k!} = e^{-\alpha(1-z)}.$$

Continuous distributions of random variables used in the textbook

- *Exponential distribution*

density function	$f_X(x) = \mu e^{-\mu x}, \qquad x \geq 0$
distribution function	$F_X(x) = 1 - e^{-\mu x}$
mean value	$E[X] = \dfrac{1}{\mu}$
variance	$\mathrm{Var}[X] = \dfrac{1}{\mu^2}$
coefficient of variation	$C_X^2 = \dfrac{\mathrm{Var}[X]}{E[X]^2} = 1$

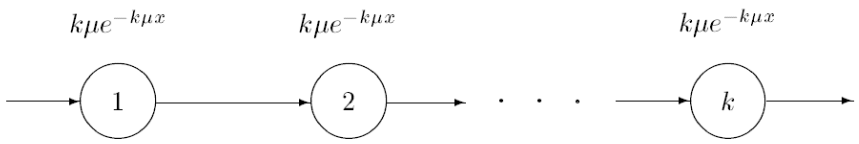


Figure 9.1: Erlang distribution of order k

- *Erlang distribution of order k* ($k \geq 1$)
  A variable with Erlang distribution of order k may be interpreted as the sum of k independent random variables having the same exponential distribution — Fig. 9.1.

  The density function of the Erlang distribution is the k-fold convolution of the exponential density. Assume that the parameter of the exponential distribution in each phase has the value $k\mu$.

  density function $f_X(x) = k\mu e^{-k\mu x} * k\mu e^{-k\mu x} * \cdots * k\mu e^{-k\mu x} = \dfrac{k\mu(k\mu x)^{k-1}e^{-k\mu x}}{(k-1)!}$

  distribution function $F_X(x) = 1 - e^{-k\mu x} \displaystyle\sum_{i=0}^{k-1} \frac{(k\mu x)^i}{i!}$

  mean value $E[X] = \dfrac{1}{\mu}$

  variance $\mathrm{Var}[X] = \dfrac{1}{k\mu^2}$

  coefficient of variation $C_X^2 = \dfrac{\mathrm{Var}[X]}{E[X]^2} = \dfrac{1}{k} \leq 1$

  where $*$ denotes convolution.

  For $k = 1$, the Erlang distribution reduces to the exponential distribution. As k increases, the variance decreases, the variable becomes more concentrated around its mean value, and the distribution approaches a single-point distribution.

- *Hyperexponential distribution of order k* ($k \geq 1$)
  The variable is drawn from one of k exponential distributions, chosen with probabilities a_i, $i = 1, 2, \ldots, k$, and having parameters μ_i — Fig. 9.3.

