Arab American University

Faculty of Graduate Studies

A Markovian model for IoT Network

By

Fadi Khaleel AbuAlrub

Supervisors

Dr. Osama Salameh

&

Prof. Mohammed Awad

This thesis was submitted in partial fulfillment of the requirements for the Master's degree in Computer Science.

March 2019

A Markovian Model for IoT Network

By

Fadi Khaleel AbuAlrub

This thesis was defended successfully on **02/02/2019** and approved by:

Committee Members	Signature
1. Supervisor: **Dr. Osama Salameh**	…………………
2. Co-Supervisor: **Prof. Dr. Mohammed Awad**	…………………
3. Internal Examiner: **Prof. Dr. Adwan Yasin**	…………………
4. External Examiner: **Prof. Dr. Khalid Khanfar**	…………………

Declaration

I declare that I have committed to all regulations, instructions, decisions and laws of the Arab American University in force and by accepted scientific and ethical standards, including the scientific secretariat, and that I bear all the consequences, results and liability if the opposite is shown including the right of the Deans' Council to cancel the degree and certificate obtained by them or I will get them based on this acknowledgment and without the right to object or appeal the decision of the Council of Deans in this regard.

Dedication

It is our genuine gratefulness and warmest regard that we dedicate this work to my parents, my wife, my children, my brothers, my sisters, and all my friends.

Acknowledgments

Dr. Osama Salameh and Prof. Dr. Mohamed Awad have been the best thesis supervisors. Their sage advice, support, and patient encouragement aided the writing of this thesis in the best ways.

Abstract

Internet of Things is one of the modern technological models that exemplify an extension of the existing heterogeneous networks. It allows the communication among human-to-human, human-to-things, and things-to-things that are incorporated into an information network allowing automatic information interchange and the processing of data at real time. This field has attracted the attention of researchers to address the potential of the IoT in different fields taking into account the different challenges that face this technology. In this thesis, we conduct a performance analysis of a real application defined through four traffic classes with priorities present in smart cities using Continuous Time Markov Chains (CTMC). Based on a finite capacity buffer system, we propose a new cost-effective analytical model with a push-out scheme in favor of highest priority (emergency) traffic. Based on the analytical model, several performance measures for different traffic classes have been studied extensively including blocking probability, push out probability, delay, channel utilization as well as overall system performance.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

IoT	Internet of Thing
HW	Hardware
SW	Software
CTMC	Continuous Time Markov Chain
FSC	Food Supply Chain
QOS	Quality of Service
PR	Pre-emptive Resume
PAN	Personal Area Networks
LAN	Local Area Network
WSN	Wireless Sensor Networks
RFID	Radio Frequency Identification
EPC	Electronic Product Code
IP	Internet Protocol
Wi-Fi	Wireless Fidelity
NFC	Near Field Communication
WiMAX	Wireless Metropolitan Area Network
GPS	Global Positioning System
GPRS	General Packet Radio Services
GSM	Global System for Mobile
WLAN	Wireless LAN
WPAN	Wireless Personal Area Network
FTP	File Transfer Protocol
GoS	Grade of Service

Chapter 1

Introduction

1 Introduction

The purpose of the internet of things is to create a common infrastructure to enable objects to communicate with each other at any time and anywhere [1] [2] [3]. Internet of things makes life smarter by adding intelligent capabilities without the total dependence on human participation. A number of applications are being developed and deployed in different industries including smart cities, food supply chain (FSC), environmental monitoring, healthcare services, surveillance, agriculture, and others. IoT seeks to achieve three main objectives. These objectives are more comprehensive interconnection, more intensive information perception, and more comprehensive intelligent service [4].

Various devices used in IoT are connected to different types of networks such as 3G networks, RFID, Wi-Fi, GPRS or GSM. This has resulted in a massive flow of heterogeneous traffic towards IoT systems that pose a challenge to buffer and service management [5] [6]. Other challenges are the complexity to represent such heterogeneous entities and the absence of procedures to ensure the Quality of Service (QoS) within IoT [7].

The devices used in IoT mostly have limited processing capabilities and cannot accommodate spectrum management solutions used in Cognitive Radio Networks [8]. Furthermore, such devices are prepared with tiny buffers which must deal with IoT services that generate a massive quantity of data. An efficient buffer system with an appropriate scheduling mechanism is, therefore, necessary to ensure immediate connectivity and meet the QoS requirements [9]. These specifications are usually determined by performance measures including queueing delay and packet loss [10]. QoS is one of the most essential metrics to verify the goodness and efficiency of IoT services. It is, therefore, needful to layout a network that can fit dynamically to the varying wants of

QoS in IoT systems [9]. A review of various QoS schemas, applications and structural designs for different Quality of Service (QoS) parameters of IoT is presented in [11].

The key to analyzing IoT performance is to understand the nature of the traffic involved in these systems. This is important in order to be able to construct an efficient model to accurately evaluate network performance and meet the QoS requirements. In this thesis, our main goal is to investigate IoT Performance depending on detailed traffic model by probabilistic methodology taking in to account the QoS requirements. Our work is motivated by the research in [8] [12] where the authors provide realistic traffic model for an IoT system. This traffic model consists of four traffic classes as follows:

• **Class-1-Traffic:** This traffic class includes packets that are delay acceptable. Examples are packets used for telemetry and FTP. It has the lowest priority but assumed to require the highest bandwidth.

• **Class-2-Traffic:** This traffic class defines packets such as RFID packets used to identify things and resources. These packets require very small bandwidth and supposedly to have non-preemptive preference over packets of class 1.

• **Class-3-Traffic:** This class defines packets used for information processing services. For example, packets used for data automation. Packets of this class are supposed to have non-preemptive preference over packets of class 2.

• **Class-4-Traffic:** This traffic class defines packets transmitted in emergency situations. It includes packets exchanged during human-machine-human services such as patient monitoring. The packets of this class have the highest priority among all traffic classes. It is assumed that the packets of this class have preemptive priority over all the traffic classes except packets of class 2. In this case, packets of class 4 have non-preemptive priority.

This assumption seems reasonable since packets of class 2 are very small in size and inefficient to interrupt.

In this thesis, we proposed a new analytical model by considering the four traffic classes described above. Packets share a single buffer with finite capacity under a push out buffer scheme that assures the proper level of QoS. Such scheme reflects the heterogeneity of the IoT devices in terms of services offered and communication requirements. The analytical model can be used to assess the performance of the intelligent gateway under different traffic circumstances to meet the QoS limitations.

1.1 Motivation

IoT is connected to different communication technologies which make the massive flow of heterogeneous traffic towards IoT systems. This may lead to:

1. Spectrum resources sharing problem
2. Increasing pressure on the available limited spectrum resources.
3. The absence of procedures to ensure the Quality of Service (QoS) within IoT.
4. The absence of analytical models that capture realistic traffic models in details.
5. Most previous work assumes that all traffics have the same priorities.
6. The absence of models that support priority based spectrum allocation.
7. The devices in IoT mostly have limited processing capabilities.

1.2 Problem

- Understand the performance of the proposed complicated system using our analytical model?

- How to ensure the Quality of Service (QoS) within IoT?
- How to manage the spectrum demands of IoT devices with various QoS requirements?

1.3 Contribution

We proposed a new cost-effective analytical model by considering traffic differentiation that classifies the IoT services into four types of different services which are Type-I, II, III and IV stored in one finite buffer, with blocking, hybrid of relative and preemptive priority for both buffer and server, and push-out buffer scheme that assure appropriate level of QoS. Such a mechanism must reflect the heterogeneity of the IoT devices in terms of services offered and communication requirements. The analytical model can be used to evaluate the performance of smart devices under different traffic conditions to meet the QoS constraints. In this thesis, we proposed a five-dimension Continuous Time Markov Chain (CTMC) with traffic modeling for IoT which satisfies the QoS.

Based on the analytical model, several performance measures for different traffic classes have been extensive studies including blocking probability, push out probability, delay, channel utilization as well as overall system performance.

1.4 Overview

The remainder of the thesis is structured as follows. Chapter 2 presents a background that includes an introduction to IoT, description of why IoT is important, IoT characteristics, technologies used in the implementation of IoT, IoT applications, Quality of Service (QoS), Markovian model, Continuous Time Markov Chain (CTMC) in addition to related work. In Chapter 3, we discuss the system model and model description followed by performance measures. Chapter 4 explains the experimental classification results and analysis. In Chapter 5, we conclude.

Chapter 2

Background

2 Background

In this chapter, we will introduce an overview of an IoT technology, their components, characteristics, applications, in addition, to explain the Quality of Service (QoS), Markovian model and Continuous-Time Markov Chain (CTMC).

2.1 Internet of Things

Internet of Things is also referred to as M2M (machine-to-machine), internet of everything. It is one of the modern technological models that exemplify an extension of the existing heterogeneous networks.

The diagram shown in Figure 1 shows the tasks of IoT according to the layers in which they are formed. IoT technology consists of three tiers: sensor devices, gateways, and the data center. In the first layer, we observe that it consists of a number of embedded devices that detect and monitor objects within their surroundings. The second layer consists of a gateway that receives data from embedded devices and is more computationally powerful than embedded sensors. While the third layer contains servers or data centers that store the data received from the gateway to be processed through complex analysis by developing models that illustrate the behavior of the application [13].

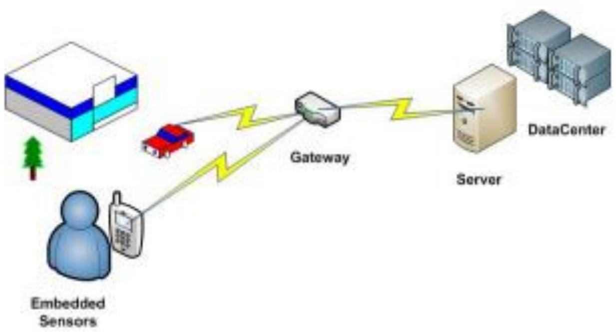


Figure 2-1: Internet of Things **[13]**

Embedded sensors (Layer 1): Through interconnected devices, this layer is responsible for data collection via sensors. These devices are connected with each other and with the gate using the Zigbee protocol, which falls within the IEEE 802.15.4 standard known as the Personal Area Networks (PAN). This protocol regulates the sensors in a star, ring or mesh topology, with a lower transmission power than those in WLAN. Due to their small size and their low cost, they can be integrated into many different devices such as laptops [13].

Gateway (Layer 2): It sometimes referred to as a control layer which acts as a connection between the embedded sensors and the back-end servers. They are often equipped with WLAN (802.11) and WPAN networks. It provides a place for pre-processing of data before being sent to the data center, reducing the amount of unnecessary data that is routed. In addition, they are computational powerful and able to transmit data over longer distances [13].

Back-end servers (Layer 3): Back-end servers consist of a variety of heterogeneous components, such as routers and data centers that typically communicate via high-bandwidth wired connections. It is capable of housing, processing and analyzing complex real-time data in addition to device management. Typically, big data computing infrastructures such as Hadoop and Map Reduce are used to conduct IoT analytics [13].

2.2 Why the Internet of Things is Important:

Imagine a world where every device is connected to the home, the workplace, the car, and the city. When the traffic light opens automatically when the car approaches it, the coffee starts to prepare when the alarm sounds. This is the kind of world that IoT can create.

In order for the Internet to be fully realized, all devices must be able to communicate with each other, regardless of the product manufacturer or companies that have business relationships with one another.

Internet of Things systems have attracted much attention in scientific and academic research over the past few years as a future technology that contributes to the creation of an intelligent environment that enables objects to interact with each other with minimal human intervention to facilitate our daily lives by providing convenience in everyday activities, energy efficiency, efficient transportation, security, reduced power, and energy consumption, smart homes; due to their diverse abilities and of the impact on the industry because of high revenues and the provision of expenditures across various industries in this area [3] [8] [14].

The main thrust of IoT research is to use its ability to track objects within a diverse environment, reduce errors and prevent theft as well as speed up operations [15].

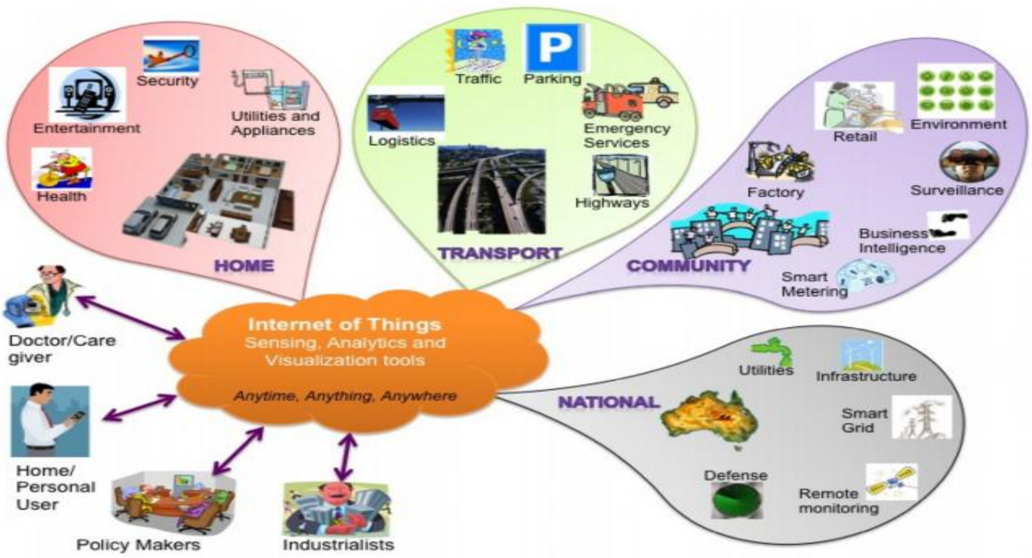


Figure 2-2: Importance of the Internet of Things **[2]**

2.3 IoT Characteristics

The IoT is a complex system with a number of characteristics. Some of the general and key characteristics identified as follows [16]:

- **Interconnectivity:** Enables IoT by connecting things to each other and allowing them to connect, access and compatible with the information network. Since simple interactions at the organism level contribute to collective intelligence in IoT network.
- **Things-related services:** IoT is capable to provide thing-related services within the restrictions of things, such as privacy protection and semantic consistency between physical things and their related virtual things. To do that a lot of changes are needed in the information world.
- **Heterogeneity:** Devices within the IoT environment are heterogeneous on the basis of different hardware platforms and networks so they can interact with other devices or service platforms across different networks. IoT architecture should support network connectivity between heterogeneous networks. The main design requirements for heterogeneous things and their environments in IoT are scalability, modularity, extensibility, and interoperability.
- **Dynamic changes:** The primary activity of IoTs is to collect data from their environment, and this is achieved through dynamic changes that occur around the devices. The state of these devices changes dynamically and efficiently, such as sleep, wake up, connection and/or disconnection as well as the context of devices including temperature, location, and speed. In addition to changing the number of devices dynamically with a person, place and time.
- **Enormous scale:** The number of devices to be managed and connected to each other is much larger than the devices connected to the current Internet. So that their

management and interpretation for purposes of the application becomes more important.

2.4 Enabling Technologies Used in Implementation of the Internet of Things

Enabling technologies, which are considered as an integral part of the IoT world and as the basics of communication and sensing technologies necessary to make the abundant existing content in IoT. An example of such enabling technologies include Wireless Sensor Networks (WSN), Radio Frequency Identification (RFID), Electronic Product Code (EPC), Barcode, Internet Protocol (IP), Bluetooth, Wireless Fidelity (Wi-Fi), data storage and analytics, wireless wide area network (3G/4G mobile networks), ZigBee, Near Field Communication (NFC), wireless metropolitan area network (WiMAX), satellite network (GPS), addressing , visualization, an actuator and cloud computing [2] [5].

2.5 IoT Applications

IoT is now part of every aspect of our lives. Not only are IoT applications that enhance the conveniences of our lives but also they give us more control by simplifying routine work life and personal tasks.

Despite advances in IoT applications, these applications are still in their early stages. However, the use of IoT is evolving rapidly and growing. A number of Internet applications are being developed and / or deployed in various industries including inventory, production management, Food Supply Chain (FSC), environmental monitoring, health care services, security, surveillance, intelligent traffic information and monitoring systems, transportation, agriculture, workplace, home support, home automation, emergency management processes and response times such as fires, floods, earthquakes and so on [17] [18].

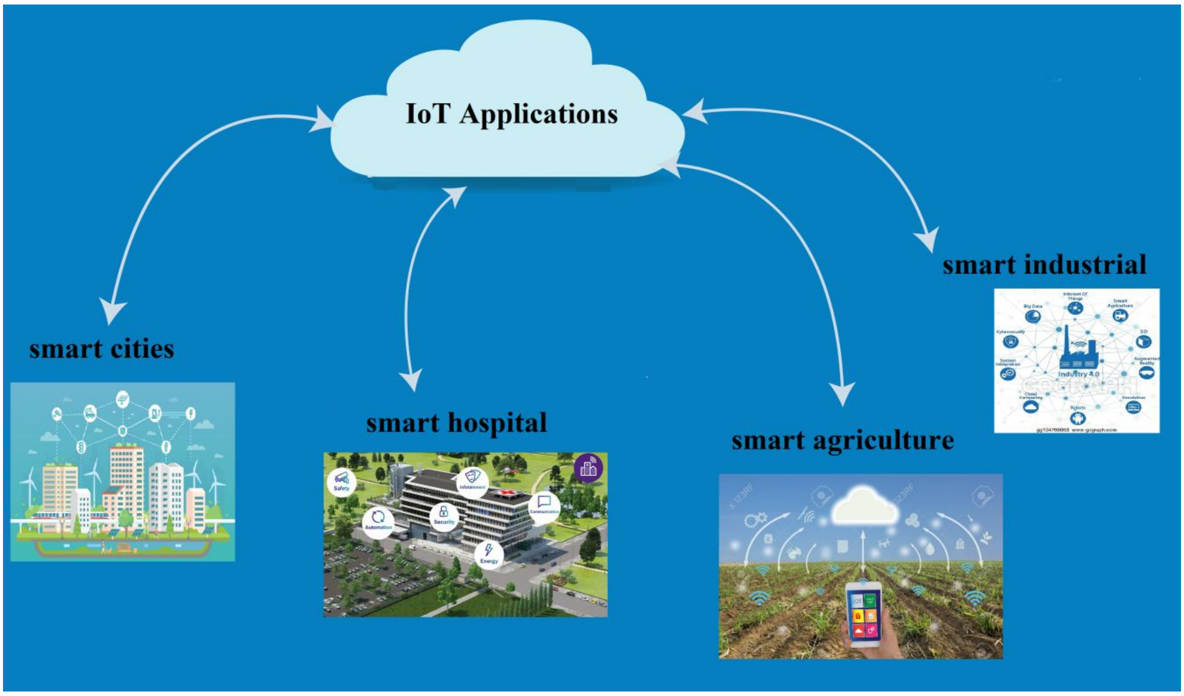


Figure 2-3: IoT Applications

2.6 Quality of Service (QoS)

Quality of Service (QoS) manages the network's capabilities and resources to provide a reliable, high-efficiency network to ensure optimal utilization of traffic within the IoT environment.

Due to the heterogeneous nature of IoT, the QoS in IoT is the ability to provide the service in an optimal manner and at a lower cost than various service providers. In order to offer secure and predictable services, QoS will manage the average number of packets, average delay, overall push out probability, the blocking probability, utilization, and performance/cost proportion (Z) in the system [9] [11] [19].

2.7 Markovian model

Markov Chain is a mathematical term that is a random process (Stochastic Process) with a Markovian property used to model randomly varying systems. It is, therefore, a method that deals with the possibility of occur a specific event in the future based on the analysis of certain possibilities, that is, a scientific method for studying and analyzing a phenomenon in the current state in order to predict its future behavior without relying on events that have occurred before (that is, it assumes the Markov property) [20].

The Markov model is graphically represented with a number of potential states of that system, representative arrows point to the transition paths among states and the rate parameters of those transformations. The diagram in **Figure 2-4** represents the Markov process for a single composition containing only two states named E and A. Each number represents the probability of changing the Markov process from one state to another, with the direction indicated by the arrow. For example, if the Markov process is in state A, the probability that changes to state E is 0.4, while the probability of being in state A is 0.6 [20].

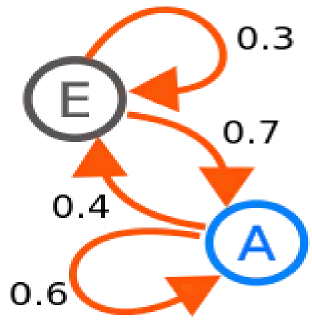


Figure 2-4: Markovian Model

2.8 Continuous-Time Markov Chains

Continuous-time Markov chains (CTMCs) are mathematical models that can describe the behavior of dynamic systems under stochastic uncertainty. In order to formulate a problem through Continuous-Time Markov Chain (CTMC), many assumptions must be met. For example, it is commonly assumed that the user is capable of specifying an accurate value for all the parameters of the model. The second important assumption is the Markov condition, which states that the future behavior of the system relies only on its current state, not in its history. Another example is homogeneity, which assumes that the dynamics of the system are independent of time, and some technical differentiability assumptions. As a result of all these assumptions, Continuous-Time Markov Chains (CTMCs) can be described in simple analytical terms [21].

In a Continuous-Time Markov Chain (CTMC), the state transitions may occur at any time, and the time among transitions is exponentially distributed. Since the exponential distribution is memoryless, the future results of the process rely only on the present state and not depend on when the last transition occurred or what any of the previous states were [21].