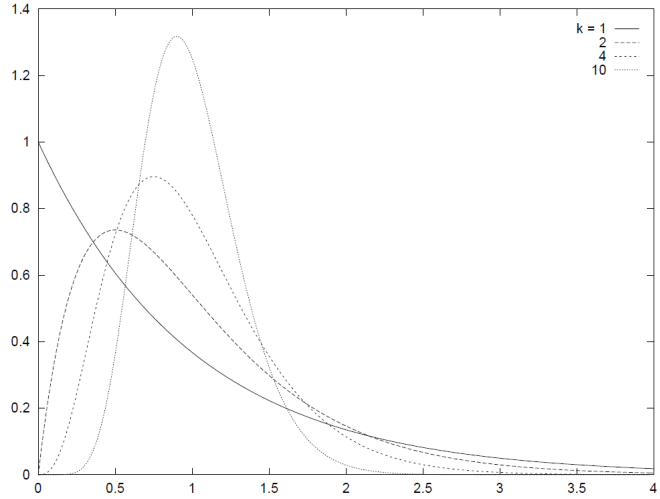


Figure 9.2: Density of Erlang distribution of order k

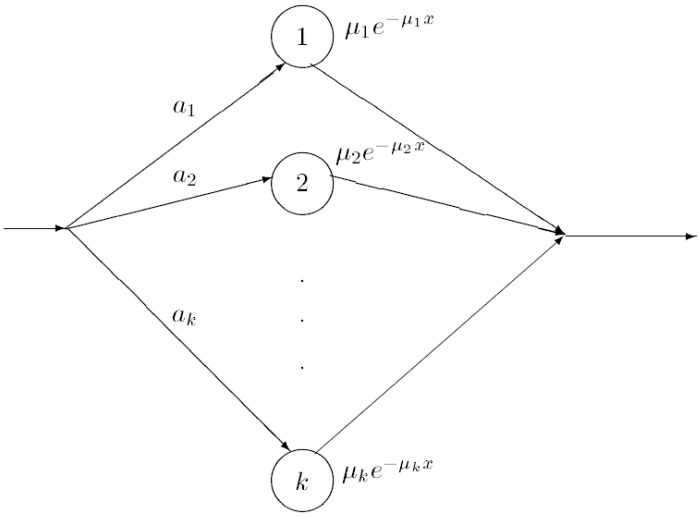


Figure 9.3: Hyperexponential distribution of order k

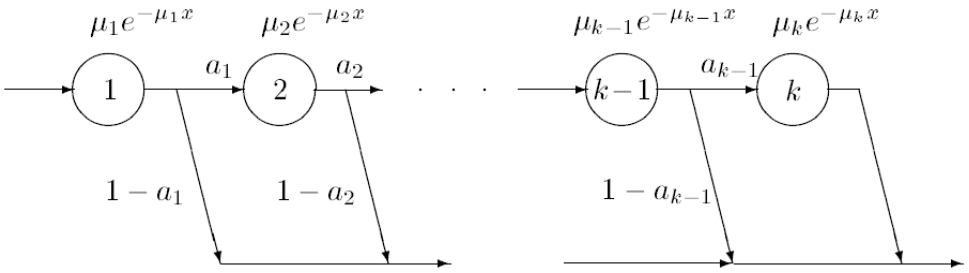


Figure 9.4: Cox distribution of order k

density function $$f_X(x) = \sum_{i=1}^{k} a_i \mu_i e^{-\mu_i x}, \qquad a_i \geq 0, \quad \sum_{i=1}^{k} a_i = 1$$

distribution function $$F_X(x) = 1 - \sum_{i=1}^{k} a_i e^{-\mu_i x}$$

mean value $$E[X] = \frac{1}{\mu} = \sum_{i=1}^{k} \frac{a_i}{\mu_i}$$

variance $$\mathrm{Var}[X] = \frac{1}{\mu^2} + \sum_{i=1}^{k} \sum_{j=1}^{k} a_i a_j \left(\frac{1}{\mu_i} - \frac{1}{\mu_j} \right)^2$$

coefficient of variation $$C_X^2 = \frac{\mathrm{Var}[X]}{E[X]^2} = 1 + \mu^2 \sum_{i=1}^{k} \sum_{j=1}^{k} a_i a_j \left(\frac{1}{\mu_i} - \frac{1}{\mu_j} \right)^2 \geq 1$$

- <u>*Cox distribution of order k*</u> ($k \geq 1$)
  A variable with Cox distribution of order k is the passage time through the network shown in Fig. 9.4. All phases have exponential distributions with parameters $\mu_1, \ldots, \mu_k$. Phase 1 always occurs. Phase $i+1$ follows phase i ($i \geq 1$) with probability a_i, while with probability $1 - a_i$ phase i is the last phase. We assume $a_k = 0$.

density function $$f_X(x) = \sum_{i=1}^{k} \Big[(1-a_i) A_i \, \mu_1 e^{-\mu_1 x} * \cdots * \mu_i e^{-\mu_i x} \Big]$$

where $$A_1 = 1, \qquad A_i = \prod_{j=1}^{i-1} a_j, \qquad i = 2, \ldots, k$$

mean value $$E[X] = \sum_{i=1}^{k} \frac{A_i}{\mu_i}$$

variance $$\mathrm{Var}[X] = \sum_{i=1}^{k} \frac{A_i}{\mu_i^2}$$

coefficient of variation $$C_X^2 = \frac{\mathrm{Var}[X]}{E[X]^2} = \frac{\sum_{i=1}^{k} A_i/\mu_i^2}{\left(\sum_{i=1}^{k} A_i/\mu_i\right)^2}$$

Using the Cox distribution (similarly to a linear combination of Erlang distributions), it is possible to approximate any distribution $F_X(x)$ of a non-negative random variable with finite mean value; cf. [36]. Namely, for every $\varepsilon > 0$ there exists a Cox distribution $F_C(x)$ such that

$$\int_0^\infty |F_X(x) - F_C(x)| \, dx < \varepsilon$$

and

$$\left| \int_0^\infty x \, [F_X(x) - F_C(x)] \, dx \right| < \varepsilon.$$

However, there is no general prescription for how to choose the order of the distribution and its coefficients. In practice, a second-order Cox distribution is often used, whose parameters are calculated on the basis of a fixed mean value $1/\mu$ and squared coefficient of variation C_X^2:

$$\mu_1 = 2\Theta\mu, \qquad \mu_2 = 2(1-\Theta)\mu, \qquad a = 1 - \Theta - \frac{(1-\Theta)^2}{\Theta},$$

where

$$\Theta = \frac{1}{2}\left(1 + \sqrt{\frac{C_X^2 - 1}{C_X^2 + 1}}\right).$$

Appendix B

The Laplace transform is one of the integral transforms of the form

$$\bar{f}(s) = \int_{-\infty}^{\infty} K(s,x)\, f(x)\, dx;$$

$f(x)$ is the function being transformed (we restrict ourselves here to the one-variable case), $K(s,x)$ is called the *kernel of the transform*, $\bar{f}(s)$ is the transform of the originaal function $f(x)$.