A continuous time Markov chain on a finite or countable set, S, is a family of random variables, $\{X(t), t \geq 0\}$ with values in $\{0, 1, 2, . . .\}$, such that "given the present, the past and future are independent", i.e., for all s, $t \geq 0$ and all states *i, j*, x(u) [21],

$$P[X(t + s) = j \mid X(s) = i, \{X(u) = x(u), 0 \leq u < s\}] = P[X(t + s) = j \mid X(s) = i] = P_{ij}(t). \quad (1)$$

The probabilities $P_{ij}(t)$ are called transition probabilities and the $|S| \times |S|$ matrix is called the transition probability matrix. The matrix P(t) is for all *t* a stochastic matrix [21].

$$P_{ij}(t) = \begin{pmatrix} P_{00}(t) & P_{01}(t) & \cdots \\ P_{10}(t) & P_{11}(t) & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

stationary transition probability function (independent of s) [21]:

$$P_{ij}(t) = P[X(t + s) = j \mid X(s) = i] \qquad (2)$$

We denote the state of the system at time t as X (t). The state probability at time t is the probability that the system is in state j at time t, and is denoted as [21]:

$$P_j(t) = Pr\{X(t) = j\} \qquad (3)$$

The steady-state or limiting probability of being in state j is [21]:

$$P_j = \lim_{t\to\infty} P_j\ (t) \qquad (4)$$

And the steady-state vector is given by [21]:

$$\vec{p} = \begin{bmatrix} p_0 & p_1 & p_2 & \cdots \end{bmatrix} \qquad (5)$$

For a Continuous-Time Markov Chain (CTMC), we can define its intensity matrix or rate matrix, Q. The elements $q_{i,j}$ of Q indicate the rate of transitions from state i to state j for $i \neq j$. In other words, the time to make a transition to state j given that the process is in state i is exponentially distributed with rate parameter $q_{i,j}$ [21]. For $i \neq j$,

$$q_{i,j} = -\sum_{i=1}^{N} q_{i,j} \qquad i \neq j, \text{N is the number states in thesystem} \qquad (6)$$

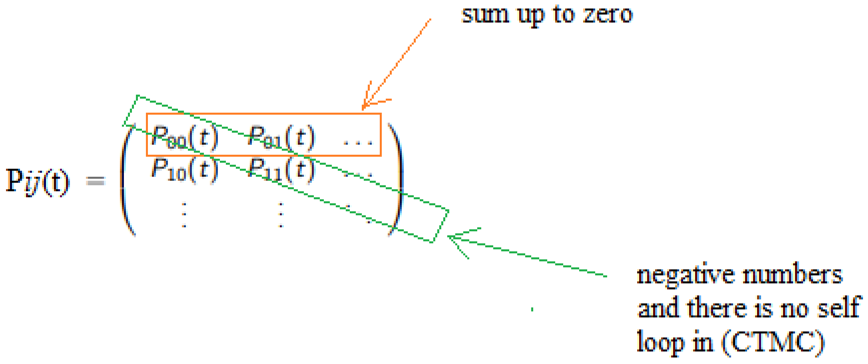
The steady-state probabilities can be found from Q using [21]:

$$\vec{p} \cdot Q = 0 \qquad (7)$$

and

$$\sum_i \mathrm{p}i = 1 \qquad (8)$$

In Continuous-Time Markov Chains (CTMC), there is no meaning for periodicity since self-loop is forbidden and so the diagonal values of Q matrix should be the sum of each matrix line will be zero.



Example: Let,

- λ_i be the arrival rate. We assume the packets arrive according to a Poisson process.
- μ_i be the transmission rate. We assume that the transmission time by the server is exponentially distributed.
- $\lambda < \mu_i$ otherwise the system is unstable.

Solution:

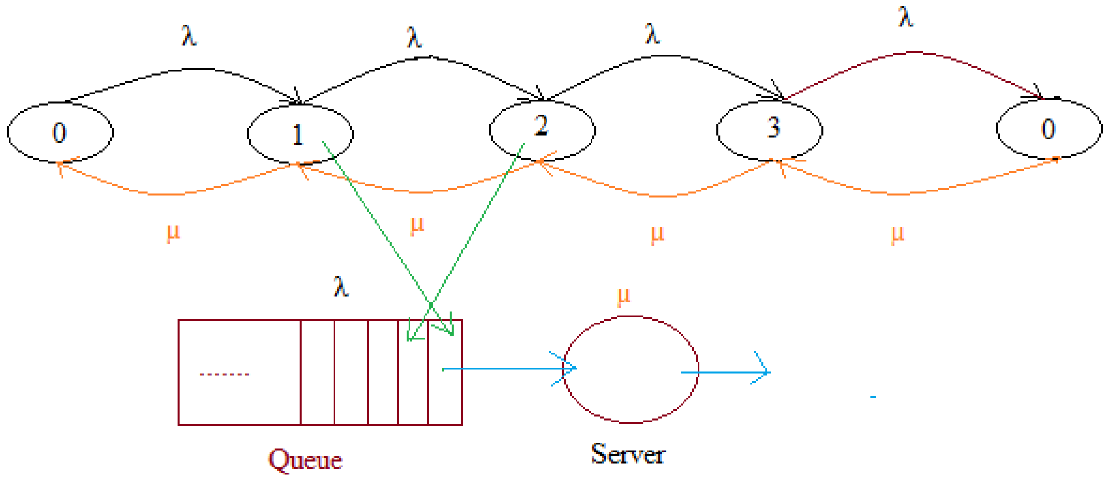


Figure 2-5: Transition diagram for Markov chain Example

Inner = outer

$$-\mathrm{P}_0\,\lambda + \mathrm{P}_1\,\mu = 0$$

$$\mathrm{P}_1 = \mathrm{P}_0\,\lambda/\mu$$

$$-\mathrm{P}_1\,(\lambda + \mu) + \mathrm{P}_0\,\lambda + \mathrm{P}_2\,\mu = 0$$

$$-\mathrm{P}_0\,\lambda/\mu\,(\lambda + \mu) + \mathrm{P}_0\,\lambda + \mathrm{P}_2\,\mu = 0$$

$$-\mathrm{P}_0\,\lambda^2/\mu - \lambda\,\mu + \mathrm{P}_0\,\lambda + \mathrm{P}_2\,\mu = 0$$

$$\mathrm{P}_2\,\mu = \mathrm{P}_0\,\lambda^2/\mu$$

$$\mathrm{P}_2 = \mathrm{P}_0\,\lambda^2/\mu^2$$

$$\mathrm{P}_3 = \frac{\mathrm{P}_0\,\lambda^3}{\mu^3} = \mathrm{P}_0\,\rho^3 \quad where\ \rho = \lambda/\mu$$

$$\mathrm{P}_3 = (1-\rho)\rho^3$$

$$\mathrm{P}_4 = \frac{\mathrm{P}_0\,\lambda^4}{\mu^4} = \mathrm{P}_0\,\rho^4 \quad where\ \rho = \lambda/\mu$$

$$\mathrm{P}_4 = (1-\rho)\rho^4$$

$$\sum_{i} \mathrm{p}i = 1 \ \textit{steady state probability vector}$$

$$\vec{p} \cdot Q = 0$$

$$\vec{\mathrm{P}} = \mathrm{P}_0 \,, \mathrm{P}_1 \,, \mathrm{P}_2 \,, \mathrm{P}_3 \,, \mathrm{P}_4$$

$$Q = \begin{bmatrix} -\lambda & \lambda & \cdots & \cdots & \cdots \\ \mu & -(\mu+\lambda) & \lambda & \cdots & \cdots \\ 0 & \mu & -(\mu+\lambda) & \lambda & \cdots \\ 0 & 0 & \mu & -(\mu+\lambda) & \lambda \end{bmatrix}$$

2.9 Related works

The IoT is a relatively new research topic and a small number of publications are devoted to performance analysis of these systems. We review some of Markov chain analysis which is conducted in [9] [22] [23] [24]. In [9], the authors consider a finite-capacity buffer with two traffic classes; normal and emergency. The emergency traffic has pre-emptive priority over normal traffic. A push-out mechanism is utilized where the arrival of an emergency packet pushes out a normal packet from the buffer if the buffer is full. Basic performance measures are obtained including buffer length and blocking probability for each traffic class. In [22], the performance of a finite capacity slot- based IoT node with three traffic classes is analyzed. The traffic classes include high, medium and low priority classes. In each slot, the scheduler allocates bandwidth to each traffic class using a round-robin policy dynamically through a prediction based approach. This study does not consider blocking and push out probabilities. In [23], the authors investigate the performance of haphazard access narrow-band Internet of Things. The system throughput is extensively analyzed. In

[24] several analytical models of prioritized contention access and CSMA/CA channel access protocols for IoT applications are presented. The authors investigate the reliability and power consumption of a node and delay for successful packet transmission.

Simulations based studies are conducted in [25] [13] [26]. In [25], hybrid scheduling is proposed to accommodate QoS constraints of heterogeneous traffics of IoT. Traffics with preemptive and non-preemptive priorities are considered. The analysis is conducted using the NS-2 simulator and concentrates only on buffer length for different traffics. Two traffic scenarios are considered in [13]. In this work, the factors of the performance of an IoT network are analyzed. Three traffic classes with priorities are considered in [26]. An efficient packet scheduler is proposed and the average packet drop ratio and average jitter for all traffic classes are analyzed.

Finite buffers are investigated using the generating function approach in [27] [28] [29]. In [27], the authors study a single server Markovian system with two classes of customers. Customers of class 2 have non-preemptive precedence over customers of class 1. The mean buffer length for each class of customers is calculated. In [28], a non-preemptive priority buffer with a push-out buffer management mechanism is studied and packet loss probabilities are obtained. A preemptive buffer with two customer classes and the randomized push-out mechanism is analyzed in [29]. In this work, the packet loss probability and average relative waiting times are thoroughly investigated.

In our work, we analyze a buffer system with realistic traffic model reported in [8] [12]. Four classes of traffic with pre-emptive and non-preemptive priorities are considered. Furthermore, a push-out mechanism is employed where a customer of a higher class pushes out a lower priority customer if the buffer is full. To the best of our knowledge, an accurate analysis of this model has been rarely seen in the literature.

In [30], the authors provided a brief overview of the most widely used classes in the Markovian models, their solutions and applications. Where they studied discrete-time Markov chains, continuous-time Markov chains, and semi-Markov chains and the relationship between them. Then, they solved a number of linear equations (both direct and iterative ones) and systems of differential equations that arise when solving the steady-state and transient behavior of Markovian models.

In [31], the authors performed a systematic mapping on QoS approaches in the IoT. They have evaluated 2809 papers with 162 chosen for the mapping. Approaches are classified by focus area, research type, contribution type, and quality factors. They identify and discuss several research gaps in the current literature.

In [32], the authors proposed a new practical quality model for IoT applications according to the quality attribute in ISO 9126. They defined four quality factors, criteria, and four metrics (Func, Rel, Eff, Por) that can be effectively used to derive the overall quality of IoT applications. The effectiveness of this model was verified through scenario-based case studies. The quality model for the IoT application using the IA-QM suggested in this paper can be measured with relative accuracy.

In [33], the authors proposed a set of redundancy mathematical models based on the Markov Chain capable of estimating the reliability, dependability, and availability of IoT applications considering redundancy aspects. The results showed advantages for integrating system redundancies, particularly in case of joint failures. Passive approaches were more satisfactory when compared to active approaches. The explanation for this behavior is that the components in the active redundancy operate in parallel and are therefore likely to fail.

In [34], the authors proposed a model for evaluating network systems in the context of failure and repair so that the proposed model was used to estimate system longevity and appropriate maintenance planning during the design or maintenance phases of the system.

Markov chains were used to model system failures states based on historical data. The effectiveness of this model was measured through preliminary experiments and case studies, which also emphasize the intuition about the effects of network topology on network reliability.

In [35], the authors introduced an approach to developing the concept of a smart home by integrating the Internet of Things (IoT) with web services and cloud computing. The approach was based on integrating intelligence into sensors and actuators using the Arduino platform, communicating with smart objects using ZigBee technology, facilitating interaction with smart things using cloud services for easy access at different locations, and improving data exchange efficiency using JSON notation. This approach has been successfully used to demonstrate services to measure home conditions, control home appliances, and control access to the home. Infrastructure can be adopted or adapted to other applications.

Chapter 3

The Proposed Method

3 The Proposed Method

In this chapter, the proposed model is illustrated; it aims to build a cost-effective analytical model by considering traffic differentiation that classifies the IoT services into four types of different services that assure appropriate level of QoS. It starts by calculating the number of states. Then constructing the state space of the system which contains all the states based on the state transition table distinguishes 7 transition cases. Then, filling the Q matrix and calculate the transition probability. Next, it illustrates the deployed model which is the CTMC, which is used to implement our 5 dimension model. Finally, it includes different metrics that were used in the evaluation process to meet the QoS constraints.

3.1 System Model

This thesis proposes a Markov-Chain based model to investigate the performance of a single server and a single buffer system with four priority classes described in section 1. A finite capacity buffer with complete buffer sharing scheme is considered for all type of classes (See **Figure. 1**). The model parameters are summarized as follows: Let,

• λ_i be the arrival rate for traffic of class i, i=1,2,3,4 where class 4 has the highest priority. We assume the packets of each class arrive according to a Poisson process.

• μ_i be the transmission rate for traffic of class i, i=1, 2, 3, 4. We assume that transmission is error free i.e. always successful unless pre-empted by a class 4 packet arrival. Also, we assume that the transmission time by the server for packets of each class is exponentially distributed.

• N be the total system capacity including that in service i.e. the maximum number of packets that may wait in the buffer including the packet in transmission.

Based on these assumptions and since the buffer is not time slotted, we use Continuous Time Markov Chain (CTMC) in our analysis.

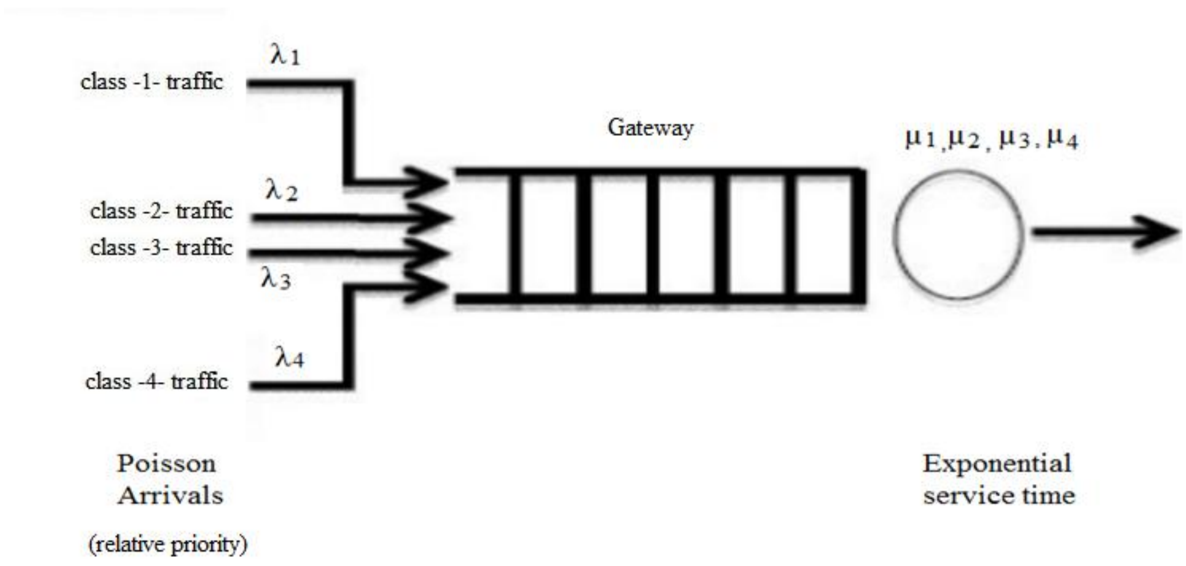


Figure 3-1: System model

Let system state be represented by x = (n_1, n_2, n_3, n_4, c) where n_1, n_2, n_3, n_4 are the number of class 1, class 2, class 3, class 4 packets in the system (including the packet in transmission) correspondingly and c = 0, 1, 2, 3, 4 is the class of the packets in transmission and 0 if the system is empty. It can be easily seen that the state transition process is a Markov chain where the next state is determined only by the current state. State space S contains all the states such that $\sum_{i=1}^{4} n \leq$ N and satisfying the conditions C

$$C = \begin{cases} c \neq i, & if\ n_i = 0, \\ c \neq 1{,}3, & if\ n_4 > 0, \end{cases} \qquad (9)$$

where the first case ensures that a packet of class i cannot be in transmission if it is not in the system and the second case states that no class 1 or 3 packets can be in transmission if there are packets of class 4 in the system. This is explained by preemptive priority of class 4 packets over the packets of classes 1, 3. The transition rates between states are presented in **Table 3-1**.

Table 3-1: Transition rates between states

Rate	From State x to State y	Condition
λ_i	$x = (\ldots, n_i, \ldots, c = 0)$ $y = (\ldots, n_i + 1, \ldots, c = i)$	$\sum_{j=1}^{4} n_j = 0$ $i = 1,2,3,4$
λ_i	$x = (\ldots, n_i, \ldots, c)$ $y = (\ldots, n_i + 1, \ldots, c)$	$\sum_{j=1}^{4} n_j < N,$ $c \neq 0,\ i = 1,2,3$
λ_4	$x = (n_1, n_2, n_3, n_4, c)$ $y = (n_1, n_2, n_3, n_4 + 1, c)$	$\sum_{j=1}^{4} n_j < N,$ $c = 2,4$
λ_4	$x = (n_1, n_2, n_3, n_4, c = 1,3)$ $y = (n_1, n_2, n_3, n_4 + 1, c = 4)$	$\sum_{j=1}^{4} n_j < N$
λ_4	$x = (\ldots, n_i, \ldots, c = 1,3,4)$ $y = (\ldots, n_i - 1, \ldots, n_4 + 1, c = 4)$	$\sum_{j=1}^{4} n_j = N, i = min(l)$ $l = 1,2,3, n_i > 0$
λ_4	$x = (\ldots, n_i, \ldots, c = 2)$ $y = (\ldots, n_i - 1, \ldots, n_4 + 1, c = 2)$	$\sum_{j=1}^{4} n_j = N, i = min(l)$ $l = 1,2,3$ $n_i > 0$ if $i = 1,3$ $n_i > 1$ if $i = 2$
μ_i	$x = (\ldots, n_i, \ldots, c = i)$ $y = (\ldots, n_i - 1, \ldots, c = j)$	$i = 1,2,3,4,\ l = 1,2,3,4$ $j = \begin{cases} max(l) & n_l > 0, \sum_{i=1}^{4} n_i > 1 \\ 0 & \sum_{i=1}^{4} n_i = 1 \end{cases}$

Next, we describe the transitions between states of the proposed CTMC model. Within this model, seven transition cases are distinguished (see **Table 3-1**). An arriving packet from any class will get immediately transmitted when the system is empty (case 1). In case 2, an arriving packet of classes 1, 2, 3 will join the buffer if the system is not empty and space is available in the buffer. An arriving packet of classes 1, 2 or 3 is lost if the system is full. An arriving packet of class 4 will join the buffer if the packet in transmission belongs to either class 2 or class 4 and the system is not full (case 3). In case 4, an arriving packet of class 4 preempts the packet in transmission if the packet belongs to class 1 or 3. In this case, the interrupted packet re-joins the buffer if a space in the buffer is available. In cases 5 and 6, an arriving packet of class 4 pushes out a packet of lower priority when the system is full. The pushed out packet is lost. The packet of the lowest priority is pushed out of the buffer (case 5). In this case, the arriving packet preempts the packet in transmission if the packet in transmission belongs to class 1 or class 3. The system is blocked if the buffer is fully occupied by packets of class 4. When a packet of class 2 is in transmission and the buffer is full, an arriving packet of class 4 joins the buffer and pushes out a packet of the lowest priority waiting in the buffer (case 6). When a packet finishes transmission, the packet of the highest priority waiting the buffer will start transmission next (case 7) and if there are no more packets in the buffer, c is set to 0.

Let S be the set of all states of the Continuous-Time Markov Chain (CTMC) described above and Q is the infinitesimal generator matrix. Then vector π with steady-state probabilities of the Continuous-Time Markov Chain (CTMC) is obtained as the solution of a set of balance equations represented by $\pi Q = 0$ using the normalization condition $\sum_{x \in S} \pi_x = 1$ where π_x represents the probability for the system to be in state x.

3.2 Performance Measures

We are interested in obtaining several performance measures. For each class, we calculate the average number of packets in the system, the blocking probability, and utilization. These measures are typical when analyzing a queuing system with finite buffer capacity. We also calculate the push out the probability for packets of classes 1, 2 and 3. Furthermore, we define a function Z to measure the overall system performance.

The average number of the packets of class i, N_i, can be calculated by

$$N_i = \sum_{x \in S} \pi_x n_i \qquad i = 1,2,3,4 \tag{10}$$

The blocking probability is the probability that an incoming packet discovers the buffer full and lost. It is important to note that packets of classes i = 1, 2, 3 views the system as blocked differently from packets of class i = 4. Let S_N denote the set of states where $\sum_{i=1}^{4} n_i = N$. The system is blocked for packets of classes i = 1, 2, 3 if the system is in one of the states S_N whereas it is blocked for packets of class i = 4 only if the system is in one of two states $S_{N,4} = (x_1, x_2)$. The states $x_1 = (0, 1, 0, N - 1, 2)$, $x_2 = (0, 0, 0, N, 4)$ denote the cases when the buffer is fully occupied by packets of class 4 and a packet of class 2 or class 4 are in transmission correspondingly. Hence, the blocking probability for class i, γ_i is the same for packets of the first three classes and can be obtained by

$$\gamma_i = \sum_{x \epsilon S_N} \pi_x \qquad i = 1,2,3 \qquad (11)$$

and the blocking probability for packets of class 4, γ_4, is

$$\gamma_4 = \sum_{x \epsilon S_{N,4}} \pi_x \qquad (12)$$

Applying little's law, the average delay of packets of class i is then calculated as

$$E_i = N_i / (\lambda_i (1 - \gamma_i)) \quad i = 1,2,3,4 \qquad (13)$$

The push out probability is the probability that a packet that is waiting in the buffer or in transmission is pushed out and lost upon the arrival of a class 4 packet when the buffer is full. The push out probability α_i is applicable for packets of classes i = 1, 2, 3 and is calculated as follows

$$\alpha_i = \sum_{x \epsilon S_{N,i}} \pi_x \lambda_4 / (\lambda_4 + \mu_c) \quad i = 1,2,3 \qquad (14)$$

Where $S_{N,i}$ is a subset of $S_N - S_{N,4}$. Namely, $S_{N,1}$ is the subset of $S_N - S_{N,4}$ with $n_1 > 0$. Likewise, $S_{N,2}$, $S_{N,3}$ are subsets of $S_N - S_{N,4}$ with $n_1 = 0$, $n_2 > 0$ and $n_1 = n_2 = 0$, $n_3 > 0$ respectively. μ_c is the transmission rate of the class of the packet currently occupying the channel. The overall push out probability α is then equaled $\alpha_1 + \alpha_2 + \alpha_3$. We are also interested in the utilization of packets of class i, U_i. This measure is obtained by

$$U_i = \sum_{x \epsilon S_i} \pi_x \quad i = 1,2,3,4 \qquad (15)$$

where S_i is the set of all states where the class of the packet in a transmission is i. The design goal is to reach a little push out the probability of packets, decrease the blocking probability and increase the utilization of our system. We define a Grade of Service (GoS) function as

$$\mathrm{GoS} = \sum_{i=1}^{4} \gamma_i + \beta\alpha \tag{16}$$

where parameter β indicates a penalty weight for pushing out the probability of packets over blocking probability of packets. This penalty seems justified as push out implementation is complex. The system will have to keep track of all the packets in the buffer in order to be able to push out the right one when needed.

We define the performance of the system and a cost function of system operation [36]

$$\mathrm{Performance} = 1/\mathrm{GoS} \tag{17}$$

$$\mathrm{Cost} = 1/\sum_{i=1}^{4} U_i \tag{18}$$

As can be seen, our objectives are to raise the performance of the system by decreasing the GoS function and to reduce the cost of the system by increasing the utilization of the system. In order to measure the overall system performance, we define a function Z by combining the equations (16), (17) [36].

$$Z = performance \;/\; Cost \tag{19}$$

The performance measures defined above allow understanding system behavior under specific input parameters including buffer size needed to ensure QoS specifications. Also, the input parameters values can be calibrated to maximize the overall system performance function Z as will be discussed below.

3.2.1 General Model Algorithms

The MC states and transitions for small buffer size (N=2) are shown **Figure 3-2** in and the general computation algorithm that was used in performing all experiments is shown in **Figure 3-3.**

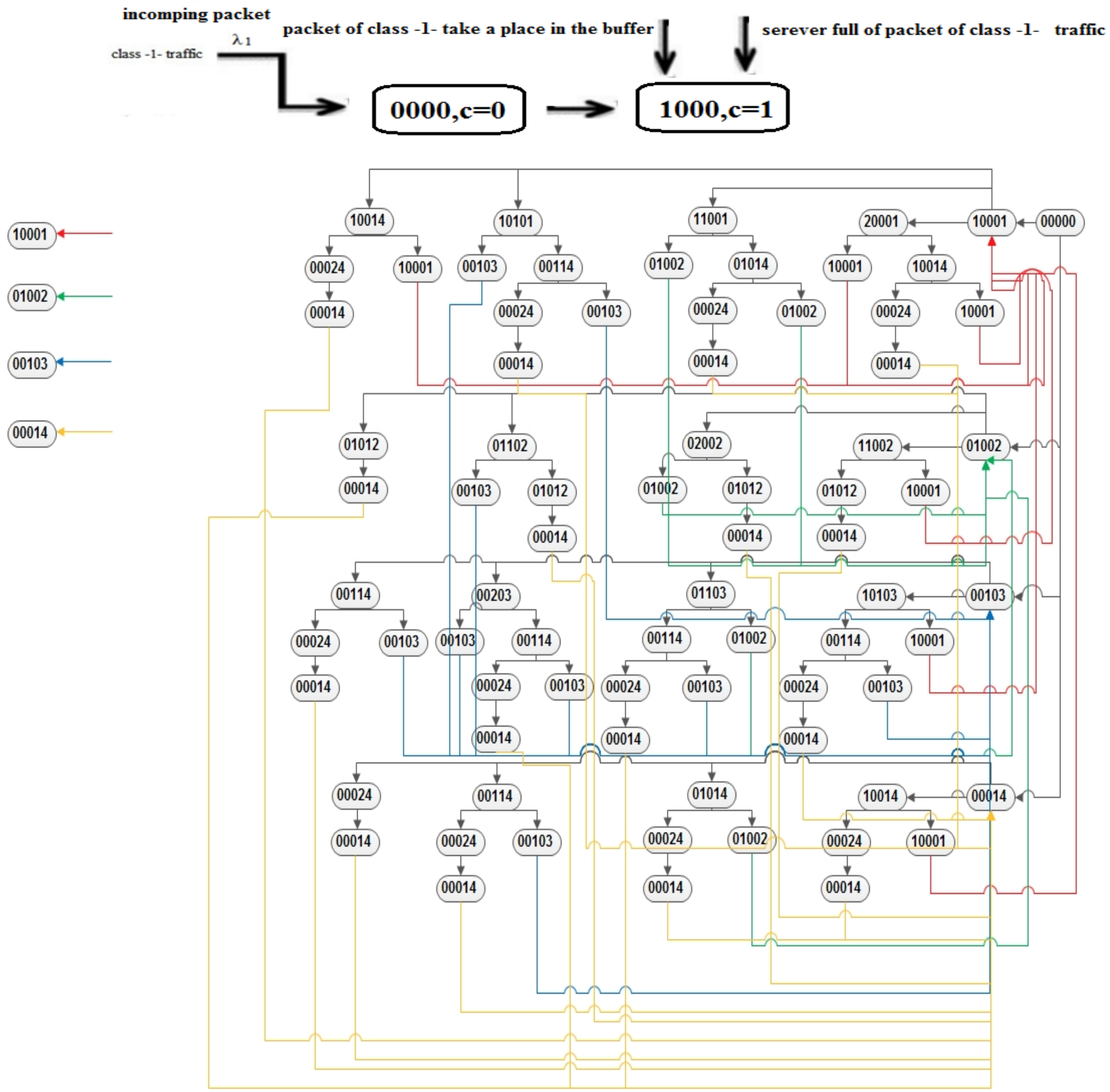
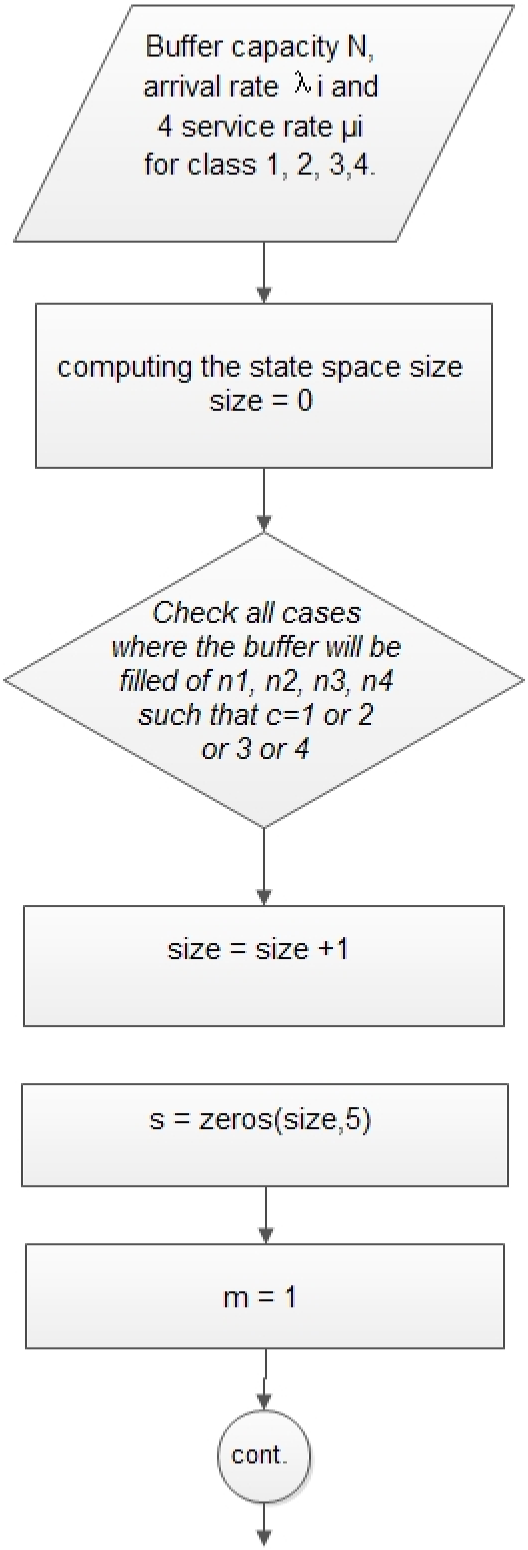
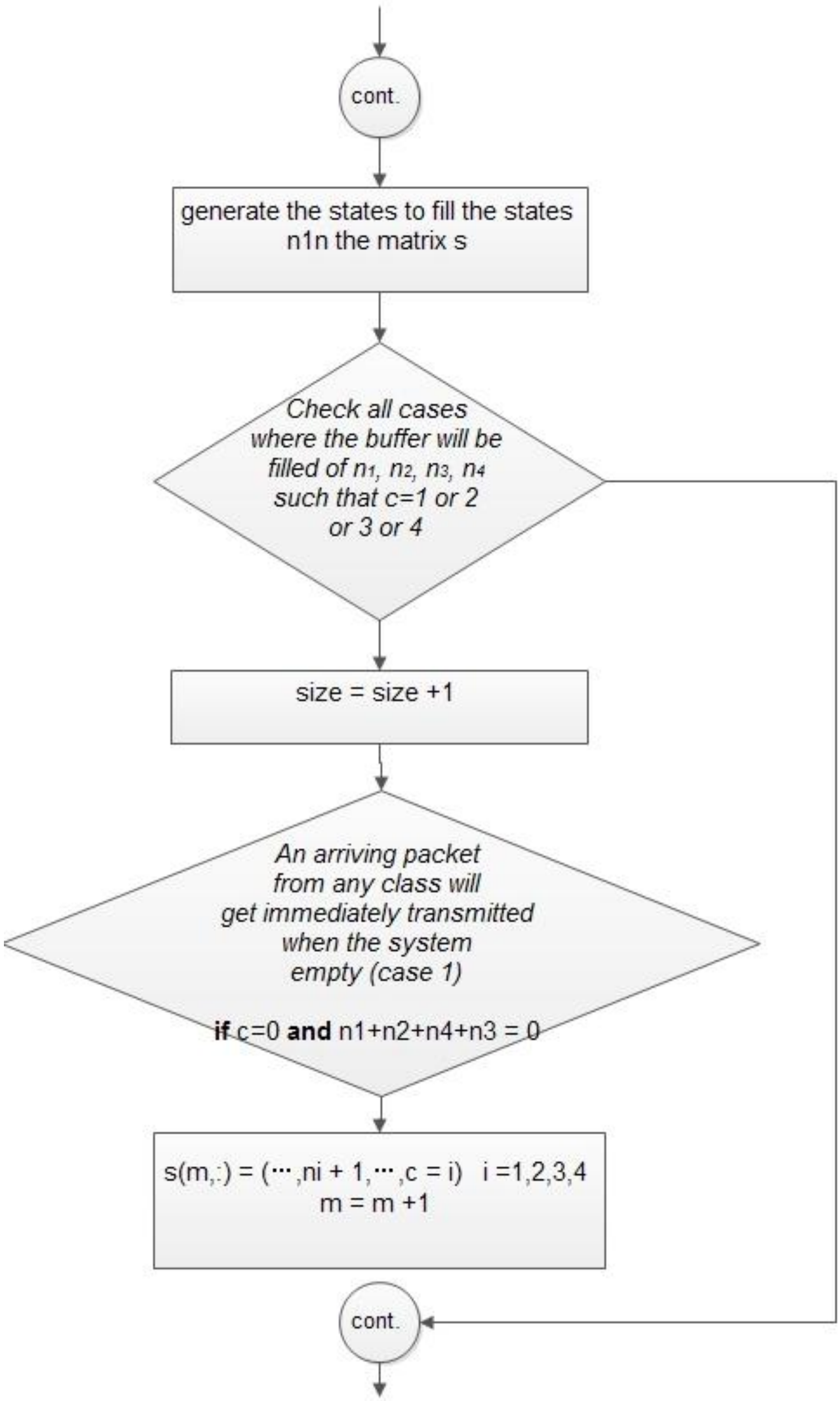
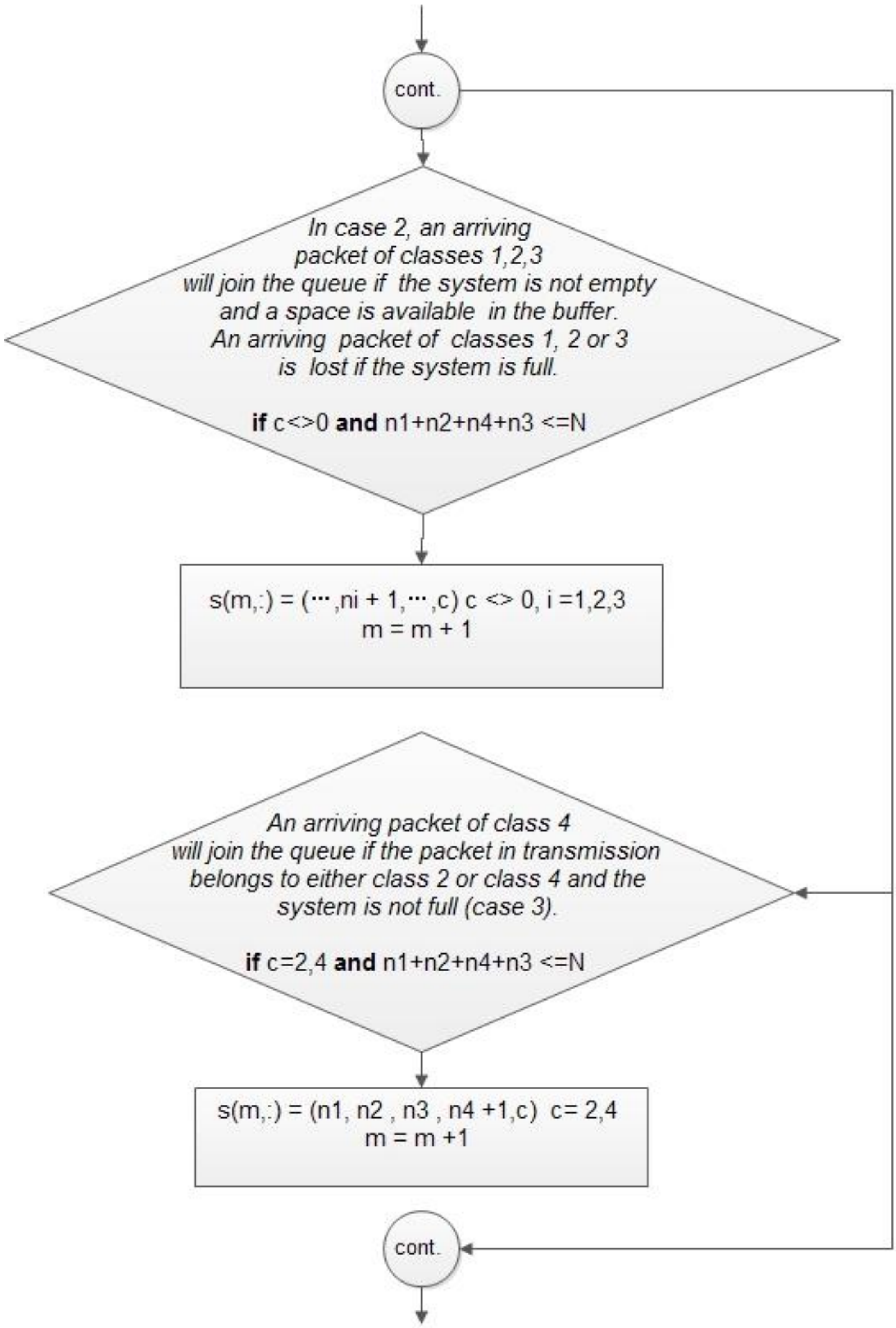
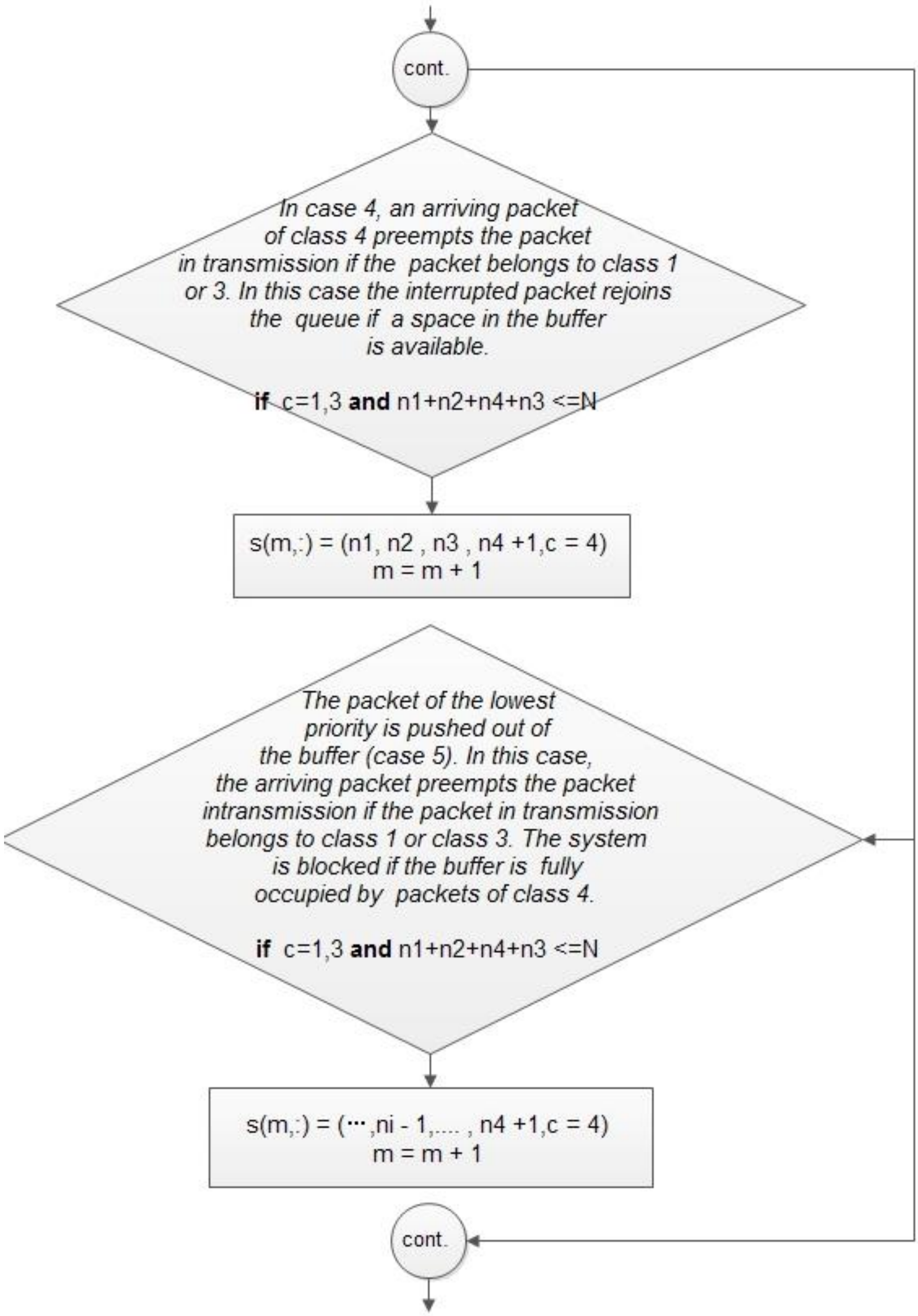


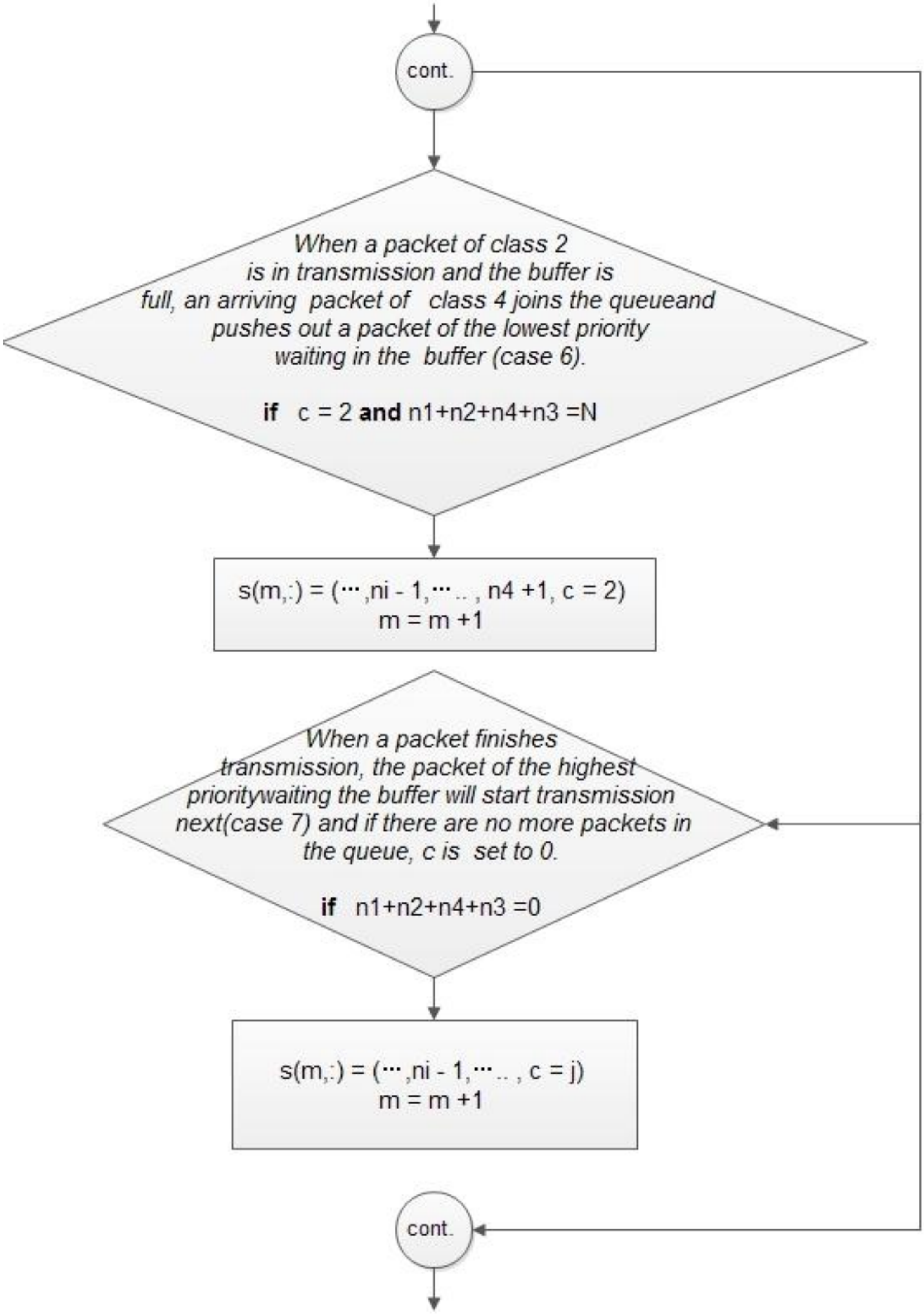
Figure 3-2: MC States Transitions

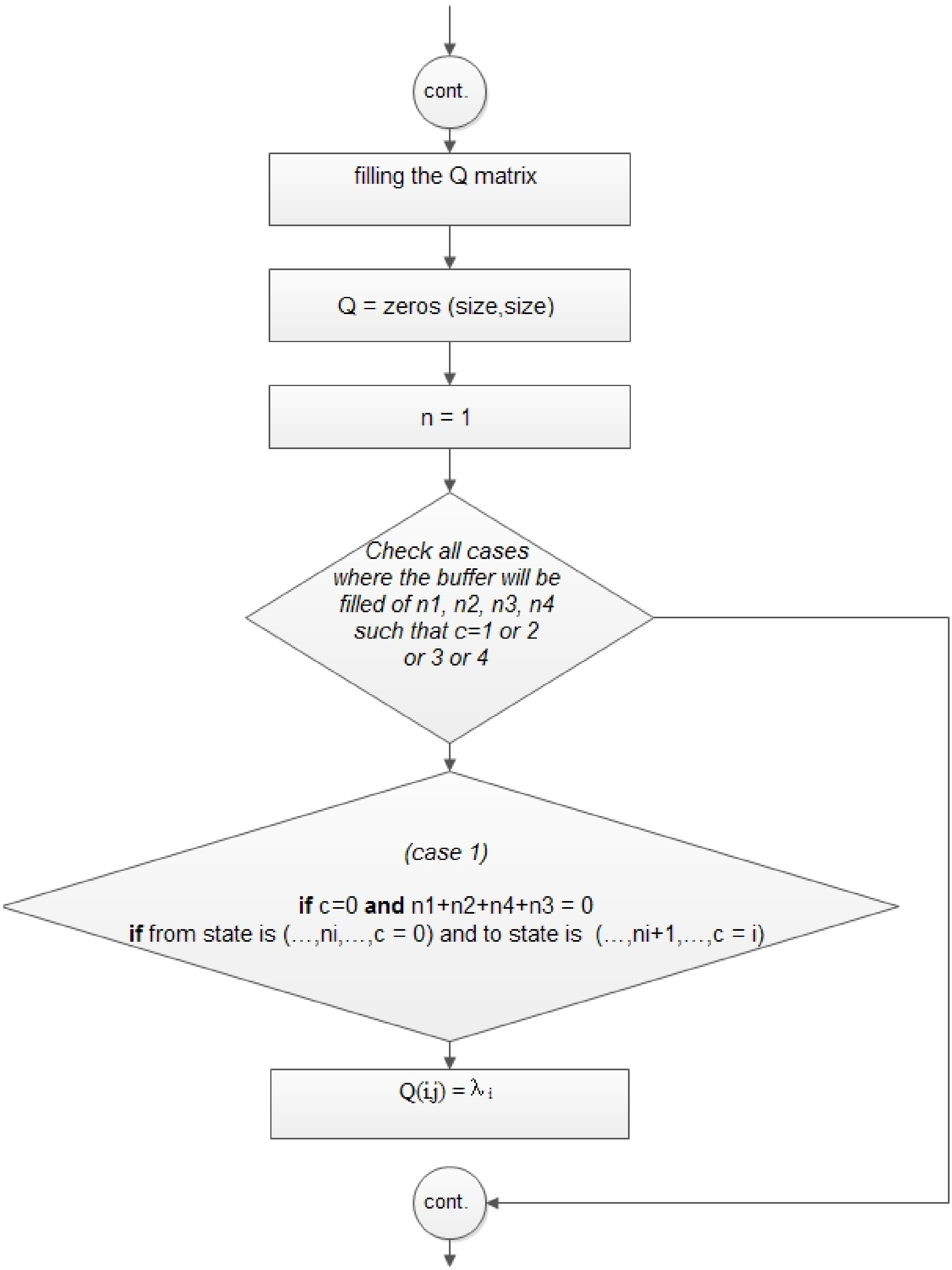

Buffer capacity N,
arrival rate λi and
4 service rate μi
for class 1, 2, 3,4.
computing the state space size
size = 0
Check all cases
where the buffer will be
filled of n1, n2, n3, n4
such that c=1 or 2
or 3 or 4
size = size +1
s = zeros(size,5)
m = 1
cont.