In the case of the Laplace transform,

$$K(s,x) = \begin{cases} 0, & \text{for } x < 0, \\ e^{-sx}, & \text{for } x \geq 0, \end{cases}$$

and therefore

$$\bar{f}(s) = \mathcal{L}[f(x)] = \int_0^{\infty} f(x) e^{-sx}\, dx.$$

Examples of computing the Laplace transform ($\mathcal{L}$-transform):

- exponential function $f(x) = \lambda e^{-\lambda x}$

$$\bar{f}(s) = \int_0^{\infty} \lambda e^{-\lambda x} e^{-sx}\, dx = \lambda \int_0^{\infty} e^{-(\lambda+s)x}\, dx = -\frac{\lambda}{\lambda+s}\, e^{-(\lambda+s)x}\Big|_0^{\infty} = \frac{\lambda}{\lambda+s},$$

- Dirac delta distribution $f(x) = \delta(x)$

$$\bar{f}(s) = \int_0^{\infty} \delta(x) e^{-sx}\, dx = 1,$$

- Heaviside unit-step function $f(x) = \mathbf{1}(x)$

$$\bar{f}(s) = \int_0^{\infty} \mathbf{1}(x) e^{-sx}\, dx = \int_0^{\infty} e^{-sx}\, dx = -\frac{1}{s}\, e^{-sx}\Big|_0^{\infty} = \frac{1}{s}.$$

Some properties of the Laplace transform:

- $$\begin{aligned}
\mathcal{L}[af(x)+bg(x)] &= a\bar{f}(s)+b\bar{g}(s),\\
\mathcal{L}\left[\frac{d^n f(x)}{dx^n}\right] &= s^n\bar{f}(s)-\sum_{k=0}^{n-1}s^{n-k-1}\left.\frac{d^k f(x)}{dx^k}\right|_{x\to 0^+},\\
\mathcal{L}\left[\underbrace{\int_0^x\cdots\int_0^x}_{n\text{ times}}f(x)\,(dx)^n\right] &= \frac{\bar{f}(s)}{s^n},\\
\mathcal{L}\left[f\left(\tfrac{x}{a}\right)\right] &= a\,\bar{f}(as), \qquad a>0,\\
\mathcal{L}[\mathbf{1}(x-x_0)f(x-x_0)] &= e^{-x_0 s}\bar{f}(s), \qquad x_0\geq 0,\\
\mathcal{L}[x^n f(x)] &= (-1)^n\tfrac{d^n\bar{f}(s)}{ds^n},\\
\mathcal{L}[f(x)*g(x)] &= \bar{f}(s)\bar{g}(s),\\
\mathcal{L}\left[\tfrac{f(x)}{x^n}\right] &= \int_{s_1=s}^{\infty}ds_1\int_{s_2=s}^{\infty}ds_2\cdots\int_{s_n=s}^{\infty}ds_n\,\bar{f}(s_n),\\
\mathcal{L}\left[\tfrac{\partial f(x)}{\partial a}\right] &= \frac{\partial\bar{f}(s)}{\partial a}, \qquad a \text{ is a parameter.}
\end{aligned}$$

- $$\bar{f}(0)=\int_0^{\infty}f(x)\,dx,$$

- $$\lim_{s\to\infty}s\bar{f}(s)=\lim_{x\to 0}f(x),$$

  provided that $s\bar{f}(s)$ is analytic for $\Re(s)\geq 0$,

- $$\lim_{s\to 0}s\bar{f}(s)=\lim_{x\to\infty}f(x),$$

  provided that $s\bar{f}(s)$ is analytic for $\Re(s)\geq 0$.

If $f(x)$ is a probability density function, $f(x)=f_X(x)$, (note that in this case $\int_0^{\infty}f(x)\,dx=1$, and therefore

$$\bar{f}_X(0)=\int_0^{\infty}f_X(x)e^{-0x}\,dx=1,$$

) then the Laplace transform of the density function can be used to calculate the moments of the distribution:

$$\begin{aligned}
\left.\frac{d\bar{f}_X(s)}{ds}\right|_{s=0} &= \left.\frac{d}{ds}\int_0^{\infty}f_X(x)e^{-sx}\,dx\right|_{s=0}=\int_0^{\infty}f_X(x)(-x)\,e^{-sx}\big|_{s=0}\,dx\\
&= -\int_0^{\infty}xf_X(x)\,dx=-E[X],\\
\left.\frac{d^n\bar{f}_X(s)}{ds^n}\right|_{s=0} &= \int_0^{\infty}f_X(x)(-1)^n x^n e^{-sx}\,dx\Big|_{s=0}=(-1)^n E[X^n].
\end{aligned}$$

Examples:

- Example: $f_X(x) = \mu e^{-\mu x}$

$$\begin{aligned}
\bar{f}_X(s) &= \frac{\mu}{\mu+s}, \\
\bar{f}'_X(s) &= -\frac{\mu}{(\mu+s)^2} \qquad \Longrightarrow \qquad E[X] = -\bar{f}'_X(0) = \frac{1}{\mu}, \\
\bar{f}''_X(s) &= \frac{2\mu}{(\mu+s)^3} \qquad \Longrightarrow \qquad E[X^2] = \bar{f}''_X(0) = \frac{2}{\mu^2}.
\end{aligned}$$

- Erlang distribution of order k,

$$f_X(x) = \left(\mu e^{-\mu x}\right)^{k*} = \frac{k\mu(k\mu x)^{k-1}e^{-k\mu x}}{(k-1)!}$$

$$\begin{aligned}
\bar{f}_X(s) &= \left(\frac{\mu}{\mu+s}\right)^k, \\
\bar{f}'_X(s) &= -\frac{k\mu^k}{(\mu+s)^{k+1}} \qquad \Longrightarrow \qquad E[X] = -\bar{f}'_X(0) = \frac{k}{\mu}, \\
\bar{f}''_X(s) &= \frac{k(k+1)\mu^k}{(\mu+s)^{k+2}} \qquad \Longrightarrow \qquad E[X^2] = \bar{f}''_X(0) = \frac{k(k+1)}{\mu^2}.
\end{aligned}$$