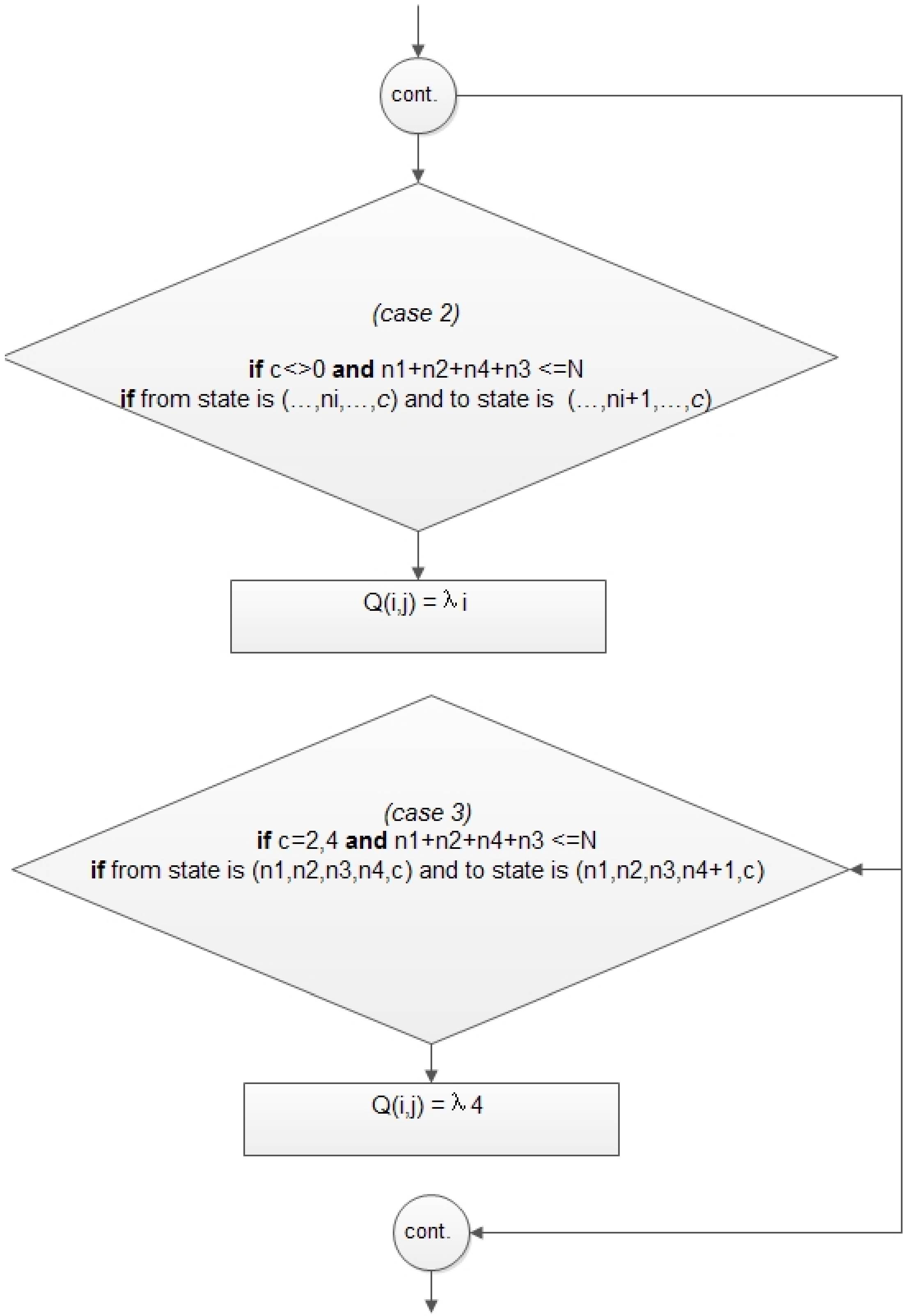

cont.
generate the states to fill the states n1n the matrix s
Check all cases where the buffer will be filled of n_1, n_2, n_3, n_4 such that c=1 or 2 or 3 or 4
size = size +1
An arriving packet from any class will get immediately transmitted when the system empty (case 1)
if c=0 **and** n1+n2+n4+n3 = 0
s(m,:) = (⋯,ni + 1,⋯,c = i) i =1,2,3,4
m = m +1
cont.

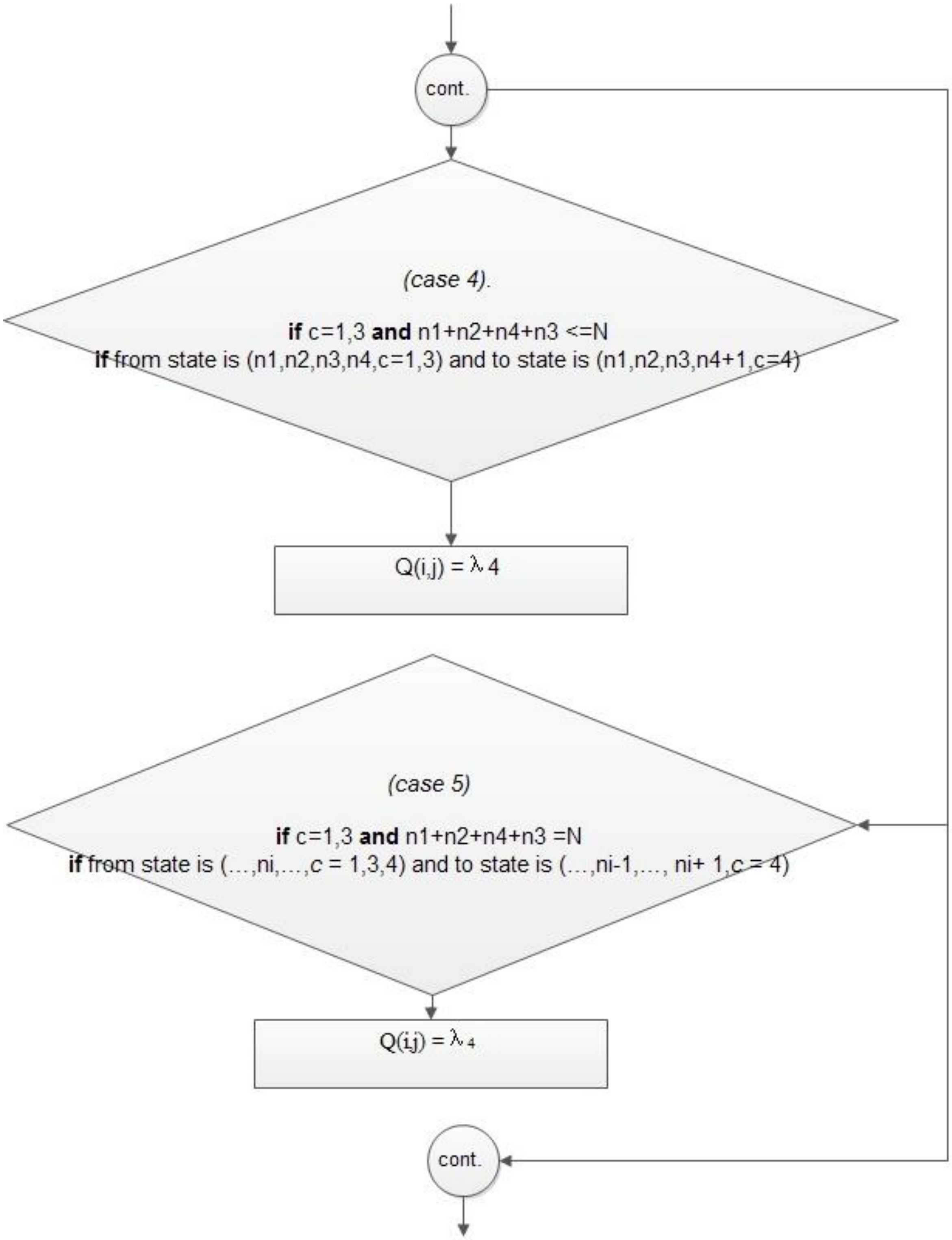

cont.
In case 2, an arriving
packet of classes 1,2,3
will join the queue if the system is not empty
and a space is available in the buffer.
An arriving packet of classes 1, 2 or 3
is lost if the system is full.
if c<>0 **and** n1+n2+n4+n3 <=N
s(m,:) = (⋯ ,ni + 1,⋯ ,c) c <> 0, i =1,2,3
m = m + 1
An arriving packet of class 4
will join the queue if the packet in transmission
belongs to either class 2 or class 4 and the
system is not full (case 3).
if c=2,4 **and** n1+n2+n4+n3 <=N
s(m,:) = (n1, n2 , n3 , n4 +1,c) c= 2,4
m = m +1
cont.

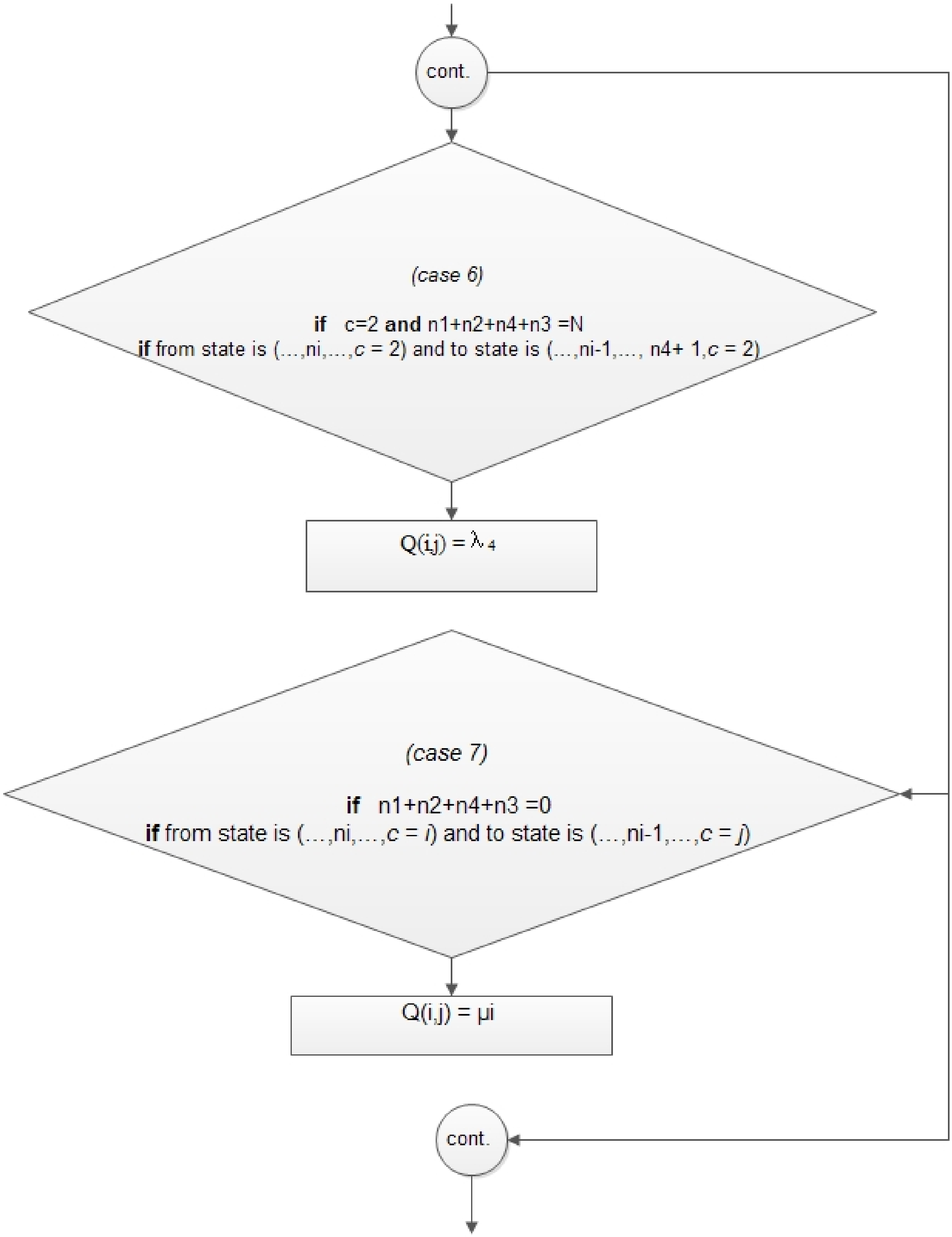

cont.
In case 4, an arriving packet of class 4 preempts the packet in transmission if the packet belongs to class 1 or 3. In this case the interrupted packet rejoins the queue if a space in the buffer is available.
if c=1,3 and n1+n2+n4+n3 <=N
s(m,:) = (n1, n2 , n3 , n4 +1,c = 4)
m = m + 1
The packet of the lowest priority is pushed out of the buffer (case 5). In this case, the arriving packet preempts the packet intransmission if the packet in transmission belongs to class 1 or class 3. The system is blocked if the buffer is fully occupied by packets of class 4.
if c=1,3 and n1+n2+n4+n3 <=N
s(m,:) = (⋯ ,ni - 1,.... , n4 +1,c = 4)
m = m + 1
cont.


cont.
When a packet of class 2
is in transmission and the buffer is
full, an arriving packet of class 4 joins the queueand
pushes out a packet of the lowest priority
waiting in the buffer (case 6).
if c = 2 and n1+n2+n4+n3 =N
s(m,:) = (⋯,ni - 1,⋯.. , n4 +1, c = 2)
m = m +1
When a packet finishes
transmission, the packet of the highest
prioritywaiting the buffer will start transmission
next(case 7) and if there are no more packets in
the queue, c is set to 0.
if n1+n2+n4+n3 =0
s(m,:) = (⋯,ni - 1,⋯.. , c = j)
m = m +1
cont.


cont.
filling the Q matrix
Q = zeros (size,size)
n = 1
Check all cases where the buffer will be filled of n1, n2, n3, n4 such that c=1 or 2 or 3 or 4
(case 1)
if c=0 and n1+n2+n4+n3 = 0
if from state is (…,ni,…,c = 0) and to state is (…,ni+1,…,c = i)
Q(i,j) = λ_i
cont.


cont.
(case 2)
if c<>0 and n1+n2+n4+n3 <=N
if from state is (…,ni,…,c) and to state is (…,ni+1,…,c)
Q(i,j) = λ i
(case 3)
if c=2,4 and n1+n2+n4+n3 <=N
if from state is (n1,n2,n3,n4,c) and to state is (n1,n2,n3,n4+1,c)
Q(i,j) = λ 4
cont.


cont.
(case 4).
if c=1,3 and n1+n2+n4+n3 <=N
if from state is (n1,n2,n3,n4,c=1,3) and to state is (n1,n2,n3,n4+1,c=4)
Q(i,j) = λ 4
(case 5)
if c=1,3 and n1+n2+n4+n3 =N
if from state is (…,ni,…,c = 1,3,4) and to state is (…,ni-1,…, ni+ 1,c = 4)
Q(i,j) = λ 4
cont.


cont.
(case 6)
if c=2 and n1+n2+n4+n3 =N
if from state is (…,ni,…,c = 2) and to state is (…,ni-1,…, n4+ 1,c = 2)
Q(i,j) = λ 4
(case 7)
if n1+n2+n4+n3 =0
if from state is (…,ni,…,c = i) and to state is (…,ni-1,…,c = j)
Q(i,j) = μi
cont.

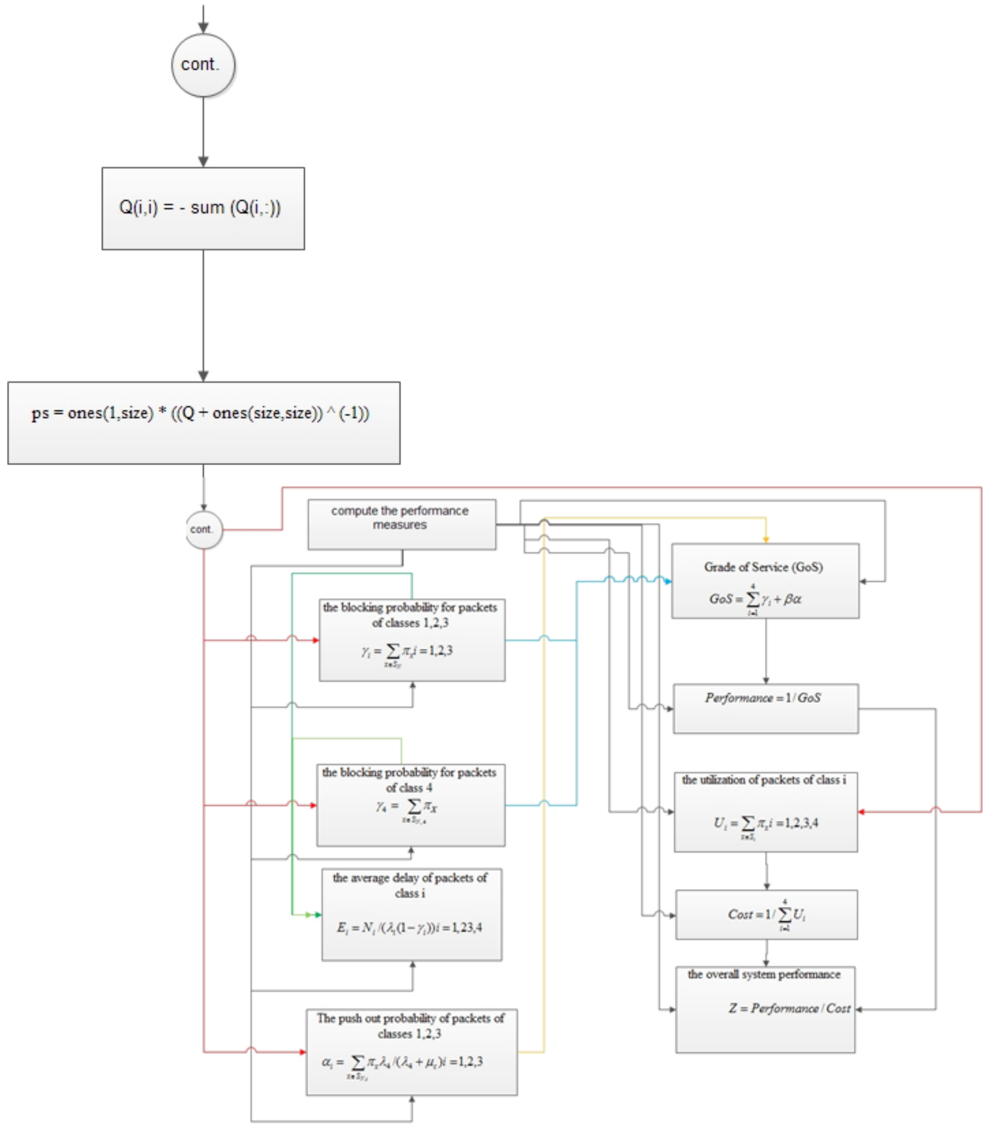


Figure 3-3: General Computation Algorithm

Chapter 4

Experiments and Results

4 Experiments and Results

Let the transmission rates (in milliseconds) be $\mu_1 = 0.1$, $\mu_2 = 100$, $\mu_3 = 10$, $\mu_4 = 100$ for packets of classes 1- 4 respectively [12] . The transmission rate for packets of classes 2 and 4 are the same. This assumption seems logical to justify why the transmission of packets of class 2 is not interrupted upon the arrival of a packet of class 4. The transmission rate of packets of class 3 μ_3 and packets of class 1 μ_1 are 10 times slower and 1000 times slower than packets of class 4 respectively. The system capacity N =10 packets. The values of arrivals rates (in milliseconds) are as follows: $\lambda_1 = 5$, $\lambda_2 = 10$, $\lambda_3 = 7$. The values of these parameters ensure that the system is heavily loaded.