- Hyperexponential distribution,

$$f_X(x) = \sum_{i=1}^{k} a_i \mu_i e^{-\mu_i x}$$

$$\begin{aligned}
\bar{f}_X(s) &= \sum_{i=1}^{k} a_i \frac{\mu_i}{\mu_i+s}, \\
\bar{f}'_X(s) &= -\sum_{i=1}^{k} a_i \frac{\mu_i}{(\mu_i+s)^2} \qquad \Longrightarrow \qquad E[X] = -\bar{f}'_X(0) = \sum_{i=1}^{k} \frac{a_i}{\mu_i}, \\
\bar{f}''_X(s) &= \sum_{i=1}^{k} a_i \frac{2\mu_i}{(\mu_i+s)^3} \qquad \Longrightarrow \qquad E[X^2] = \bar{f}''_X(0) = \sum_{i=1}^{k} a_i \frac{2}{\mu_i^2}.
\end{aligned}$$

In many cases, the determination of the original $f(x)$ from a known transform $\bar{f}(s)$ must be performed numerically. We have already encountered this in Section **??**, in the example of finding the original function of the diffusion-process transform representing queue length in a transient state.

For a fixed x, we may calculate $f(x)$ as

$$f(x) = \frac{\ln 2}{x} \sum_{i=1}^{N} V_i \, \bar{f}\left(\frac{\ln 2}{x}\, i\right),$$

where the coefficients V_i are given by

$$V_i = (-1)^{N/2+i} \sum_{k=\lfloor \frac{i+1}{2} \rfloor}^{\min(i,N/2)} \frac{k^{N/2}(2k)!}{(N/2-k)!\,k!\,(k-1)!\,(i-k)!\,(2k-i)!}.$$

Here $\lfloor x \rfloor$ denotes the greatest integer less than or equal to x, and N is an even number.

Theoretically, the larger N is, the greater the accuracy of the calculation. In practice, however, due to numerical errors, it is necessary to restrict oneself to $N = 18$ when using double-precision arithmetic, and to $N = 10$ when using single precision. It is best to determine N empirically, by numerically inverting the transform of a known function and comparing the algorithm's output with the exact result.

The theoretical aspects of this method are described in [34], and its numerical aspects in [113]. .

Bibliography

[1] Fluid flow analysis of red algorithm with modified weighted moving average. In *Modern Probabilistic Methods for Analysis of Telecommunication Networks. BWWQT 2013*, vol. 356 of *Communications in Computer and Information Science*. Springer, Berlin/Heidelberg, Germany, 2013, pp. 50–58.

[2] Abate, J., and Whitt, W. The fourier-series method for inverting transforms of probability distributions. *Queueing Systems 10* (1992), 5–88.

[3] Abdeljaouad, I., Rachidi, H., Fernandes, S., and Karmouch, A. Performance analysis of modern tcp variants: A comparison of cubic, compound and new reno. In *Proceedings of the 2010 25th Biennial Symposium on Communications* (Kingston, ON, Canada, 2010), pp. 80–83.

[4] Alam, M. J., and Chowdhury, T. Performance evaluation of tcp vegas over tcp reno and tcp new reno over tcp reno. *International Journal of Informatics and Visualization 3* (2019), 275–282.

[5] Atmaca, T., Czachorski, T., Jouaber, B., and Pastuszka, M. Modele interfejsu sieciowego oraz mechanizmu przesuwnego okna – wykorzystanie aproksymacji dyfuzyjnej, booktitle = Seminarium *Sieci Komputerowe*, address = Gliwice, Silesian University of Technology, year = March 1998.

[6] Baskett, F., Chandy, M., Muntz, R., and Palacios, J. Open, closed and mixed networks of queues with different classes of customers. *J. ACM 22*, 2 (1975).

[7] Blondia, C., and Casals, O. Statistical multiplexing of VBR sources: A matrix-analytic approach. *Performance Evaluation 16*, 1-3 (1992), 5–20.

[8] Braden, B., Clark, D., Crowcroft, J., Davie, B., Deering, S., Estrin, D., Floyd, S., Jacobson, V., Minshall, G., Partridge, C., Peterson, L., Ramakrishnan, K. K., Shenker, S., Wroclawski, J., and Zhang, L. Recommendations on queue management and congestion avoidance in the internet. RFC 2309, IETF, 1998.

[9] Bruell, S. C., and Balbo, G. Computational algorithms for closed queueing networks. North Holland, New York, 1980.

[10] Burke, P. The output of a queueing system. *Operations Research 4*, 6 (1956), 699–704.

[11] Buzen, J. P. Computational algorithms for closed queueing networks with exponential servers. *Comm. ACM 16*, 9 (1973).

[12] Champernowne, D. C. An elementary method of solution of the queueing problem with a single server and constant parameters. J. R. Statist. Soc. B18, pp. 125-128, 1956.