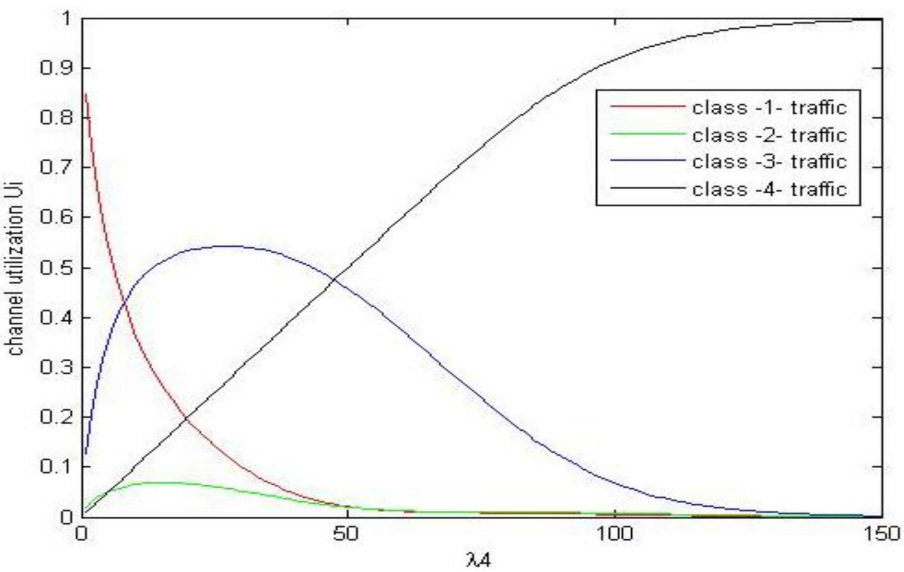


Figure 4-1: Utilization of the channel $U_{i,}$ i = 1, 2, 3, 4 versus λ_4 for N = 10

In Fig. 4.1, we show the utilization of the channel by packets of traffic classes 1, 2, 3 and 4 as a function of λ_4. When λ_4 is very small, the channel is primarily occupied by packets of class 1 due to their larger size and to the fact that traffic of class 2 and 3 have non-preemptive priority over packets of class 1. Increasing λ_4, the utilization of the channel by packets of class 1 sharply decreases as they get preempted by packets of class 4 while having the lowest priority of all classes. The utilization by packets of class 3 can be split into two parts. In the first part, increasing λ_4 the utilization of channel by packets of class 3 increases to a certain point as there are more chances to packets of class 3 to utilize the channel after the preemption of the larger packets of class 1. In the second part and while increasing λ_4 further, the utilization of the channel by packets of class 3 decreases as the channel is more occupied by packets of class 4. Similar behavior can be noticed for the utilization of channel by packets of class 2. The peak point for the curve of packets of class 2 is smaller than the curve of class 3 because $\mu_2 > \mu_3$. Increasing λ_4, the utilization of the channel by packets of class 4 increases almost linearly since they are not influenced by packets of classes 1 and 3 and there is a minimal influence of small packets of class 2. For $\lambda_4 > 150$, the buffer is fully occupied by packets of class 4.

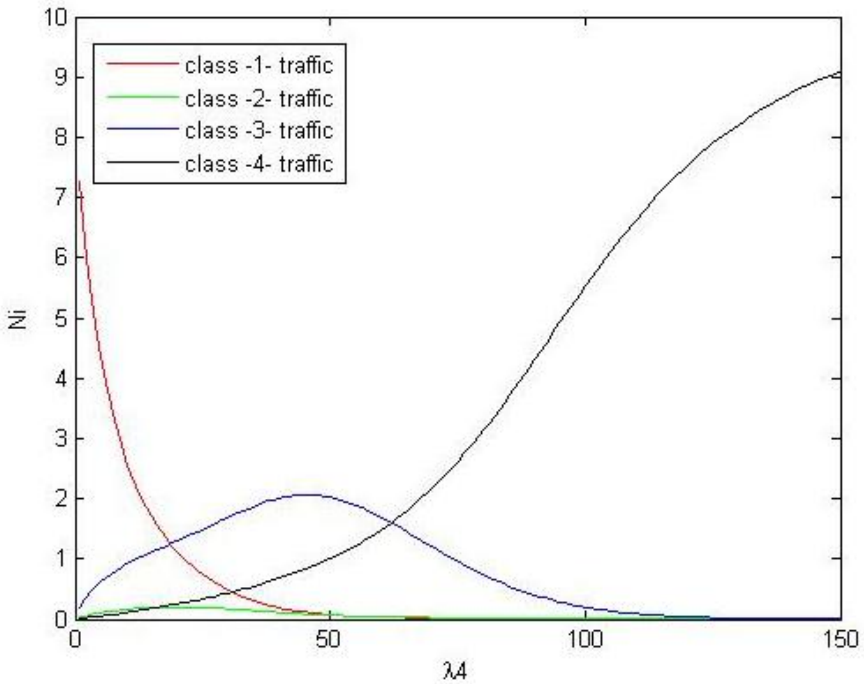


Figure 4-2: Average number of packets of traffic classes N_i, i = 1, 2, 3, 4 versus arrival rate λ_4

Fig. 4.2 demonstrate the effect of increasing λ_4 on an average number of packets of classes 1, 2, 3 and 4 for the same parameter set of Fig. 4.2. As expected the behavior of N_i, i = 1,2,3,4 in Fig. 4.1 is similar to the corresponding utilization U_i of the channel in Fig. 4.2. For small values of λ_4, we notice that N_1 is highest due to the fact that these packets have the lowest priority and the channel utilization is high. Increasing λ_4 further, N_1 sharply decreases as these packets are pushed out first. This creates more space for a packet of class 2 and 3 to join the buffer, thus the average number of packets of class 2 (N_2) and class 3 (N_3) increase. Further increase in λ_4 pushes out packets of classes 2 and 3 and the system is primarily occupied by packets of class 4. Also, we notice that the peak point for class 2 comes earlier than the peak point for class 3 as packets of class 2 are pushed out before packets of class 3. For λ_4 >150, the system is almost completely occupied by packets of class 4.

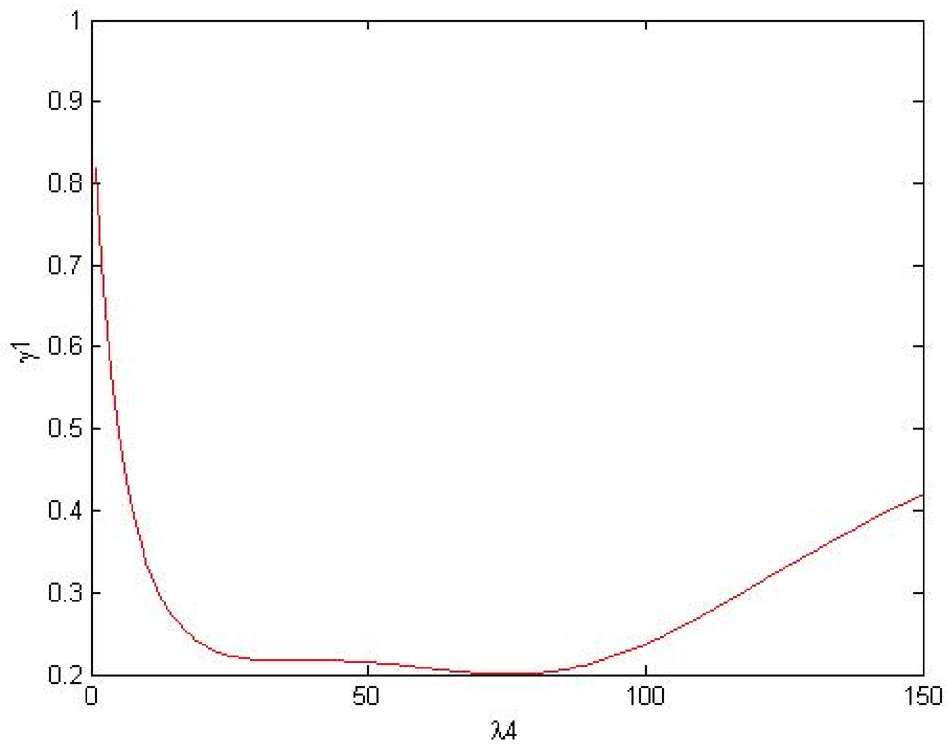


Figure 4-3: Effect of increasing λ_4 on Blocking probability γ_1 for N = 10

Fig. 4.3 shows the blocking probability of packets of class 1 as a function of λ_4. For this parameter set and when λ_4 is very small, the buffer is highly occupied and the blocking probability is high too. Increasing λ_4 pushes out packets of class 1 first that occupy the channel longer time. The channel becomes busy with packets of the higher transmission rate and as a result, there is more space in the buffer. This explains why blocking probability decreases. Increasing λ_4 further, the blocking probability starts to increase because the buffer is more occupied by packets of class 4. The blocking probability for packets of classes 2 and 3 is the same for packets of class 1 above.

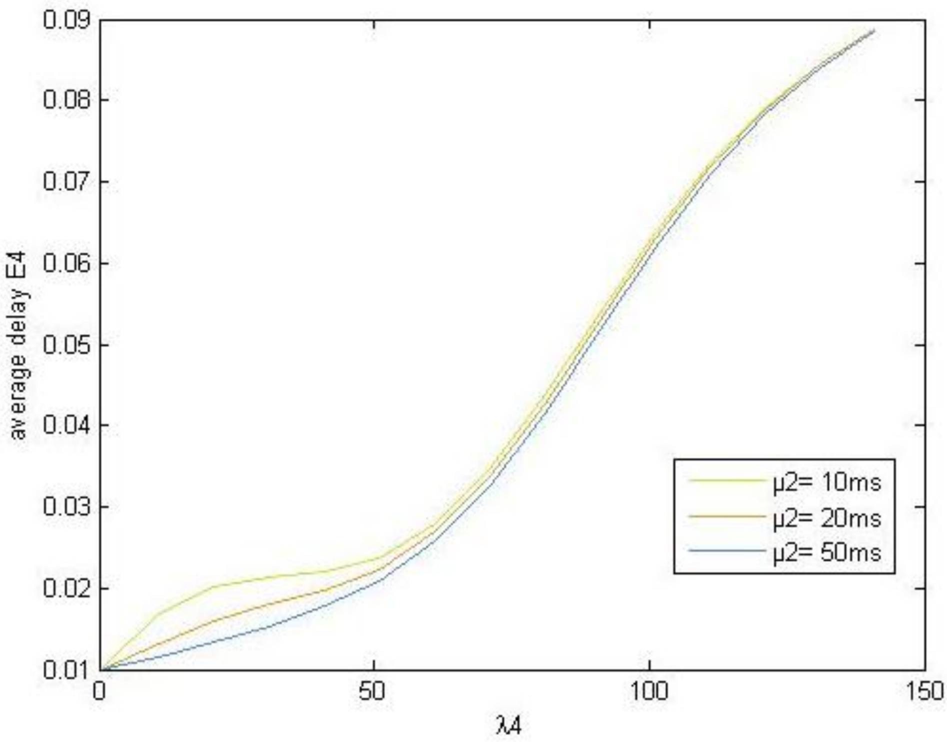


Figure 4-4: Average Delay of class 4 packets versus λ_4 for various values of μ_2 and N = 10

Fig. 4.4 shows the average delay of packets of class 4 as a function of λ_4 for different values of service rate μ_2. It can be seen that increasing λ_4 increases the number of packets of class 4 and the delay increases. For small values of λ_4 and μ_2 = 10, the delay of packets of class 4 increases sharply because of larger utilization of the channel by packets of class

2. Increasing λ_4 further, this effect decreases since the priority of packets of class 4 are higher than class 2 and packets of class 2 have less chance to occupy the channel. Increasing μ_2, and small values of λ_4, we notice that the delay of packets of class 4 decreases as packets of class 2 leave the channel faster. For values of λ_4>50, μ_2 have a small effect on the delay of packets of class 4.

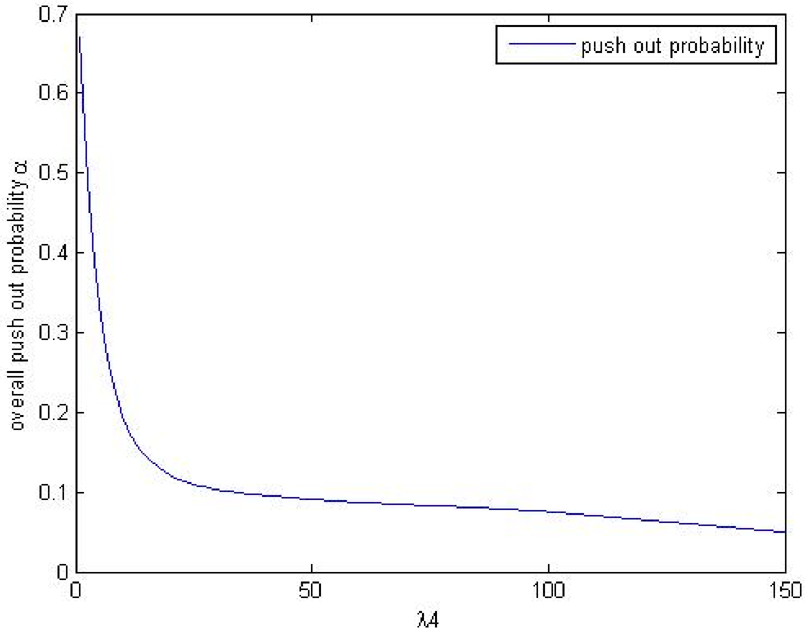


Figure 4-5: Overall Push out probability α versus λ_4 for N = 10

Fig. 4.5 shows the overall push out probability α of packets of classes 1, 2, 3 as a function of λ_4. The overall push out probability starts high due to the fact the buffer is highly occupied with packets of class 1 and these packets are pushed out first. Increasing λ_4, the overall push out probability falls sharply as the overall number of packets of classes 1, 2 and 3 decreases. For λ_4 >30, α decreases slowly. This is explained by the fact that packets of classes 2 and 3 are served more quickly than packets of class 1 decreasing the buffer length. Also, for high values of λ_4, the system becomes occupied primarily with packets of class 4 and there is less number of classes 1, 2 and 3 to push out.

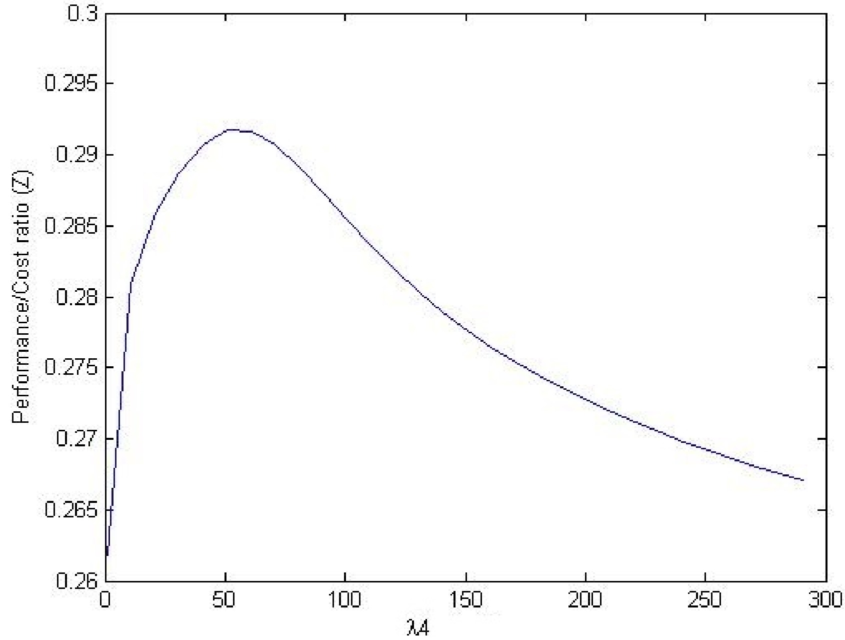


Figure 4-6: Performance/Cost Proportion (Z) versus λ_4 for $\beta = 10$

Fig. 4.6 shows the performance/cost Proportion (Z) as a function of λ_4. This Figure presents the case when the cost function is minimum i.e. the utilization of the channel is close to 1. It can be seen that the value of the function Z can be maximized for a certain value of λ_4. This figure can be split into two parts. In the first part, increasing λ_4, the value of Z increases. This happens because in this case both the blocking probability and the push out the probability for the first three classes of traffic decrease. In the second part, increasing λ_4, the value of Z decreases. In this case and while the value of the push out the probability for the first three classes decreases further, the value of blocking probability for the first three classes starts to increase as the buffer becomes more occupied by class 4 packets. Also and throughout both parts, it is evident that increasing λ_4 increase the blocking probability for packets of class 4. This behavior seems to persist for smaller arrival rates of λ_1, λ_2, and λ_3 and for other values of β between 1 and 10.

Chapter 5

Conclusion and Future Works

Conclusion and Future Works

We have developed an analytical model to evaluate a real application present in IoT networks. Detailed analysis is conducted and several performance measurements have been investigated. We have assessed the impact of the arrival rate of packets of class 4 on the performance measures. The analysis clearly shows that the overall performance of the system Z can be maximized for certain values of λ_4. The suggested model can be used to assess the performance measures of intelligent devices to meet numerous QoS constraints under different input parameterization.

Our future work is to use the performance measures to design an adaptive buffer with a dynamic threshold to allow other types of packets to go through the buffer before it will be full of type 4 packets based on the output of our performance measures. So when the buffer level is below the threshold, the adaptive buffer accepts both high priority and low priority packets and when the buffer level is over a predetermined threshold, low priority packets cannot access the buffer and are discarded.

Bibliography

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of Things for Smart Cities," *the final version of this manuscript will appear in the IEEE Internet Of Things Journal,* vol. 1, no. 1, pp. 22--32, 2014.

[2] J. Gubbi, . R. Buyya, S. Marusic and M. Palaniswami, "Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions," *Future generation computer systems,* vol. 29, no. 7, pp. 1645--1660, 2013.

[3] E. Alsaadi and A. Tubaishat, "Internet of Things: Features, Challenges, and Vulnerabilities," *International Journal of Advanced Computer Science and Information Technology (IJACSIT),* vol. 4, no. 1, pp. 1--13, 2015.

[4] H.-D. Ma, "Internet of things: Objectives and scientific challenges," *Journal of computer science and Technology,* vol. 26, no. 6, pp. 919-924, 2011.

[5] P. Pande and A. R. Padwalkar, "Internet of Things –A Future of Internet: A Survey," *International Journal of Advanced Research in Computer Science and Management Studies,* vol. 2, no. 2, 2014.

[6] L. Rosenbaum, M. Agrawal, L. Birch, and Y. Kureh, "Calculating Call Blocking, Preemption Probabilities and Bandwidth Utilization for Satellite Communication Systems," 2011.

[7] B. Costa, P. F. Pires, F. C. Delicato, W. Li and A. Y. Zomay, "Design and Analysis of IoT Applications: A Model-Driven Approach," in *IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing*, Auckland, New Zealand, 2016.

[8] S.. P. Eswaran and J. Bapat, "Service Centric Markov based Spectrum Sharing for Internet of Things (IoT)," *IEEE Region 10 Symposium,* pp. 9--12, 2015.

[9] I. Awan, M. Younas and W. Naveed, "Modelling QoS in IoT Applications," in *International Conference on Network-Based Information Systems*, 2014.

[10] S. Kausha and R. Sharma, "Modeling and Analysis of Adaptive Buffer Sharing Scheme for Consecutive Packet Loss Reduction in Broadband Networks," *International Journal of Electrical and Electronics Engineering,* 2010.

[11] R. C. Bhaddurgatte, V. K. BP and S., "A Review: QoS Architecture and Implementations in IoT Environment," *Research & Reviews: Journal of Engineering and Technology,* pp. 6--12, 2015.

[12] S. P. Eswaran, A. V and J. Bapat, "Event Driven Opportunistic Communication Enabler for Smart City," in *14 Eighth International Conference on Next Generation Mobile Applications, Services, and Technology,* 2014.

[13] S. Sankaran, "Modeling the performance of IoT networks," in *IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS),*

Bangalore, India, 2016.

[14] H. O. Al-Sakran, "Intelligent Traffic Information System Based on Integration of Internet of Things and Agent Technology," *International Journal of Advanced Computer Science and Applications (IJACSA)* , vol. 6, no. 2, pp. 37--43, 2015.

[15] S. Madakam, R. Ramaswamy and S. Tripathi, "Internet of Things (IoT): A Literature Review," *Journal of Computer and Communications,* vol. 3, no. 05, pp. 164--173 , 2015.

[16] D. O. Vemesan and D. P. Friess, Internet of Things - From Research and Innovative to Market Deployment, River publishers Aalborg, 2014.

[17] L. D. Xu, W. He and S. Li, "Internet of Things in Industries: A Survey," *IEEE Transactions on Industrial Informatics,* vol. 10, no. 4, pp. 2233--2243, 2014.

[18] L. Xiao and Z. Wang, "Internet of Things: a New Application for Intelligent Traffic Monitoring System," *JOURNAL OF NETWORKS,* vol. 6, no. 6, p. 887, 2011.

[19] J. Jin, J. Gubbi, T. Luo and M. Palaniswami, "Network Architecture and QoS Issues in the Internet of Things for a Smart City," in *International Symposium on Communications and Information Technologies (ISCIT)*, 2012.

[20] "Markov model," [Online]. Available: https://en.wikipedia.org/wiki/Markov_model. [Accessed 5 11 2018].

[21] A. TOLVER, An Introduction to Markov Chains, Lecture Notes For Stochastic Processes, 2016.

[22] R. Sharma, N. Kumar, N. B. Gowda and T. Srinivas, "Probabilistic Prediction based Scheduling for Delay Sensitive Traffic in Internet of Things," in *The 6th International Conference on Ambient Systems, Networks and Technologies (ANT 2015), Procedia Computer Science*, 2015.

[23] Y. Sun, F. Tong, Z. Zhang and S. He, "Throughput modeling and Analysis of Random Access in Narrow -band Interner of Things," *IEEE Internet of Things Journal,* vol. 5, no. 3, pp. 1485--1493, 2018.

[24] M. Kiran and P. Rajalakshmi, "Performance Analysis of CSMA/CA and PCA Time critical Industrial IoT Applications," *IEEE Transactions on Industrial Informatics,* vol. 14, no. 5, pp. 2281--2293, 2018.