[13] Chandy, K. M., Howard, J. H., and Towsley, D. F. Product form and local balance in queuing networks. *Journal of the ACM 24*, 2 (1977), 250–263.

[14] Chandy, K. M., and Neuse, D. Linearizer: A heuristic algorithm for queueing network models of computing systems. *Comm. ACM 25*, 2 (1982).

[15] Chang, C.-J., and Wang, S.-Y. Performance analysisi of a statistical multiplexer for integrated service in the customer-premise equipment. *Computer Networks and ISDN Systems 25* (1992), 191–201.

[16] Cheng, A. H.-D., and Sidauruk, P. Approximate inversion of the laplace transform. *The Mathematical Journal 4* (1994). issue 2, pp. 76-81.

[17] Cohen, J. W. The single server queue. North Holland, Amsterdam, 1969.

[18] Cox, R. P., and Miller, H. D. The theory of stochastic processes. Chapman and Hall, London, 1965.

[19] Czachorski, T. Przyblizone rozwiazania dokladnych modeli i dokladne rozwiazania modeli przyblizonych w ocenie efektywnosci zlozonych ukladow komputerowych: model oparty o symulacje. Podstawy Sterowania, tom 9, z. 2, s. 125-138, 1979.

[20] Czachorski, T. Zadanie konserwatora – opis przy pomocy aproksymacji dyfuzyjnej. Podstawy Sterowania, t. 14, z. 4, pp. 301-328, 1984.

[21] Czachorski, T. Aproksymacja dyfuzyjna rozkladu czasu czekania w stanowisku $m/g/1$. Podstawy Sterowania, t. 15, z. 1-2, pp. 65-90, 1985.

[22] Czachorski, T. Proces powrotow elementarnych w doskonaleniu aproksymacji dyfuzyjnej jako metody modelowania pracy systemow komputerowych. Zeszyty Naukowe Politechniki slaskiej, z. 9, Gliwice 1987 (praca habilitacyjna), 1987.

[23] Czachórski, T. A method to solve diffusion equation with instantaneous return processes acting as boundary conditions, howpublished = bulletin of polish academy of sciences, technical sciences **41** (1993), no. 4, 1993.

[24] Czachórski, T., Fourneau, J. M., and Pekergin, F. Diffusion Models to Study Nonstationary Traffic and Cell Loss in ATM Networks, booktitle = ACM 2nd Workshop on ATM Networks.

[25] Czachórski, T., and Pekergin, F. Transient diffusion analysis of cell loses and ATM multiplexer behaviour under correlated traffic. In *5th IFIP Workshop on Performance Modelling and Evaluation of ATM Networks* (Ilkley, UK, 21-23, 1997).

[26] Czachorski, T., and Pekergin, F. Dyfuzyjny model przelacznika sieciowego uwzgledniajacy zmienny w czasie i skorelowany strumien pakietow. In *Seminarium Sieci Komputerowe* (Gliwice, 18 – 19 marca 1998 r, 1998).

[27] Denning, P. J., and Kahn, K. C. Some distribution free properties of throughput and response time. Tech. Rep. Computer Science Department, Report CSD-TR-159, Purdue University, 1975.

[28] Duda, A. Diffusion approximations for time-dependent queueing systems. IEEE J. on Selected Areas in Communications, Vol. SAC-4, no. 6, 1986.

[29] Elwalid, A. I., and Mitra, D. Stochastic Fluid Models in the Analysis of Access Regulation in High Speed Networks. pp. 1626-1633, GLOBECOM, 1991.

[30] et al., E. D. L. *Quantitative System Performance, Computer System Analysis Using Queueing Network Models.* Prentice Hall, Englewood Cliffs, New Jersey, 1984.

[31] Filipiak, J. Dynamiczna teoria przeplywow sieciach. Zeszyty Naukowe Akademii Gorniczo-Hutniczej, nr. 898, 1983.

[32] Filipiak, J., and Pach, A. R. Selection of coefficients for a diffusion-equation model of multi-server queue, performance 84. In *Proc. of The 10-th International Symposium on Computer Performance* (North Holland, Amsterdam, 1984).

[33] Floyd, S., and Jacobson, V. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking 1* (1993), 397–413.

[34] Gaver, D. P. Observing stochastic processes. *and approximate transform inversion* (1966). Operations Research, vol. 14. no. 3, pp. 444-459.

[35] Gaver, D. P. Diffusion approximations and models for certain congestion problems. *Journal of Applied Probability 5* (1968), 607–623.

[36] Gelenbe, E. On approximate computer systems models. *J. ACM 22*, 2 (1975).

[37] Gelenbe, E. A non-markovian diffusion model and its application to the approximation of queueing system behaviour. Tech. rep., IRIA Rapport de Recherche no. 158, 1976.

[38] Gelenbe, E. Probabilistic models of computer systems. part ii. Acta Informatica, vol. 12, pp. 285-303, 1979.

[39] Gelenbe, E., Mang, X., and Feng, Y. A diffusion cell loss estimate for atm with multiclass bursty traffic. in: D. D. Kouvatsos (Editor), Performance Modelling and Evaluation of ATM Networks, Vol. 2, Chapman and Hall, London, 1996.

[40] Gelenbe, E., and Pujolle, G. The behaviour of a single queue in a general queueing network. Acta Informatica, Vol. 7, Fasc. 2, pp.123-136, 1976.

[41] Gordon, W. J., and Newell, G. F. Closed queueing systems with exponential servers. *Operations Research 15*, 2 (1967), 252–267.

[42] Harrison, P. G., and Patel, N. M. Performance modelling of communication networks and computer architectures. Addison-Wesley, Workingham, England, 1992.

[43] Hébuterne, G., and Gravey, A. A space priority queueing mechanism for multiplexing atm channels, race 1022 meeting, 24 janvier 1989. In *extended version in ITC Specialist Seminar* (Adelaide, 1989).

[44] Heffes, H., and Lucantoni, D. M. A markov modulated characterization of packetized voice and data traffic and related statistical multiplexer parformance. IEEE J. SAC, SAC-4(6), pp. 856-867, 1986.

[45] Herzog, U., Woo, L., and Chandy, K. M. Solution of queuing problems by a recursive technique. *IBM Journal of Research and Development 19*, 3 (1975), 295–300.

[46] Heyman, D. P. An approximation for the busy period of the $m/g/1$ queue using a diffusion model. J. of Applied Probility, vol. 11, pp. 159-169, 1974.

[47] Hollot, C. V., Misra, V., Towsley, D., and Gong, W.-B. Analysis and design of controllers for aqm routers supporting tcp flows. In *Proceedings of IEEE INFOCOM 2001* (Anchorage, AK, USA, Apr. 2001), vol. 3, IEEE, pp. 1726–1734.

[48] Holtsinger, D. S. *Performance Analysis of Leaky Bucket Policing Mechanisms.* PhD thesis, 1992.

[49] Hsieh, C. T., and Lam, S. S. Pam – a noniterative approximate solution method for closed multichain queueing networks. In *Proc. of ACM SIGMETRICS Conf. on Measurement and Modeling of Computer Systems* (May 24-27 1988, Santa Fe, New Mexico; Performance Evaluation Revue, Special Issue, vol. 16, no. 1, pp. 261-269, 1988).

[50] Iglehart, D. Weak convergence in queueing theory. Advances in Applied Probability, vol. 5, pp. 570-594, 1973.

[51] Iglehart, D., and Whitt, W. Multiple channel queues in heavy traffic. Part I-III, Advances in Applied Probability, vol. 2, pp. 150-177, 355-369, 1970.

[52] Jackson, J. R. Jobshop-like queueing systems. Management Science, vol. 10, pp. 131-142, 1963.

[53] Jain, R. *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation and Modeling.* Wiley, New York, NY, USA, 1991.

[54] Jones, S. K., Cavin, R. K., and Johnston, D. A. An efficient computational procedure for the evaluation of the $m/m/1$ transient state occupancy probabilities. *IEEE Trans. on Comm. COM-28*, 12 (1980), 2019–2020.

[55] Jouaber, B., Atmaca, T., Pastuszka, M., and Czachórski, T. Modelling the sliding window mechanism. ICC'98, Atlanta, Georgia, USA, czerwiec, 1998.

[56] Kanagarathinam, M. R., Singh, S., Sandeep, I., Kim, H., Maheshwari, M. K., Hwang, J., Roy, A., and Saxena, N. Nexgen d-tcp: Next generation dynamic tcp congestion control algorithm. *IEEE Access 8* (2020), 164482–164496.

[57] Kendall, D. G. Some problems in the theory of queues. J. of the Royal Statistical Society, Ser. B, vol. 13, pp. 151-185, 1951.

[58] Kingman, J. F. C. Some inequalities for the queue $gi/g/1$. Biometrika, Vol. 49, pp. 315-324, 1962.

[59] KINGMAN, J. F. C. A martingale inequality in the theory of queues. In *Proc. Camb. Phil. Soc.* (Vol. 59, pp. 359-361, 1964).

[60] KINGMAN, J. F. C. Inequalities in the theory of queues. J. R. Statist. Soc. Vol. B32, pp. 102–110, 1970.

[61] KLEINROCK, L. *Queueing Systems, vol. I: Theory.* Wiley, New York, 1976.

[62] KLOUL, L. Méthodes d'évaluation des performances pour les réseaux ATM. Thèse de doctorat de l'Université Versailles-Saint-Quentin, Versailles, 1996.

[63] KNUTH, D. E. The art of computer programming. vol. 1, Fundamental Algorithms, Addison-Wesley, Reading, Mass., 1973.

[64] KOBAYASHI, H. Application of the diffusion approximation to queueing networks, part 1: Equilibrium queue distributions. *J.ACM 21*, 2 (1974), 316–328. Part 2: Nonequilibrium distributions and applications to queueing modeling, J.ACM, vol. 21, no. 3, pp. 459-469.

[65] KOBAYASHI, H., AND REN, Q. A Diffusion Approximation Analysis of an ATM Statistical Multiplexer with Multiple Types of Traffic. part i: Equilibrium State Solutions, booktitle = proc. of ieee international conf. on communications.