[25] W.-X. Li, J. Xu, and H. Jiang, "Queuing States Analysis on a Hybrid Scheduling Strategies for Heterogeneous Traffics in IOT," in *International Conference on Computer Science and Service Systems*, Nanjing, China, 2012.

[26] R. Sharma, N. Kumar, N. B. Gowda, and T. Srinivas, "Packet Scheduling scheme yo guarantee QoS in Internet of Things," *Wireless Personal Communications,* vol. 100, no. 2, pp. 557--569, 2018.

[27] P. S. Tomar, "Finite Markovian Queue With Service Priorities," *Indian Journal of Mathematics and Mathematical Sciences,* vol. 7, no. 2, pp. 135--143, 2011.

[28] K. E. Avrachenkov, N. O. Vilchevsky and G. L. Shevlyakov, "Priority Queueing with Finite Buffer Size and Randomized Push-Out Mechanism," *ACM Sigmetrics Performance Evaluation Review,* vol. 31, no. 1, pp. 324--325, 2003.

[29] V. Muliukha, A. Ilyshenko, O. Zayats, and V. Zaborovsky, "Preemptive queueing system with randomized push-out mechanism," *Communications in Nonlinear Science and Numerical Simulation,* vol. 21, no. 1–3, pp. 147--158, 2015.

[30] B. R. Haverkort, Markovian models for performance and dependability evaluation, Springer, 2000.

[31] G. White, V. Nallur and S. Clarke, "Quality of service approaches in IoT: A systematic mapping," *Journal of Systems and Software,* vol. 132, pp. 186--203, 2017.

[32] M. Kim, "A quality model for evaluating iot applications," *International Journal of Computer and Electrical Engineering (IACSIT Press),* vol. 8, no. 1, p. 66, 2016.

[33] D. Macedo, L. Affonso and I. Silva, "A dependability evaluation for Internet of Things incorporating redundancy aspects," in *Networking, Sensing and Control (ICNSC), 2014 IEEE 11th International Conference on*, 417--422.

[34] T. Arjannikov, S. Diemert, S. Ganti, C. Lampman and E. C. Wiebe, "Using Markov Chains to Model Sensor Network Reliability," in *Proceedings of the 12th International Conference on Availability, Reliability and Security (ACM)*, 2017.

[35] M. Soliman, T. Abiodun, T. Hamouda, J. Zhou and C.-H. Lung, "Smart home: Integrating internet of things with web services and cloud computing," in *IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom),* 2013.

[36] I. Candan and M. Salameh, "Analytical modeling of a time-threshold based bandwidth allocation scheme for cellular networks," *Computer Communication,* vol. 30, no. 5, pp. 1036--1043, 2007.

Appendix

This part includes the result of all the functions that performed on IoT experiment using MATLAB.

Appendix A

The general model algorithms are illustrated by **Algorithm 1, 2, 3, 4 and 5** in detail.

Algorithm 1: The general procedure that was used to construct the state probabilities (PS)

Input: Buffer capacity N, arrival rate λ_i for class i traffic where i=1, 2, 3, 4 and class 4 has the highest priority, service rate μ_i for class i traffic where i=1, 2, 3, 4.

Output: PS steady-state probabilities, state space S

Data Preprocessing:

```
        function [ps,s] =constructPS(N, lambda1,lambda2,lambda3,lambda4,mu1,mu2,mu3,mu4)
%size is the dimension of the Q matrix.
%s is the state matrix

size ← 0
/* This nested loop to calculate the number of states*/

        for n1= 0 to N do
          for n2 = 0 to N do
            for n3 = 0 to N do
              for n4 = 0 to N do
if n1+ n2+ n3+n4 <=N then
                        /* Check all cases where the buffer will be filled of n1, n2, n3, n4 such that c=1 or 2
                           or 3 or 4*/
size ← size + 1

end if
        end for
              end for
                    end for
                          end for

s ← zeros(size, 5)
m ← 1

/* This nested loop to fill the states n1n the matrix s*/

        for n1= 0 to N do
          for n2 = 0 to N do
            for n3 = 0 to N do
              for n4 = 0 to N do

                        /* Check all cases where the buffer will be filled of n1, n2, n3, n4 such that c=1 or 2
                           or 3 or 4 and the states will be filled with n1, n2, n3, n4 */
```

$\text{size} \leftarrow \text{size} + 1$

/ An arriving packet from any class will get immediately transmitted when the system empty (case 1). */*

if c=0 **and** $n_1+ n_2+ n_3+n_4= 0$ **then**
$s(m,:) \leftarrow (\ldots,n_i + 1,\ldots,c = i)\ \ i = 1,2,3,4$
$m \leftarrow m + 1$
end if

/ In case 2, an arriving packet of classes 1,2,3 will join the buffer if the system is not empty and space is available in the buffer. An arriving packet of classes 1, 2 or 3 is lost if the system is full. */*

if $c\neq 0$ **and** $n_1+ n_2+ n_3+n_4 <=N$ **then**
$s(m,:) \leftarrow (\ldots,n_i + 1,\ldots,c)\ \ c \neq 0, i = 1,2,3$
$m \leftarrow m + 1$
end if

/ An arriving packet of class 4 will join the buffer if the packet in the transmission belongs to either class 2 or class 4 and the system is not full (case 3). */*

if c= 2,4 **and** $n_1+ n_2+ n_3+n_4 <=N$ **then**
$s(m,:) \leftarrow (n1, n2, n3, n4 + 1, c)\ \ \ \ c = 2,4$
$m \leftarrow m + 1$
end if

/ In case 4, an arriving packet of class 4 preempts the packet in transmission if the packet belongs to class 1 or 3. In this case, the interrupted packet rejoins the buffer if a space in the buffer is available. */*

if c= 1,3 **and** $n_1+ n_2+ n_3+n_4 <=N$ **then**
$s(m,) \leftarrow (n1, n2, n3, n4 + 1, c=4)$
$m \leftarrow m + 1$
end if

/ In cases 5 and 6, an arriving packet of class 4 pushes out a packet of lower priority when the system is full. The pushed out packet is lost. */*
/ The packet of the lowest priority is pushed out of the buffer (case 5). In this case, the arriving packet preempts the packet in transmission if the packet in transmission belongs to class 1 or class 3. The system is blocked if the buffer is fully occupied by packets of class 4. */*

if c= 1,3 **and** $n_1+ n_2+ n_3+n_4 =N$ **then**
$s(m,:) \leftarrow (\ldots,n_i - 1,\ldots,n_4 + 1,c = 4)$
$m \leftarrow m + 1$
end if

/ When a packet of class 2 is in transmission and the buffer is full, an arriving a packet of class 4 joins the buffer and pushes out a packet of the lowest priority waiting in the buffer (case 6). */*

if c= 2 **and** $n_1+ n_2+ n_3+n_4 =N$ **then**
$s(m,:) \leftarrow (\ldots,n_i - 1,\ldots,n_4 + 1,c = 2)$
$m \leftarrow m + 1$
end if

/ When a packet finishes transmission, the packet of the highest priority waiting for the*
the buffer will start transmission next (case 7) and if there are no more packets in
*the buffer, c is set to 0. */*

if $n_1+ n_2+ n_3+n_4 =0$ **then**
$s(m,:) \leftarrow (\ldots,n_i - 1,\ldots, c = j)$
$m \leftarrow m + 1$
end if

end for
end for
end for
end for

/ filling the Q matrix*/*

$Q \leftarrow \text{zeros}(\text{size}, \text{size})$
$n \leftarrow 1$

for i = 1 **to** size **do**
$n1 \leftarrow s(i,1)$
$n2 \leftarrow s(i,2)$
$n3 \leftarrow s(i,3)$
$n4 \leftarrow s(i,4)$
$c \leftarrow s(i,5)$
for j = 1 **to** size **do**

/ An arriving packet from any class will get immediately transmitted when the system is empty (case 1). */*

if $c=0$ **and** $n_1+ n_2+ n_3+n_4 = 0$ **then**
if from state is $(\ldots,n_i,\ldots,c = 0)$ and to state is $(\ldots,n_i + 1,\ldots,c = i)$ **then**
$Q(i,j) \leftarrow \lambda i$
end if
end if

/ In case 2, an arriving packet of classes 1,2,3 will join the buffer if the system is not empty and space is available in the buffer. An arriving packet of classes 1, 2 or 3 is lost if the system is full. */*

if $c\neq 0$ **and** $n_1+ n_2+ n_3+n_4 <= N$ **then**
if from state is $(\ldots,n_i,\ldots,c)$ and to state is $(\ldots,n_i + 1,\ldots,c)$ **then**
$Q(i,j) \leftarrow \lambda i$
end if
end if

/ An arriving packet of class 4 will join the buffer if the packet in the transmission belongs to either class 2 or class 4 and the system is not full (case 3). */*

if $c= 2,4$ **and** $n_1+ n_2+ n_3+n_4 <= N$ **then**
if from state is (n_1,n_2,n_3,n_4,c) and to state is $(n_1,n_2,n_3,n_4 + 1,c)$ **then**
$Q(i,j) \leftarrow \lambda 4$

end if
end if

/ In case 4, an arriving packet of class 4 preempts the packet in transmission if the packet belongs to class 1 or 3. In this case, the interrupted packet rejoins the buffer if a space in the buffer is available. */*

if c= 1,3 **and** $n_1+n_2+n_3+n_4$ <=N **then**
if from state is (n_1,n_2,n_3,n_4,c=1,3) and to state is (n_1,n_2,n_3,n_4+1,c=4) **then**
$Q(i,j) \leftarrow \lambda 4$
end if
end if

/ In cases 5 and 6, an arriving packet of class 4 pushes out a packet of lower priority when the system is full. The pushed out packet is lost. */*
/ The packet of the lowest priority is pushed out of the buffer (case 5). In this case, the arriving packet preempts the packet in transmission if the packet in transmission belongs to class 1 or class 3. The system is blocked if the buffer is fully occupied by packets of class 4. */*

if c= 1,3 **and** $n_1+n_2+n_3+n_4$ =N **then**
if from state is (…,n_i,…,c = 1,3,4) and to state is (…,n_i-1,…,n_4+1,c = 4) **then**
$Q(i,j) \leftarrow \lambda 4$
end if
end if

/ When a packet of class 2 is in transmission and the buffer is full, an arriving a packet of class 4 joins the buffer and pushes out a packet of the lowest priority waiting in the buffer (case 6). */*

if c= 2 **and** $n_1+n_2+n_3+n_4$ =N **then**
if from state is (…,n_i,…,c = 2) and to state is (…,n_i-1,…,n_4+1,c = 2) **then**
$Q(i,j) \leftarrow \lambda 4$
end if
end if

/ When a packet finishes transmission, the packet of the highest priority waiting for the the buffer will start transmission next (case 7) and if there are no more packets in the buffer, c is set to 0. */*

if $n_1+n_2+n_3+n_4$ =0 **then**
if from state is (…,n_i,…,c = i) and to state is (…,n_i-1,…,c = j) **then**
$Q(i,j) \leftarrow \mu i$
end if
end if
end for
end for

$qq \leftarrow zeros(size,2)$
for i = 1 **to** size **do**
$qq(i,1) \leftarrow sum(Q(i,:))$
$Q(i,i) \leftarrow -sum(Q(i,:))$
$qq(i,2) \leftarrow Q(i,i)$

end for

ps ← ones(1, size) ∗ ((Q + ones(size, size))^(−1))

Algorithm 2: The general procedure that was used to compute blocking probability

Input: Buffer capacity N, arrival rate λ_i for class i traffic where i=1, 2, 3, 4 and class 4 has the highest priority, service rate μ_i for class i traffic where i=1, 2, 3, 4 which are needed to execute constructPs function.
Output: Blocking probability and utilization and the average number of packets

Data Preprocessing:

for k = 1 **to lambdas2 do**
for n1 = 0 **to lambdas4 do**

num1 ← 0
num2 ← 0
num3 ← 0
num4 ← 0
full123 ← 0
full4 ← 0
util1 ← 0
util2 ← 0
util3 ← 0
util4 ← 0

[ps, s] ← constructPS(N, lambda1, lambda2(k), lambda3, lambda4(j), mu1, mu2, mu3, mu4)
[x, y] ← size(ps)

for i = 2 **to y do**

if s(i,1)+s(i,2)+s(i,3)+s(i,4) ==N **then**
full123 ← full123 + ps(1, i)
end if
if s(i,4)==N || s(i,4)==N-1 && s(i,5)==2 **then**
full4 ← full4 + ps(1, i)
end if

c ← s(i, 5)

if c==1 **then**
num1 ← num1 + ps(1, i) ∗ s(i, 1)
util1 ← util1 + ps(1, i)
end if
if c==2 **then**
num2 ← num2 + ps(1, i) ∗ s(i, 2)
util2 ← util2 + ps(1, i)
end if
if c==3 **then**
num3 ← num3 + ps(1, i) ∗ s(i, 3)
num3 ← util3 + ps(1, i)
end if
if c==4 **then**

```
num4 ← num4 + ps(1, i) ∗ s(i, 4)
util4 ← util4 + ps(1, i)
end if

end for

util ← 0

end for
end for
```

Algorithm 3: The general procedure that was used to compute the average delay

---

Input: Buffer capacity N, arrival rate λ_i for class i traffic where i=1, 2, 3, 4 and class 4 has the highest priority, service rate μ_i for class i traffic where i=1, 2, 3, 4 which are needed to execute constructPs function.

Output: Average delay

Data Preprocessing:

```
g ← zeros(lambdas2, lambdas4)
g1 ← zeros(lambdas2, lambdas4)
g2 ← zeros(lambdas2, lambdas4)
g3 ← zeros(lambdas2, lambdas4)
g4 ← zeros(lambdas2, lambdas4)


for k = 1 to lambdas2 do
for n1 = 0 to lambdas4 do

num1 ← 0
num2 ← 0
num3 ← 0
num4 ← 0
full123 ← 0
full4 ← 0

[ps, s] ← constructPS(N, lambda1, lambda2(k), lambda3, lambda4(j), mu1, mu2, mu3, mu4)
[x, y] ← size(ps)


        for i = 2 to y do

                if s(i,1)+s(i,2)+s(i,3)+s(i,4) ==N then
                full123 ← full123 + ps(1, i)
                end if
                if s(i,4)==N || s(i,4)==N-1 && s(i,5)==2 then
                full4 ← full4 + ps(1, i)
                end if
                c ← s(i, 5)

                if c==1 then
                num1 ← num1 + ps(1, i) ∗ s(i, 1)
                end if
                if c==2 then
```

$\mathrm{num2} \leftarrow \mathrm{num2} + \mathrm{ps}(1, \mathrm{i}) * \mathrm{s}(\mathrm{i}, 2)$
end if
if c==3 **then**
$\mathrm{num3} \leftarrow \mathrm{num3} + \mathrm{ps}(1, \mathrm{i}) * \mathrm{s}(\mathrm{i}, 3)$
end if
if c==4 **then**
$\mathrm{num4} \leftarrow \mathrm{num4} + \mathrm{ps}(1, \mathrm{i}) * \mathrm{s}(\mathrm{i}, 4)$
end if
$\mathrm{rho4} \leftarrow (\mathrm{lambda4}(\mathrm{j})/\mathrm{mu4})$

end for

$\mathrm{util} \leftarrow 0$
$\mathrm{g1}(\mathrm{k}, \mathrm{j}) \leftarrow \mathrm{num1}/(\mathrm{lambda1} * (1 - \mathrm{full123}))$
$\mathrm{g2}(\mathrm{k}, \mathrm{j}) \leftarrow \mathrm{num2}/(\mathrm{lambda2}(\mathrm{k}) * (1 - \mathrm{full123}))$
$\mathrm{g3}(\mathrm{k}, \mathrm{j}) \leftarrow \mathrm{num3}/(\mathrm{lambda3} * (1 - \mathrm{full123}))$
$\mathrm{g}(\mathrm{k}, \mathrm{j}) \leftarrow \mathrm{num4}/(\mathrm{lambda4}(\mathrm{j}) * (1 - \mathrm{full4}))$

end for
end for

Algorithm 4: The general procedure that was used to compute push out the probability

---

Input: Buffer capacity N, arrival rate λ_i for class i traffic where i=1, 2, 3, 4 and class 4 has the highest priority, service rate μ_i for class i traffic where i=1, 2, 3, 4 which are needed to execute constructPs function.
Output: Overall push out the probability

Data Preprocessing:
$\mathrm{g} \leftarrow \mathrm{zeros}(\mathrm{lambdas2}, \mathrm{lambdas4})$
$\mathrm{g1} \leftarrow \mathrm{zeros}(\mathrm{lambdas2}, \mathrm{lambdas4})$
$\mathrm{g2} \leftarrow \mathrm{zeros}(\mathrm{lambdas2}, \mathrm{lambdas4})$
$\mathrm{g3} \leftarrow \mathrm{zeros}(\mathrm{lambdas2}, \mathrm{lambdas4})$
$\mathrm{g4} \leftarrow \mathrm{zeros}(\mathrm{lambdas2}, \mathrm{lambdas4})$

for k = 1 **to lambdas2 do**
for n1 = 0 **to lambdas4 do**

$\mathrm{alpha} \leftarrow 0$
$\mathrm{alpha1} \leftarrow 0$
$\mathrm{alpha2} \leftarrow 0$
$\mathrm{alpha3} \leftarrow 0$

$[\mathrm{ps}, \mathrm{s}] \leftarrow \mathrm{constructPS}(\mathrm{N}, \mathrm{lambda1}, \mathrm{lambda2}(\mathrm{k}), \mathrm{lambda3}, \mathrm{lambda4}(\mathrm{j}), \mathrm{mu1}, \mathrm{mu2}, \mathrm{mu3}, \mathrm{mu4})$
$[\mathrm{x}, \mathrm{y}] \leftarrow \mathrm{size}(\mathrm{ps})$

for i = 2 **to y do**

```
sum ← s(i,1) + s(i,2) + s(i,3) + s(i,4)

if (sum ==N) then

    if (s(i,1)==0 && s(i,2)==1 && s(i,3)==0&&s(i,4)==N-1 && s(i,5)==2) || (s(i,1)==0 &&
        s(i,2)==0 && s(i,3)==0&&s(i,4)==N && s(i,5)==4) then
    continue
    end if

    c ← s(i,5)

    if c==1 then
    mc ← mu1
    elseif c==2 then
    mc ← mu2
    elseif c==3 then
    mc ← mu3
    elseif c==4 then
    mc ← mu4
    end if
    alpha ← alpha + ps(1,i) ∗ lambda4(j)/(lambda4(j) + mc)

    if s(i,1)>0 then
    alpha1 ← alpha1 + ps(1,i) ∗ lambda4(j)/(lambda4(j) + mc)
    elseif s(i,2)>0 then
    alpha2 ← alpha2 + ps(1,i) ∗ lambda4(j)/(lambda4(j) + mc)
    elseif s(i,3)>0 then
    alpha3 ← alpha3 + ps(1,i) ∗ lambda4(j)/(lambda4(j) + mc)
    end if

end if

    end for

g(k,j) ← alpha
g1(k,j) ← alpha1
g2(k,j) ← alpha2
g3(k,j) ← alpha3

    end for
        end for
```

Algorithm 5: The general procedure that was used to compute overall system performance as a function of Z

Input: Buffer capacity N, arrival rate λ_i for class i traffic where i=1, 2, 3, 4 and class 4 has the highest priority, service rate μ_i for class i traffic where i=1, 2, 3, 4 which are needed to execute constructPs function, penalty weight β .

Output: Overall system performance as a function of Z

Data Preprocessing:

beta ← 10

```
g ← zeros(lambdas2, lambdas4)
g1 ← zeros(lambdas2, lambdas4)
g2 ← zeros(lambdas2, lambdas4)
g3 ← zeros(lambdas2, lambdas4)
g4 ← zeros(lambdas2, lambdas4)


b1 ← zeros(lambdas2, lambdas4)
b2 ← zeros(lambdas2, lambdas4)

gos2 ← zeros(lambdas2, lambdas4)
util ← gos2
zz ← zeros(lambdas2, lambdas4)

for k = 1 to lambdas2 do
for n1 = 0 to lambdas4 do

full123 ← 0
full4 ← 0
alpha ← 0
alpha1 ← 0
alpha2 ← 0
alpha3 ← 0
rho1 ← 0
rho2 ← 0
rho3 ← 0
rho4 ← 0

[ps, s] ← constructPS(N, lambda1, lambda2(k), lambda3, lambda4(j), mu1, mu2, mu3, mu4)
[x, y] ← size(ps)

        for i = 2 to y do

                sum ← s(i, 1) + s(i, 2) + s(i, 3) + s(i, 4)
                c ← s(i, 5)


                if c==1 then
                mc ← mu1
                rho1 ← rho1 + ps(1, i)
                elseif c==2 then
                mc ← mu2
                rho2 ← rho2 + ps(1, i)
                elseif c==3 then
                mc ← mu3
                rho3 ← rho3 + ps(1, i)
                elseif c==4 then
                mc ← mu4
                rho4 ← rho4 + ps(1, i)
                end if

                if s(i,1)+s(i,2)+s(i,3)+s(i,4) ==N then
                full123 ← full123 + ps(1, i)
                end if
```

```
                    if s(i,4)==N || s(i,4)==N-1 && s(i,5)==2 then
                    full4  ← full4 + ps(1, i)
                    end if


                    if (s(i,1)==0 && s(i,2)==1 && s(i,3)==0&&s(i,4)==N-1 && s(i,5)==2) || (s(i,1)==0 &&
                       s(i,2)==0 && s(i,3)==0&&s(i,4)==N && s(i,5)==4) then
                    continue
                    end if


if s(i,1)+s(i,2)+s(i,3)+s(i,4) ==N then
                    alpha  ← alpha  +  ps(1, i) ∗ lambda4(j)/(lambda4(j) + mc)
elseif s(i,1)>0 then
                    alpha1 ← alpha1 +  ps(1, i) ∗ lambda4(j)/(lambda4(j) + mc)
                    elseif s(i,2)>0 then
                    alpha2 ← alpha2 +  ps(1, i) ∗ lambda4(j)/(lambda4(j) + mc)
                    elseif s(i,3)>0    then
                    alpha3 ← alpha3 +  ps(1, i) ∗ lambda4(j)/(lambda4(j) + mc)

 end if
end if


           end for


         b1(k, j)  ← full123
         b2(k, j) ← full4


g(k, j)  ← alpha
g1(k, j)  ← alpha1
g2(k, j) ← alpha2
g3(k, j) ←  alpha3

GOS1   ← full123  +  beta ∗ (alpha1 + alpha2 + alpha3) + full4
gos2(k, j) ← GOS1
U ← rho1 + rho2 + rho3 + rho4
util(k, j) ← U
Z ← U/GOS1
zz(k, j) ← Z

end for
end for
```