[66] KOTIAH, T. C. T. Approximate transient analysis of some queueing systems. *Operations Research 26*, 2 (1978), 334–346.

[67] KRÄMER, W., AND BELZ, M. L. Approximate formulae for the delay in the queuing system gi/g/1. Preprints of ITC 8, pp. 235–1, 235–8, Melbourne, 1976.

[68] KRÖNER, H., HÉBUTERNE, G., BOYER, P., AND GRAVEY, A. Priority management in ATM switching nodes. IEEE J. Selected Areas Comm, vol. 9, no. 3, pp. 418-427, 1991.

[69] KULKARNI, L. A. Transient behaviour of queueing systems with correlated traffic. *Performance Evaluation 27-28* (1996), 117–146.

[70] LAM, S. S., AND LIEN, Y. L. Tree convolution algorithm. Tech. Rep. Dept. of Comp. Sci., Research Report TR165, University of Texas, Austin, 1981.

[71] LAVENBERG, S. S., AND REISER, M. Stationary state probabilities at arrival instants for closed queueing networks with multiple types of customers. J. Appl. Probab., vol. 17, no. 4, 1980.

[72] LEE, D.-S., AND LI, S.-Q. Transient analysis of multi-sever queues withMarkov-modulated Poisson arrivals and overload control. *Performance Evaluation 16* (1992), 49–66.

[73] LI, S., AND MARK, J. Traffic characterization for integrated services network. *IEEE Trans. on Communications 38*, 8 pp. 1231-1243 (1990).

[74] LINDLEY, D. V. The theory of queues with single server. In *Proc. of Cambridge Phil. Soc.* (Vol. 48, pp. 277–289, 1952).

[75] LITTLE, J. D. C. A proof for the queueing formula $l = \lambda w$. Operations Res., vol. 9, pp. 383-387, 1961.

[76] Liu, Y., Lo Presti, F., Misra, V., Towsley, D., and Gu, Y. Fluid models and solutions for large-scale ip networks. In *Proceedings of the 2003 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems* (San Diego, CA, USA, 2003), SIGMETRICS '03, Association for Computing Machinery, pp. 91–101.

[77] Maglaris, B., Anastassiou, D., Sen, P., Karlsson, G., and Rubins, J. Performance models of statistical multiplexing in packet video communications. *IEEE Trans. on Communications 24559 36*, 7 pp. 834-844 (1988).

[78] Marshall, K. T. Bounds for some generalizations for the gi/g/1 queue. *Operations Research 16* (1968), 841–848.

[79] Marshall, K. T. Some inequalities in queuing. *Operations Research 16* (1986), 651–665.

[80] Merle, D. Contribution a l'étude d'un analyseur de modèles à réseaux de files d'attente. algorithme de calcules et applications, Thèse de docteur ingénieur, Institut National Polytechnique de Lorraine, 1978.

[81] Meyer, J. F., Montagna, S., and Paglino, R. Dimentioning of an ATM switch with shared buffer and threshold priority. *Computer Networks and ISDN Systems 26* (1993), 95–108.

[82] Misra, V., Gong, W.-B., and Towsley, D. Fluid-based analysis of a network of aqm routers supporting tcp flows with an application to red. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication* (Stockholm, Sweden, 2000), SIGCOMM '00, Association for Computing Machinery, pp. 151–160.

[83] Mohamed, K., Hussein, S., Abdi, A., and Seddiq, A. E. Studying the tcp flow and congestion control mechanisms impact on internet environment. *International Journal of Computer Science and Information Security 16* (2018), 174–179.

[84] Moran, P. A. P. A probability theory of dams and storage systems. *Australian Journal of Applied Science* (1954), 116–124.

[85] Muntz, R. R., and Wong, J. W. Asymptotic properties of closed queueing network models. In *Proc. of 8-th Princeton Conference on Information Sciences and Systems* (1974).

[86] Nagai, K., Akimaru, H., and Okuda, T. A simplified performance evaluation for bursty multiclass traffic in atm systems. *IEEE Transactions on Communications* (1994). COM-42 No. 5, pp. 2078-2083.

[87] Newell, G. F. Queues with time-dependent rates. Part I: The transition through saturation; Part II: The maximum queue and return to equilibrium;, Part III: A mild rush hour, J. Appl. Prob. vol. 5, pp. 436-451, 579-590, 591-606. 1968.

[88] Newell, G. F. Applications of queueing theory. Chapman and Hall, London, 1971.

[89] Noetzel, A. S. A generalized queueing discipline for product form network solutions. *J. ACM 26*, 4 (1979), 779–793.

[90] Nycz, M., and Czachórski, T. Modelowanie dynamiki nateżenia przesyów tcp/ip. In *Zastosowania Internetu*, P. Pikiewicz, Ed. Wydawnictwo WSB, Dabrowa Górnicza, Poland, 2012.

[91] Nycz, T., Nycz, M., and Czachórski, T. A numerical comparison of diffusion and fluid-flow approximations used in modelling transient states of tcp/ip networks. In *Communications in Computer and Information Science*, vol. 431. Springer, Berlin, Germany, 2014, pp. 213–222.