Appendix B

Table A. 1: Example of state space where N = 5

n1	n2	n3	n4	c
0	0	0	0	0
0	0	0	1	4
0	0	0	2	4
0	0	0	3	4
0	0	0	4	4
0	0	0	5	4
0	0	1	0	3
0	0	1	1	4
0	0	1	2	4
0	0	1	3	4
0	0	1	4	4
0	0	2	0	3
0	0	2	1	4
0	0	2	2	4
0	0	2	3	4
0	0	3	0	3
0	0	3	1	4
0	0	3	2	4
0	0	4	0	3
0	0	4	1	4
0	0	5	0	3
0	1	0	0	2
0	1	0	1	2
0	1	0	1	4
0	1	0	2	2
0	1	0	2	4
0	1	0	3	2
0	1	0	3	4
0	1	0	4	2
0	1	0	4	4
0	1	1	0	2
0	1	1	0	3
0	1	1	1	2
0	1	1	1	4
0	1	1	2	2
0	1	1	2	4
0	1	1	3	2
0	1	1	3	4
0	1	2	0	2

0	1	2	0	3
0	1	2	1	2
0	1	2	1	4
0	1	2	2	2
0	1	2	2	4
0	1	3	0	2
0	1	3	0	3
0	1	3	1	2
0	1	3	1	4
0	1	4	0	2
0	1	4	0	3
0	2	0	0	2
0	2	0	1	2
0	2	0	1	4
0	2	0	2	2
0	2	0	2	4
0	2	0	3	2
0	2	0	3	4
0	2	1	0	2
0	2	1	0	3
0	2	1	1	2
0	2	1	1	4
0	2	1	2	2
0	2	1	2	4
0	2	2	0	2
0	2	2	0	3
0	2	2	1	2
0	2	2	1	4
0	2	3	0	2
0	2	3	0	3
0	3	0	0	2
0	3	0	1	2
0	3	0	1	4
0	3	0	2	2
0	3	0	2	4
0	3	1	0	2
0	3	1	0	3
0	3	1	1	2
0	3	1	1	4
0	3	2	0	2
0	3	2	0	3
0	4	0	0	2
0	4	0	1	2
0	4	0	1	4

0	4	1	0	2
0	4	1	0	3
0	5	0	0	2
1	0	0	0	1
1	0	0	1	4
1	0	0	2	4
1	0	0	3	4
1	0	0	4	4
1	0	1	0	1
1	0	1	0	3
1	0	1	1	4
1	0	1	2	4
1	0	1	3	4
1	0	2	0	1
1	0	2	0	3
1	0	2	1	4
1	0	2	2	4
1	0	3	0	1
1	0	3	0	3
1	0	3	1	4
1	0	4	0	1
1	0	4	0	3
1	1	0	0	1
1	1	0	0	2
1	1	0	1	2
1	1	0	1	4
1	1	0	2	2
1	1	0	2	4
1	1	0	3	2
1	1	0	3	4
1	1	1	0	1
1	1	1	0	2
1	1	1	0	3
1	1	1	1	2
1	1	1	1	4
1	1	1	2	2
1	1	1	2	4
1	1	2	0	1
1	1	2	0	2
1	1	2	0	3
1	1	2	1	2
1	1	2	1	4
1	1	3	0	1
1	1	3	0	2

1	1	3	0	3
1	2	0	0	1
1	2	0	0	2
1	2	0	1	2
1	2	0	1	4
1	2	0	2	2
1	2	0	2	4
1	2	1	0	1
1	2	1	0	2
1	2	1	0	3
1	2	1	1	2
1	2	1	1	4
1	2	2	0	1
1	2	2	0	2
1	2	2	0	3
1	3	0	0	1
1	3	0	0	2
1	3	0	1	2
1	3	0	1	4
1	3	1	0	1
1	3	1	0	2
1	3	1	0	3
1	4	0	0	1
1	4	0	0	2
2	0	0	0	1
2	0	0	1	4
2	0	0	2	4
2	0	0	3	4
2	0	1	0	1
2	0	1	0	3
2	0	1	1	4
2	0	1	2	4
2	0	2	0	1
2	0	2	0	3
2	0	2	1	4
2	0	3	0	1
2	0	3	0	3
2	1	0	0	1
2	1	0	0	2
2	1	0	1	2
2	1	0	1	4
2	1	0	2	2
2	1	0	2	4
2	1	1	0	1

2	1	1	0	2
2	1	1	0	3
2	1	1	1	2
2	1	1	1	4
2	1	2	0	1
2	1	2	0	2
2	1	2	0	3
2	2	0	0	1
2	2	0	0	2
2	2	0	1	2
2	2	0	1	4
2	2	1	0	1
2	2	1	0	2
2	2	1	0	3
2	3	0	0	1
2	3	0	0	2
3	0	0	0	1
3	0	0	1	4
3	0	0	2	4
3	0	1	0	1
3	0	1	0	3
3	0	1	1	4
3	0	2	0	1
3	0	2	0	3
3	1	0	0	1
3	1	0	0	2
3	1	0	1	2
3	1	0	1	4
3	1	1	0	1
3	1	1	0	2
3	1	1	0	3
3	2	0	0	1
3	2	0	0	2
4	0	0	0	1
4	0	0	1	4
4	0	1	0	1
4	0	1	0	3
4	1	0	0	1
4	1	0	0	2
5	0	0	0	1

Table A. 2: The transition probability PS

0.0553	0.0147	0.0074	0.0069	0.0116	0.0197	0.0255	0.014	0.0106	0.0135	0.0205	0.0157
0.0126	0.0149	0.0221	0.0115	0.0125	0.0173	0.0068	0.0084	0.0018	0.022	0.0055	0.0051
0.0014	0.0044	3.44E-04	0.0048	0.004	0.0052	0.0041	0.0172	0.0021	0.0116	7.73E-04	0.0111
0.0027	0.0126	7.73E-04	0.0181	5.80E-04	0.0158	0.0015	0.0179	1.45E-04	0.0134	3.99E-04	0.0143
4.35E-05	0.0049	0.0111	0.0035	0.0024	0.001	0.0029	0.0021	0.0027	0.0026	0.012	0.0016
0.0091	0.0029	0.0092	5.84E-04	0.0126	0.0013	0.0122	2.04E-04	0.0059	0.0048	0.0016	0.0011
0.0021	0.0011	0.0012	0.0053	0.002	0.0046	4.82E-04	0.0038	0.0012	0.0012	1.94E-04	6.07E-04
9.70E-04	2.42E-04	0.0207	0.0083	0.0036	0.0024	0.0021	0.0041	0.0106	0.0083	0.0055	0.0045
8.28E-04	0.009	0.0073	0.0062	1.66E-04	0.0058	0.0055	5.52E-05	0.0022	0.0028	0.0121	0.0034
0.0041	9.26E-04	0.0033	7.87E-04	0.0024	0.0011	0.0025	0.0122	0.0014	0.0092	0.0014	0.0072
3.31E-04	5.21E-04	0.0116	8.38E-04	0.01	1.47E-04	1.71E-04	0.0054	3.68E-04	0.0077	0.0026	0.0021
0.002	0.0017	2.21E-04	0.0019	0.0077	0.0024	0.0061	1.47E-04	7.37E-04	0.0054	4.91E-05	0.0026
0.002	4.79E-04	6.54E-05	0.0013	0.0019	1.09E-05	6.36E-04	0.0085	0.0036	0.0018	0.001	0.002
0.0048	0.0042	0.0028	4.50E-04	0.0039	0.0035	1.68E-04	0.0016	0.0013	0.0051	0.0015	0.0022
9.56E-04	0.0015	5.99E-04	0.0011	0.0051	0.0012	0.0043	3.37E-04	3.88E-04	0.0031	2.00E-04	0.0025
0.0017	6.43E-04	2.24E-04	0.0012	0.0016	4.99E-05	7.53E-04	0.0026	0.0013	6.73E-04	6.52E-04	0.0016
0.0015	2.67E-04	7.75E-04	4.34E-04	0.0012	6.64E-04	5.92E-04	3.56E-04	4.63E-04	7.47E-04	1.19E-04	4.82E-04
4.49E-04	2.75E-04	2.22E-04	1.41E-04	1.48E-04	1.17E-04	4.99E-05	**SUM**				**1**

Table A. 3: The result of performance measures which fill the blocking probability for packets $\gamma_{1,2,3}$ of classes 1,2,3 and the blocking probability γ_4 for packets of class 4 and the utilization U

j	**num1**	**num2**	**num3**	**num4**	**γ1,2,3**	**γ4**	**U1**	**U2**	**U3**	**U4**
1	7.257263	0.033492	0.203956	0.010103	0.819244	1.75E-15	0.845395	0.018076	0.126529	0.01
2	6.2773	0.056579	0.341397	0.02042	0.703678	4.09E-15	0.74294	0.029632	0.207425	0.02
3	5.511235	0.075619	0.451067	0.030963	0.616755	6.77E-16	0.663388	0.038321	0.268271	0.03
4	4.88931	0.092104	0.542322	0.041742	0.549086	1.64E-15	0.599219	0.045074	0.31564	0.04
5	4.370247	0.106829	0.620672	0.052764	0.495053	2.00E-14	0.545935	0.050444	0.353463	0.05
6	3.928249	0.120193	0.689543	0.06404	0.451092	6.42E-14	0.500691	0.05477	0.384235	0.06
7	3.546346	0.132385	0.751192	0.075576	0.414825	3.34E-13	0.461596	0.058277	0.409622	0.07
8	3.212802	0.143483	0.807188	0.087382	0.384592	1.40E-12	0.42734	0.061117	0.430785	0.08
9	2.919106	0.153515	0.858684	0.099465	0.359186	4.87E-12	0.396981	0.063399	0.448567	0.09
10	2.658819	0.162497	0.906573	0.111836	0.337705	1.49E-11	0.369816	0.065204	0.463602	0.1
11	2.426898	0.170443	0.951569	0.124501	0.319453	4.08E-11	0.345306	0.066598	0.476374	0.11
12	2.219286	0.177372	0.994262	0.13747	0.303885	1.02E-10	0.323032	0.06763	0.487265	0.12
13	2.032658	0.183314	1.035148	0.150753	0.290565	2.38E-10	0.302658	0.068342	0.496578	0.13
14	1.864251	0.188302	1.074649	0.164358	0.279142	5.19E-10	0.283914	0.068769	0.504557	0.14
15	1.711749	0.192375	1.113127	0.178296	0.26933	1.07E-09	0.266578	0.068941	0.511399	0.15
16	1.573194	0.195576	1.150893	0.192575	0.260895	2.11E-09	0.250471	0.068883	0.517267	0.16
17	1.446926	0.197947	1.188212	0.207208	0.253641	3.99E-09	0.235442	0.068615	0.522291	0.17
18	1.33153	0.19953	1.225309	0.222205	0.247407	7.27E-09	0.221366	0.068158	0.52658	0.18

19	1.225798	0.200369	1.262371	0.237577	0.242057	1.28E-08	0.208141	0.067526	0.530224	0.19
20	1.128693	0.200503	1.299551	0.253337	0.237476	2.19E-08	0.195678	0.066735	0.533296	0.2
21	1.039327	0.199972	1.336968	0.269496	0.233567	3.63E-08	0.183905	0.065797	0.535855	0.21
22	0.95693	0.198817	1.374705	0.286067	0.230246	5.90E-08	0.172762	0.064724	0.537951	0.22
23	0.880843	0.197073	1.412813	0.303065	0.227441	9.36E-08	0.162197	0.063527	0.539622	0.23
24	0.810489	0.194778	1.451311	0.320504	0.22509	1.46E-07	0.152168	0.062217	0.5409	0.24
25	0.745371	0.191969	1.490179	0.338399	0.223136	2.22E-07	0.142638	0.060803	0.54181	0.25
26	0.685054	0.188684	1.529366	0.356767	0.221531	3.34E-07	0.133579	0.059295	0.542371	0.26
27	0.629157	0.184958	1.568784	0.375623	0.220232	4.93E-07	0.124964	0.057702	0.542595	0.27
28	0.577345	0.180831	1.60831	0.394987	0.219198	7.17E-07	0.116773	0.056035	0.542492	0.28
29	0.529323	0.17634	1.647788	0.414878	0.218395	1.03E-06	0.108988	0.054303	0.542068	0.29
30	0.484828	0.171525	1.687028	0.435316	0.217789	1.46E-06	0.101596	0.052516	0.541326	0.3
31	0.443625	0.166427	1.72581	0.456323	0.21735	2.04E-06	0.094583	0.050684	0.540266	0.309999
32	0.405501	0.161085	1.763885	0.477924	0.21705	2.82E-06	0.087939	0.048817	0.538887	0.319999
33	0.370264	0.155543	1.800982	0.500142	0.216863	3.87E-06	0.081655	0.046926	0.537184	0.329999
34	0.337737	0.149841	1.836811	0.523005	0.216766	5.25E-06	0.075722	0.045021	0.535155	0.339998
35	0.307755	0.144022	1.871068	0.546541	0.216737	7.05E-06	0.070131	0.043113	0.532794	0.349998
36	0.280166	0.138127	1.903443	0.570782	0.216755	9.39E-06	0.064875	0.041211	0.530099	0.359997
37	0.254825	0.132198	1.933625	0.59576	0.216801	1.24E-05	0.059947	0.039326	0.527064	0.369995
38	0.231594	0.126272	1.961311	0.62151	0.21686	1.62E-05	0.055337	0.037466	0.523688	0.379994
39	0.210343	0.120388	1.986211	0.648069	0.216916	2.11E-05	0.051037	0.035642	0.519967	0.389992
40	0.190944	0.11458	2.008056	0.675478	0.216956	2.73E-05	0.047037	0.033861	0.515902	0.399989
41	0.173276	0.108881	2.026603	0.703778	0.216969	3.50E-05	0.043328	0.032131	0.511494	0.409986
42	0.157222	0.10332	2.041639	0.733014	0.216946	4.46E-05	0.039899	0.030459	0.506746	0.419981
43	0.142668	0.097923	2.052989	0.763233	0.216879	5.65E-05	0.036737	0.02885	0.501662	0.429976
44	0.129503	0.092711	2.060515	0.794485	0.216762	7.12E-05	0.033832	0.02731	0.496249	0.439969
45	0.117623	0.087703	2.064119	0.826821	0.216591	8.91E-05	0.031171	0.025842	0.490514	0.44996
46	0.106925	0.082914	2.063747	0.860298	0.216363	0.000111	0.02874	0.024449	0.484468	0.459949
47	0.097314	0.078354	2.059382	0.894971	0.216077	0.0001375	0.026527	0.023133	0.478121	0.469935
48	0.088696	0.074032	2.051049	0.930901	0.215732	0.0001695	0.024518	0.021896	0.471487	0.479919
49	0.080984	0.069952	2.038811	0.96815	0.21533	0.0002079	0.0227	0.020737	0.464577	0.489898
50	0.074095	0.066115	2.022761	1.006783	0.214873	0.0002538	0.02106	0.019657	0.457407	0.499873
51	0.067953	0.062519	2.003027	1.046865	0.214364	0.0003085	0.019584	0.018653	0.449991	0.509843
52	0.062485	0.059162	1.979761	1.088466	0.213806	0.0003733	0.018261	0.017725	0.442344	0.519806
53	0.057624	0.056037	1.953141	1.131656	0.213204	0.0004497	0.017077	0.01687	0.434483	0.529762
54	0.053308	0.053138	1.92336	1.176508	0.212563	0.0005396	0.016021	0.016085	0.426422	0.539709
55	0.049479	0.050455	1.89063	1.223094	0.211888	0.0006449	0.015083	0.015368	0.418178	0.549645
56	0.046086	0.047979	1.855171	1.27149	0.211185	0.0007677	0.01425	0.014715	0.409766	0.55957
57	0.043081	0.0457	1.817214	1.321771	0.210458	0.0009104	0.013515	0.014123	0.401202	0.569481
58	0.040421	0.043607	1.776993	1.374013	0.209715	0.0010756	0.012866	0.013589	0.392501	0.579376
59	0.038065	0.041688	1.734745	1.428291	0.208961	0.0012661	0.012296	0.013109	0.383678	0.589253
60	0.035978	0.039932	1.690706	1.48468	0.208202	0.0014851	0.011796	0.012679	0.374747	0.599109
61	0.034129	0.038328	1.645109	1.543254	0.207445	0.0017359	0.011359	0.012296	0.365723	0.608941
62	0.032487	0.036865	1.598186	1.604085	0.206695	0.002022	0.010977	0.011957	0.35662	0.618746

63	0.031028	0.035531	1.550159	1.667243	0.205959	0.0023475	0.010645	0.011658	0.347452	0.628521
64	0.029728	0.034317	1.501246	1.732793	0.205244	0.0027165	0.010357	0.011395	0.338232	0.638261
65	0.028566	0.033211	1.451656	1.800797	0.204555	0.0031332	0.010106	0.011166	0.328974	0.647963
66	0.027524	0.032205	1.40159	1.871313	0.203899	0.0036024	0.009888	0.010968	0.31969	0.657622
67	0.026584	0.031288	1.351238	1.944392	0.203282	0.004129	0.009698	0.010796	0.310395	0.667234
68	0.025733	0.030452	1.300783	2.020079	0.20271	0.0047181	0.009533	0.010649	0.3011	0.676792
69	0.024957	0.029689	1.250396	2.098411	0.20219	0.0053748	0.009387	0.010524	0.291818	0.686291
70	0.024245	0.02899	1.200237	2.179417	0.201728	0.0061048	0.009259	0.010418	0.282563	0.695727
71	0.023586	0.028349	1.150457	2.263116	0.20133	0.0069135	0.009143	0.010328	0.273346	0.705091
72	0.022972	0.027758	1.101196	2.349518	0.201002	0.0078067	0.009039	0.010252	0.264179	0.714379
73	0.022394	0.02721	1.052583	2.438621	0.20075	0.0087901	0.008942	0.010188	0.255076	0.723583
74	0.021846	0.026701	1.004737	2.530412	0.20058	0.0098696	0.00885	0.010133	0.246047	0.732696
75	0.021322	0.026223	0.957767	2.624864	0.200498	0.0110508	0.008762	0.010085	0.237106	0.741712
76	0.020817	0.025772	0.911769	2.72194	0.20051	0.0123395	0.008676	0.010043	0.228263	0.750622
77	0.020326	0.025344	0.866832	2.821586	0.20062	0.0137411	0.008589	0.010005	0.219531	0.759419
78	0.019845	0.024933	0.823033	2.923738	0.200835	0.0152609	0.008501	0.009968	0.21092	0.768096
79	0.019373	0.024536	0.780439	3.028317	0.201159	0.0169042	0.008411	0.009931	0.202442	0.776646
80	0.018905	0.024149	0.739109	3.135228	0.201596	0.0186757	0.008316	0.009894	0.194107	0.785059
81	0.01844	0.023769	0.699093	3.244365	0.202151	0.0205798	0.008216	0.009853	0.185924	0.79333
82	0.017976	0.023393	0.660428	3.355608	0.202828	0.0226206	0.008112	0.009809	0.177904	0.801451
83	0.017512	0.023018	0.623147	3.468825	0.20363	0.0248016	0.008001	0.00976	0.170056	0.809415
84	0.017048	0.022643	0.587272	3.58387	0.20456	0.027126	0.007884	0.009706	0.162388	0.817214
85	0.016583	0.022266	0.552819	3.70059	0.20562	0.0295961	0.00776	0.009645	0.154907	0.824843
86	0.016117	0.021884	0.519795	3.818817	0.206813	0.032214	0.00763	0.009576	0.147622	0.832296
87	0.015649	0.021497	0.4882	3.938378	0.20814	0.034981	0.007494	0.0095	0.140537	0.839566
88	0.01518	0.021104	0.458029	4.05909	0.209601	0.0378979	0.007351	0.009416	0.133659	0.84665
89	0.014711	0.020704	0.429269	4.180765	0.211197	0.0409645	0.007202	0.009323	0.126993	0.853542
90	0.014241	0.020297	0.401902	4.30321	0.212928	0.0441805	0.007047	0.009222	0.120541	0.860238
91	0.013772	0.019882	0.375907	4.42623	0.214793	0.0475445	0.006887	0.009112	0.114308	0.866735
92	0.013305	0.019459	0.351255	4.549625	0.21679	0.0510546	0.006721	0.008994	0.108295	0.87303
93	0.01284	0.01903	0.327916	4.673199	0.218918	0.0547084	0.006552	0.008867	0.102504	0.879121
94	0.012378	0.018593	0.305855	4.796754	0.221173	0.0585027	0.006378	0.008732	0.096935	0.885007
95	0.011919	0.01815	0.285033	4.920098	0.223554	0.0624339	0.006201	0.008589	0.091589	0.890688
96	0.011466	0.017701	0.265411	5.043039	0.226057	0.0664979	0.006022	0.008438	0.086463	0.896162
97	0.011019	0.017247	0.246946	5.165395	0.228678	0.0706899	0.00584	0.008281	0.081556	0.901431
98	0.010578	0.016789	0.229595	5.286988	0.231412	0.0750048	0.005657	0.008117	0.076866	0.906495
99	0.010144	0.016328	0.213312	5.407647	0.234257	0.0794371	0.005473	0.007948	0.072389	0.911357
100	0.009719	0.015864	0.19805	5.527213	0.237205	0.0839811	0.005289	0.007773	0.068122	0.916019
101	0.009302	0.015399	0.183765	5.645533	0.240254	0.0886305	0.005106	0.007593	0.06406	0.920483
102	0.008895	0.014934	0.170409	5.762466	0.243397	0.0933789	0.004923	0.00741	0.060199	0.924753
103	0.008498	0.01447	0.157936	5.87788	0.246629	0.09822	0.004741	0.007223	0.056533	0.928833
104	0.008112	0.014007	0.1463	5.991656	0.249944	0.1031469	0.004561	0.007033	0.053057	0.932727
105	0.007736	0.013547	0.135456	6.103684	0.253337	0.1081529	0.004384	0.006841	0.049765	0.936439
106	0.007372	0.013091	0.125361	6.213866	0.256803	0.1132312	0.00421	0.006648	0.04665	0.939975

107	0.007019	0.012639	0.115971	6.322116	0.260336	0.1183751	0.004038	0.006453	0.043707	0.943339
108	0.006678	0.012193	0.107245	6.428358	0.263929	0.1235778	0.00387	0.006259	0.040929	0.946536
109	0.006349	0.011753	0.099142	6.532526	0.267579	0.1288327	0.003706	0.006065	0.03831	0.949572
110	0.006032	0.01132	0.091624	6.634566	0.271279	0.1341333	0.003545	0.005871	0.035841	0.952453
111	0.005727	0.010895	0.084654	6.734434	0.275024	0.1394732	0.003389	0.005679	0.033519	0.955185
112	0.005434	0.010478	0.078195	6.832094	0.278808	0.1448463	0.003237	0.005488	0.031334	0.957772
113	0.005153	0.010069	0.072215	6.92752	0.282628	0.1502465	0.00309	0.005299	0.029281	0.960221
114	0.004884	0.00967	0.066681	7.020696	0.286478	0.1556682	0.002948	0.005113	0.027354	0.962538
115	0.004626	0.009281	0.061562	7.11161	0.290353	0.1611058	0.00281	0.00493	0.025546	0.964728
116	0.00438	0.008902	0.05683	7.200263	0.29425	0.1665541	0.002676	0.00475	0.023851	0.966797
117	0.004144	0.008533	0.052457	7.286658	0.298163	0.1720079	0.002548	0.004574	0.022263	0.968751
118	0.00392	0.008175	0.048418	7.370806	0.302089	0.1774626	0.002424	0.004401	0.020776	0.970594
119	0.003706	0.007828	0.044689	7.452725	0.306024	0.1829138	0.002306	0.004232	0.019385	0.972333
120	0.003503	0.007491	0.041246	7.532435	0.309965	0.1883571	0.002191	0.004067	0.018084	0.973972
121	0.003309	0.007165	0.038069	7.609964	0.313908	0.1937885	0.002082	0.003907	0.016867	0.975516
122	0.003125	0.00685	0.035138	7.685342	0.31785	0.1992045	0.001977	0.00375	0.015731	0.97697
123	0.002951	0.006546	0.032435	7.758602	0.321788	0.2046016	0.001877	0.003598	0.014669	0.97834
124	0.002785	0.006253	0.029942	7.829782	0.32572	0.2099765	0.001781	0.003451	0.013678	0.979629
125	0.002628	0.005971	0.027643	7.89892	0.329642	0.2153264	0.001689	0.003308	0.012753	0.980842
126	0.002479	0.005699	0.025524	7.96606	0.333552	0.2206484	0.001601	0.00317	0.01189	0.981983
127	0.002338	0.005437	0.02357	8.031244	0.337449	0.2259402	0.001518	0.003036	0.011085	0.983056
128	0.002205	0.005186	0.021769	8.094517	0.341329	0.2311994	0.001438	0.002906	0.010334	0.984065
129	0.002079	0.004945	0.020109	8.155925	0.345192	0.236424	0.001363	0.002782	0.009635	0.985013
130	0.001959	0.004714	0.018579	8.215516	0.349036	0.2416121	0.00129	0.002661	0.008983	0.985904
131	0.001847	0.004492	0.017169	8.273337	0.352859	0.246762	0.001222	0.002545	0.008375	0.986742
132	0.00174	0.004279	0.015868	8.329436	0.35666	0.2518722	0.001157	0.002434	0.007809	0.987529
133	0.00164	0.004076	0.01467	8.383861	0.360437	0.2569414	0.001095	0.002326	0.007282	0.988268
134	0.001545	0.003881	0.013565	8.436659	0.364189	0.2619684	0.001036	0.002223	0.00679	0.988962
135	0.001456	0.003695	0.012546	8.487878	0.367915	0.2669521	0.00098	0.002124	0.006333	0.989615
136	0.001371	0.003518	0.011607	8.537565	0.371615	0.2718916	0.000927	0.002029	0.005907	0.990227
137	0.001292	0.003348	0.01074	8.585768	0.375288	0.2767863	0.000877	0.001937	0.00551	0.990803
138	0.001217	0.003186	0.009941	8.632531	0.378933	0.2816353	0.00083	0.00185	0.00514	0.991343
139	0.001146	0.003031	0.009204	8.677901	0.382548	0.2864382	0.000785	0.001766	0.004796	0.991851
140	0.00108	0.002883	0.008524	8.721921	0.386135	0.2911945	0.000742	0.001685	0.004476	0.992328
141	0.001017	0.002743	0.007896	8.764635	0.389692	0.2959039	0.000702	0.001608	0.004177	0.992776
142	0.000958	0.002609	0.007317	8.806085	0.393219	0.3005661	0.000664	0.001535	0.003899	0.993196
143	0.000903	0.002481	0.006782	8.846313	0.396715	0.3051809	0.000627	0.001464	0.003641	0.993591
144	0.00085	0.002359	0.006288	8.88536	0.400181	0.3097483	0.000593	0.001397	0.0034	0.993962
145	0.000801	0.002243	0.005832	8.923265	0.403616	0.3142681	0.000561	0.001332	0.003175	0.994311
146	0.000755	0.002133	0.005411	8.960066	0.40702	0.3187405	0.00053	0.001271	0.002966	0.994639
147	0.000711	0.002028	0.005022	8.9958	0.410393	0.3231655	0.000501	0.001212	0.002771	0.994947
148	0.00067	0.001928	0.004662	9.030504	0.413735	0.3275433	0.000474	0.001156	0.00259	0.995236
149	0.000631	0.001833	0.004329	9.064212	0.417045	0.3318739	0.000448	0.001102	0.00242	0.995508
150	0.000595	0.001743	0.004021	9.096959	0.420325	0.3361577	0.000424	0.001051	0.002263	0.995763

Table A. 4: The result of performance measures which fill the push out probability α_i

j	num1	num2	num3	num4	α1	α2	α3	α4
1	6.411911	0.03349	0.203952	0.010103	0.671051	5.34E-07	2.69E-11	0.671051
2	5.53476	0.056551	0.341343	0.02042	0.556065	1.19E-05	7.12E-10	0.556076
3	4.849458	0.075474	0.45081	0.030963	0.461013	7.17E-05	5.64E-09	0.461085
4	4.294273	0.091666	0.541593	0.041742	0.388795	0.000241	2.57E-08	0.389035
5	3.832641	0.105853	0.619136	0.052764	0.333651	0.00058	8.48E-08	0.33423
6	3.4415	0.118412	0.686866	0.064039	0.290847	0.001133	2.27E-07	0.29198
7	3.105431	0.129543	0.747083	0.075574	0.25702	0.001917	5.21E-07	0.258938
8	2.81357	0.139364	0.801427	0.087379	0.229817	0.00293	1.07E-06	0.232748
9	2.557914	0.147961	0.851121	0.099461	0.207579	0.00415	2.01E-06	0.211731
10	2.332356	0.155404	0.897117	0.111828	0.189121	0.005546	3.51E-06	0.194671
11	2.132104	0.161762	0.940172	0.12449	0.173586	0.007084	5.80E-06	0.180676
12	1.953328	0.167102	0.9809	0.137454	0.160345	0.008731	9.13E-06	0.169085
13	1.792916	0.171489	1.019804	0.150731	0.148929	0.010453	1.38E-05	0.159396
14	1.648316	0.174987	1.057298	0.16433	0.138986	0.012223	2.02E-05	0.151229
15	1.517418	0.177659	1.093722	0.17826	0.130247	0.014017	2.87E-05	0.144293
16	1.398462	0.179561	1.129355	0.192532	0.122504	0.015816	3.97E-05	0.13836
17	1.289977	0.18075	1.16442	0.207155	0.115596	0.017602	5.38E-05	0.133252
18	1.190724	0.181275	1.199096	0.222142	0.109394	0.019363	7.15E-05	0.128828
19	1.099657	0.181185	1.233515	0.237503	0.103794	0.02109	9.34E-05	0.124977
20	1.015889	0.180525	1.267773	0.253251	0.098712	0.022775	0.00012	0.121607
21	0.938663	0.179335	1.301929	0.269398	0.094082	0.024411	0.000153	0.118646
22	0.867332	0.177657	1.336008	0.285957	0.089845	0.025996	0.000192	0.116033
23	0.801342	0.175528	1.370002	0.302942	0.085955	0.027525	0.000238	0.113718
24	0.740212	0.172985	1.403872	0.320368	0.082372	0.028996	0.000292	0.111661
25	0.68353	0.170063	1.437551	0.33825	0.079061	0.030408	0.000356	0.109826
26	0.630932	0.166796	1.470943	0.356605	0.075994	0.031759	0.00043	0.108183
27	0.582104	0.16322	1.503927	0.375449	0.073145	0.033048	0.000514	0.106707
28	0.536768	0.159368	1.536358	0.3948	0.070492	0.034275	0.000611	0.105377
29	0.494675	0.155274	1.56807	0.414677	0.068015	0.035438	0.000721	0.104174
30	0.455608	0.150971	1.598881	0.435101	0.065699	0.036537	0.000845	0.103081
31	0.419367	0.146493	1.628592	0.456092	0.063527	0.037572	0.000985	0.102084
32	0.385773	0.141872	1.656994	0.477674	0.061487	0.038542	0.00114	0.101169
33	0.354661	0.137141	1.683872	0.49987	0.059567	0.039448	0.001312	0.100327
34	0.32588	0.132333	1.709008	0.522705	0.057757	0.040289	0.001501	0.099546
35	0.299289	0.127477	1.732187	0.546206	0.056046	0.041064	0.001709	0.098819
36	0.274755	0.122603	1.7532	0.570399	0.054427	0.041775	0.001935	0.098137
37	0.252153	0.117741	1.771849	0.595315	0.052892	0.042423	0.00218	0.097495
38	0.231364	0.112916	1.787953	0.620983	0.051435	0.043006	0.002444	0.096885
39	0.212274	0.108153	1.801346	0.647436	0.050049	0.043528	0.002727	0.096304
40	0.194774	0.103476	1.811887	0.674706	0.04873	0.043988	0.003029	0.095747
41	0.178761	0.098905	1.819459	0.702828	0.047472	0.044388	0.003351	0.095211

42	0.164133	0.094457	1.823971	0.731836	0.046271	0.044731	0.00369	0.094692
43	0.150793	0.090148	1.825359	0.761768	0.045123	0.045018	0.004048	0.094188
44	0.138651	0.085993	1.823588	0.79266	0.044025	0.045251	0.004423	0.093698
45	0.127615	0.082	1.818651	0.824551	0.042974	0.045432	0.004814	0.09322
46	0.117603	0.078179	1.810567	0.857479	0.041966	0.045565	0.005221	0.092752
47	0.108532	0.074535	1.799384	0.891483	0.041001	0.045652	0.005642	0.092295
48	0.100326	0.071073	1.785171	0.926602	0.040074	0.045695	0.006077	0.091847
49	0.092913	0.067794	1.768021	0.962875	0.039186	0.045699	0.006524	0.091408
50	0.086224	0.064697	1.748046	1.000341	0.038332	0.045664	0.006983	0.090979
51	0.080195	0.061783	1.725377	1.039037	0.037513	0.045595	0.007452	0.09056
52	0.074766	0.059046	1.700159	1.079002	0.036725	0.045495	0.007929	0.090149
53	0.06988	0.056484	1.672549	1.120269	0.035969	0.045365	0.008414	0.089748
54	0.065486	0.05409	1.642714	1.162874	0.035243	0.045209	0.008906	0.089357
55	0.061535	0.051859	1.610829	1.206848	0.034544	0.04503	0.009402	0.088976
56	0.057982	0.049784	1.577073	1.252221	0.033874	0.044829	0.009903	0.088605
57	0.054788	0.047857	1.54163	1.299019	0.033229	0.04461	0.010406	0.088245
58	0.051913	0.046072	1.504684	1.347265	0.032609	0.044374	0.010911	0.087894
59	0.049325	0.044418	1.466418	1.396979	0.032014	0.044124	0.011416	0.087554
60	0.046991	0.04289	1.427013	1.448177	0.031442	0.043862	0.01192	0.087224
61	0.044883	0.041479	1.386649	1.500867	0.030892	0.043589	0.012423	0.086904
62	0.042975	0.040176	1.345499	1.555057	0.030364	0.043307	0.012923	0.086594
63	0.041244	0.038973	1.303732	1.610744	0.029856	0.043018	0.013419	0.086293
64	0.039668	0.037864	1.261511	1.667923	0.029368	0.042723	0.01391	0.086001
65	0.038229	0.03684	1.218991	1.726581	0.028899	0.042423	0.014395	0.085717
66	0.036909	0.035894	1.17632	1.786696	0.028448	0.042121	0.014873	0.085442
67	0.035693	0.035019	1.13364	1.848243	0.028014	0.041815	0.015344	0.085174
68	0.034567	0.034208	1.091083	1.911186	0.027597	0.041509	0.015806	0.084912
69	0.033518	0.033455	1.048773	1.975483	0.027196	0.041202	0.016258	0.084656
70	0.032537	0.032753	1.006828	2.041082	0.026809	0.040896	0.0167	0.084405
71	0.031612	0.032098	0.965355	2.107925	0.026437	0.04059	0.017132	0.084159
72	0.030736	0.031482	0.924452	2.175945	0.026078	0.040286	0.017551	0.083915
73	0.0299	0.030902	0.884211	2.245066	0.025732	0.039985	0.017958	0.083675
74	0.0291	0.030353	0.844715	2.315205	0.025398	0.039685	0.018352	0.083435
75	0.028328	0.029829	0.806036	2.386272	0.025075	0.039389	0.018732	0.083197
76	0.02758	0.029327	0.768242	2.458168	0.024764	0.039096	0.019098	0.082958
77	0.026852	0.028844	0.731391	2.530787	0.024462	0.038807	0.019449	0.082718
78	0.02614	0.028375	0.695532	2.604017	0.02417	0.038521	0.019784	0.082475
79	0.025441	0.027917	0.660709	2.67774	0.023886	0.03824	0.020104	0.08223
80	0.024753	0.027467	0.626958	2.751833	0.023611	0.037962	0.020408	0.081981
81	0.024075	0.027024	0.594307	2.826169	0.023343	0.037689	0.020695	0.081728
82	0.023404	0.026584	0.562778	2.900615	0.023083	0.037419	0.020966	0.081469
83	0.022739	0.026145	0.532388	2.975038	0.022829	0.037154	0.02122	0.081203
84	0.02208	0.025707	0.503146	3.0493	0.022582	0.036893	0.021456	0.080931
85	0.021427	0.025267	0.475058	3.123266	0.02234	0.036635	0.021676	0.080651

86	0.020779	0.024825	0.448121	3.196797	0.022104	0.036382	0.021878	0.080364
87	0.020135	0.024378	0.422331	3.269757	0.021872	0.036132	0.022063	0.080067
88	0.019497	0.023928	0.397678	3.342013	0.021645	0.035886	0.022231	0.079762
89	0.018865	0.023472	0.374147	3.413432	0.021422	0.035643	0.022382	0.079447
90	0.018238	0.023012	0.351721	3.483887	0.021203	0.035404	0.022516	0.079122
91	0.017618	0.022546	0.330379	3.553255	0.020987	0.035167	0.022634	0.078788
92	0.017006	0.022074	0.310097	3.621417	0.020774	0.034933	0.022735	0.078443
93	0.016401	0.021598	0.290848	3.688262	0.020565	0.034702	0.022821	0.078087
94	0.015805	0.021117	0.272603	3.753685	0.020358	0.034473	0.02289	0.077721
95	0.015218	0.020632	0.255333	3.817587	0.020154	0.034246	0.022945	0.077345
96	0.014641	0.020143	0.239005	3.879878	0.019952	0.034021	0.022985	0.076958
97	0.014075	0.019651	0.223586	3.940476	0.019753	0.033798	0.023011	0.076561
98	0.01352	0.019157	0.209043	3.999306	0.019555	0.033576	0.023022	0.076153
99	0.012977	0.018661	0.195339	4.056302	0.01936	0.033355	0.023021	0.075736
100	0.012446	0.018164	0.182441	4.111407	0.019166	0.033135	0.023007	0.075308
101	0.011928	0.017667	0.170314	4.164573	0.018974	0.032917	0.02298	0.074871
102	0.011423	0.017171	0.158921	4.215758	0.018784	0.032699	0.022942	0.074425
103	0.010932	0.016676	0.148229	4.26493	0.018596	0.032481	0.022894	0.073971
104	0.010456	0.016184	0.138203	4.312065	0.018409	0.032264	0.022834	0.073507
105	0.009993	0.015696	0.128809	4.357147	0.018224	0.032048	0.022765	0.073036
106	0.009545	0.015211	0.120015	4.400165	0.01804	0.031831	0.022686	0.072557
107	0.009111	0.014732	0.111788	4.441118	0.017858	0.031615	0.022599	0.072071
108	0.008692	0.014257	0.104097	4.480011	0.017677	0.031398	0.022503	0.071579
109	0.008288	0.01379	0.096912	4.516854	0.017498	0.031182	0.0224	0.07108
110	0.007898	0.013329	0.090203	4.551663	0.017321	0.030965	0.02229	0.070575
111	0.007523	0.012875	0.083943	4.58446	0.017145	0.030748	0.022173	0.070065
112	0.007162	0.01243	0.078105	4.615271	0.01697	0.030531	0.02205	0.069551
113	0.006815	0.011993	0.072664	4.644128	0.016797	0.030314	0.021921	0.069032
114	0.006482	0.011565	0.067593	4.671065	0.016626	0.030096	0.021787	0.06851
115	0.006163	0.011146	0.062871	4.696121	0.016456	0.029879	0.021649	0.067984
116	0.005857	0.010736	0.058475	4.719336	0.016288	0.029661	0.021506	0.067455
117	0.005565	0.010337	0.054384	4.740755	0.016122	0.029442	0.02136	0.066924
118	0.005285	0.009948	0.050577	4.760424	0.015957	0.029224	0.02121	0.066391
119	0.005017	0.009569	0.047037	4.77839	0.015794	0.029006	0.021057	0.065856
120	0.004762	0.0092	0.043746	4.794703	0.015632	0.028787	0.020901	0.065321
121	0.004519	0.008842	0.040686	4.809413	0.015472	0.028569	0.020743	0.064784
122	0.004286	0.008494	0.037842	4.822572	0.015314	0.02835	0.020583	0.064248
123	0.004065	0.008157	0.035199	4.834232	0.015157	0.028132	0.020422	0.063711
124	0.003854	0.007831	0.032743	4.844444	0.015003	0.027914	0.020258	0.063175
125	0.003654	0.007515	0.030462	4.853262	0.01485	0.027696	0.020094	0.06264
126	0.003463	0.007209	0.028342	4.860736	0.014698	0.027478	0.019929	0.062106
127	0.003282	0.006914	0.026374	4.866919	0.014549	0.027261	0.019763	0.061573
128	0.00311	0.006629	0.024545	4.871862	0.014401	0.027044	0.019596	0.061041
129	0.002946	0.006353	0.022847	4.875616	0.014255	0.026828	0.019429	0.060512

130	0.002791	0.006088	0.02127	4.878229	0.014111	0.026612	0.019262	0.059985
131	0.002643	0.005833	0.019805	4.879752	0.013968	0.026397	0.019095	0.05946
132	0.002503	0.005586	0.018444	4.880231	0.013827	0.026183	0.018928	0.058938
133	0.002371	0.005349	0.01718	4.879713	0.013688	0.025969	0.018762	0.058419
134	0.002245	0.005122	0.016006	4.878245	0.01355	0.025756	0.018596	0.057903
135	0.002126	0.004902	0.014915	4.875869	0.013415	0.025545	0.018431	0.05739
136	0.002013	0.004692	0.013902	4.87263	0.013281	0.025334	0.018266	0.05688
137	0.001906	0.00449	0.01296	4.868569	0.013148	0.025124	0.018102	0.056374
138	0.001804	0.004296	0.012085	4.863727	0.013018	0.024915	0.017939	0.055872
139	0.001708	0.004109	0.011272	4.858142	0.012889	0.024707	0.017777	0.055373
140	0.001617	0.003931	0.010516	4.851853	0.012762	0.024501	0.017617	0.054879
141	0.001531	0.003759	0.009813	4.844896	0.012636	0.024295	0.017457	0.054388
142	0.00145	0.003595	0.009159	4.837306	0.012512	0.024091	0.017298	0.053902
143	0.001373	0.003438	0.008551	4.829117	0.012389	0.023889	0.017141	0.053419
144	0.0013	0.003287	0.007986	4.820361	0.012269	0.023687	0.016985	0.052941
145	0.001231	0.003143	0.00746	4.811071	0.012149	0.023487	0.01683	0.052467
146	0.001166	0.003005	0.00697	4.801276	0.012032	0.023289	0.016677	0.051998
147	0.001104	0.002872	0.006514	4.791004	0.011916	0.023092	0.016525	0.051533
148	0.001046	0.002746	0.00609	4.780284	0.011801	0.022896	0.016375	0.051072
149	0.000991	0.002625	0.005695	4.769143	0.011688	0.022702	0.016226	0.050616
150	0.000938	0.002509	0.005327	4.757604	0.011576	0.02251	0.016078	0.050165

Table A. 5: The general procedure that was used to compute overall system performance as a function of Z

blocking probability 1,2,3	0.819244	0.214364	0.240254	0.423574	0.551647	0.633706
blocking probability 4	1.75E-15	0.000308	0.08863	0.340395	0.502628	0.601606
push out probability 1	0.67105	0.037567	0.01907	0.011483	0.007448	0.005218
push out probability 2	8.57E-08	0.047313	0.03408	0.022567	0.014863	0.010432
push out probability 3	1.40E-12	0.006722	0.023271	0.015991	0.010423	0.007305
push out probability 4	0.67105	0.091603	0.076421	0.05004	0.032734	0.022955
GOS	1.490293	0.306275	0.405305	0.814009	1.087009	1.258267
utilization	1	0.998071	0.997242	0.999522	0.999945	0.999992
Z	0.671009	3.258739	2.460474	1.227901	0.919905	0.794737

الملخص

إن إنترنت الأشياء هي واحدة من النماذج التكنولوجية الحديثة التي تجسد امتداد للشبكات الحالية غير المتجانسة بحيث تتيح التواصل بين الإنسان والإنسان ، الإنسان والأشياء وبين الأشياء و الأشياء التي يتم دمجها في شبكة المعلومات لتساعد في تبادل المعلومات بشكل تلقائي ومعالجة البيانات في الوقت الحقيقي. وقد اجتذب هذا المجال انتباه الباحثين لدراسة إمكانات تطبيق إنترنت الأشياء في مختلف المجالات مع الأخذ في الاعتبار التحديات المختلفة التي تواجه هذه التكنولوجيا. في هذه الرسالة ، قمنا بتحليل لأداء تطبيق حقيقي موجود في المدن الذكية باستخدام سلاسل ماركوفيان الزمنية المستمرة (CTMC). في هذا التطبيق تم مراعاة وجود أربعة أنواع من الرسائل ذات أولويات مختلفة، وباستخدام نظام انتظار الطوابير المحدود ، قمنا باقتراح نموذج تحليلي جديد فعال من حيث التكلفة مع آلية لدفع الرسائل بالإضافة لمنح الأولوية للرسائل الطارئة والتي تمثل الأعلى أولوية في هذا النظام. واستناداً إلى النموذج التحليلي المقترح ، تم إجراء العديد من مقاييس الأداء لمختلف فئات الرسائل على نطاق واسع بحيث شملت احتمالات الحجب، احتمالات الدفع، معدل التأخير، معدل استغلال الشبكة وتأثيرها في الأداء العام للنظام.