[92] Örs, T., and Jones, S. P. W. Performance Optimization of ATM Input Control using Multiple Leaky-Buckets. In *ACM 3rd Workshop on ATM Networks* (Ilkley, 1995).

[93] Page, E. Queuing theory in or. Butterworths, London, 1972.

[94] Pollaczek, F. Über eine aufgabe der wahrscheinlichkeitstheorie i. Mathematische Zeitschrift, Vol. 32, pp. 64–100; ibid. II. pp. 729–750, 1930.

[95] Pollaczek, F. Théorie analytique de problèmes stochastiques relatifs à un groupe de lignes téléphoniques avec dispositif d'attente. Gauthiers–Villars, Paris, 1961.

[96] Prabhu, N. U. *Queues and Inventories*. Wiley, New York, 1965.

[97] Pujolle, G., and Fdida, S. Modèles de systèmes et de réseaux. vol. I-II, Eyrolles, Paris, 1989.

[98] Reiser, M. Mean value analysis and convolution method for queue-dependent servers in closed queueing networks. *Performance Evaluation 1*, 1 (1981), 7–18.

[99] Reiser, M., and Kobayashi, H. Accuracy of the diffusion approximation for some queueing systems. IBM J. of Res. Develop., vol. 18, pp. 110-124, 1974.

[100] Reiser, M., and Kobayashi, H. Queueing network models with multiple closed chains: Theory and computational algorithms. IBM J. Res. Develop., vol. 19, 1975.

[101] Reiser, M., and Lavenberg, S. S. Mean value analysis of closed multichain queueing networks. *J. ACM 27*, 2 (1980).

[102] Reiser, M., and Sauer, C. H. *Queueing Network Models: Methods of Solution and their Program Implementation, w: K.M.Chandy, R.T.Yeh (edts.), Current Trends in Programming Methodology, vol. 3*. Prentice Hall, 1978.

[103] Rider, K. L. A simple approximation to the average queue size in the time-dependent m/m/1 queue. *J. ACM 23*, 2 (1976), 361–367.

[104] Roberts, J., Mocci, U., and [Eds.], J. V. Broadband network teletraffic. Tech. Rep. Springer, Final Report of Action COST 242, Berlin, 1996.

[105] Rose, O. Statistical properties of mpeg video traffic and their impact on traffic modeling in atm systems. Tech. Rep. University of Wïtzburg, Research report no.101, Institute of Computer Science, 1995 oraz dane pomiarowe dostepne pod adresem `http:/www-info3.informatik.uni-wuerzburg.de/~rose`, 1995.

[106] Ross, S. M. Bounds on the delay distribution in gi/g/1 queues. *Journal of Applied Probability 11* (1974), 417–421.

[107] Saedi, T., and El-Ocla, H. Tcp cerl+: Revisiting tcp congestion control in wireless networks with random loss. *Wireless Networks 27* (2021), 423–440.

[108] Sakumoto, Y., Ohsaki, H., and Imase, M. Design and implementation of flow-level simulator fsim for performance evaluation of large scale networks. *International Journal of Computer Science and Telecommunications 4* (2013), 1–10.

[109] Schweitzer, P. Approximate analysis of multiclass closed networks of queues. In *Proc. Int. Conf. Stoch. Contr. and Optim.* (Amsterdam, 1979).

[110] Sevcik, K. C., and Mitrani, I. The distribution of queueing network states at input and output instants. *w: M.Arato et al. (edts.)* (1979). Performance of Computer Systems, North Holland.

[111] Sherr, A. L. An analysis of time-shared computer system. Master's thesis, 1967.

[112] Sriram, K., and Whitt, W. Characterizing superposition arrival processes in packet multiplexers for voice and data. IEEE J. SAC, SAC-4(6), pp. 833-846, 1986.

[113] Stehfest, H. Algorithm 368: Numeric inversion of laplace transform. *Comm. of ACM 13*, 1 (1970), 47–49.

[114] Takács, L. *Introduction to the Theory of Queues.* Oxford University Press, New York, 1962.

[115] T.Czachorski, and F.Pekergin. Diffusion approximation of a closed network. Tech. Rep. Paris (w przygotowaniu), rapport de recherche EHEI, 1993.

[116] Török, T. L. Some approximations for gi/g/1 queues. KFKI Budapeszt, 1977, nieopublikowane, 1977.

[117] Unknown. Ccitt: Recommandation i.370. Congestion management for the ISDN Frame Ralying Bearer Service, CCITTcom XVIII; Q.922, CCITTcom XI, ????

[118] Unknown. Ccitt: Recommandation q.922. ISDN Data link layer specification for frame mode bearer services, Geneva, 1991.

[119] Veillon, F. Algorithm 486: Numerical Inversion of Laplace Transform. *Comm. of ACM 17*, 10 (1972), 587–589. 1974; takze: F. Veillon, Quelques méthodes nouvelles pour le calcul numérique de la transformé inverse de Laplace, Th. Univ. de Grenoble.

[120] Zahorian, J. *The Approximate Solution of Large Queueing Network Models.* PhD thesis, 1980.

[121] Zahorian, J., Sevcik, K. C., Eager, D. L., and Galler, B. I. Balanced job bound analysis of queueing networks. *CACM 2* (1982), 134–141. 1982.

.