FORK-JOIN NETWORKS IN HEAVY TRAFFIC: DIFFUSION APPROXIMATIONS AND CONTROL

ASAF ZVIRAN

FORK-JOIN NETWORKS IN HEAVY TRAFFIC: DIFFUSION APPROXIMATIONS AND CONTROL

RESEARCH THESIS

SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN OPERATIONS
RESEARCH AND SYSTEMS ANALYSIS

ASAF ZVIRAN

SUBMITTED TO THE SENATE OF THE TECHNION – ISRAEL INSTITUTE OF TECHNOLOGY

ADAR (BET), 5771 HAIFA MARCH, 2011

THE RESEARCH THESIS WAS DONE IN THE FACULTY OF INDUSTRIAL ENGINEERING AND MANAGEMENT, UNDER THE SUPERVISION OF PROFESSOR RAMI ATAR FROM THE ELECTRICAL ENGINEERING FACULTY AND PROFESSOR AVISHAI MANDELBAUM FROM THE INDUSTRIAL ENGINEERING FACULTY.

I AM DEEPLY GRATEFUL TO MY ADVISORS RAMI AND AVISHAI FOR THEIR BELIEF IN ME, THEIR SUPPORT, AND THEIR ENDLESS EFFORT THROUGHOUT OUR MUTUAL WORK. I AM GLAD THAT I HAD THE OPPORTUNITY TO LEARN SO MUCH FROM YOU.

REGARDING THE INDUSTRIAL ENGINEERING FACULTY, ALTHOUGH I COULD NOT COME MORE FREQUENTLY, WHENEVER I CAME TO THE FACULTY, I FELT AT HOME.

I DEDICATE THIS THESIS TO MY WIFE AVITAL AND DAUGHTER ELLA, YOU ARE MY INSPIRATION.

THE GENEROUS FINANCIAL HELP OF TECHNION IS GRATEFULY ACKNOWLEDGE

Contents

List of Figures

Abstract

This work addresses the problem of analysis and control of fork-join networks in the conventional Heavy-Traffic diffusion regime. Standard fork-join networks are feed-forward, which are relatively easy to control. Motivated by healthcare systems, we allow probabilistic feedback, which turns the problem into a challenging one.

In our models, activities are associated uniquely with customers. They are hence non-exchangeable in the sense that one can not combine/join activities associated with different customers - this is the case in healthcare (e.g. emergency departments) and multi-project environments (in contrast to assembly networks).

We introduce a natural concept of optimality for our model, and then solve for the optimal control, asymptotically in heavy-traffic. The central ingredient in the proof is the establishment of asymptotic equivalence between non-exchangeable and exchangeable dynamics.

1 Introduction

1.1 Background

The following sections survey the relevant work regarding Fork-Join networks, Heavy Traffic analysis and Scheduling and Routing policies. Sections 1.1.2, 1.1.4 and part of Section 1.1.1 were adapted from Neguyen [33], and Section 1.1.3 from Shaikhet [4].

1.1.1 Fork-Join Networks: Definition and Some Applications

A fork-join network consists of a group of service stations, which serve the arriving customers simultaneously and sequentially according to preset deterministic precedence constraints. More specifically, one can think in terms of "jobs" arriving to the system over time, each job consisting of different tasks that are to be executed according to the precedence constraints. The job may leave the system only after all its tasks have been completed. The distinguishing features of this model class are the so-called "fork" and "join" constructs. A *fork* occurs whenever several tasks are being processed at the same time. In the network model, this is represented by a "splitting" of the job into multiple tasks, which are then sent simultaneously to their respective servers. A *join* node, on the other hand, corresponds to a task that may not be initiated until several other tasks have been completed. Components are joined only if they correspond to the same job; thus a join is always preceded by a fork. If the last stage of an operation consists of multiple tasks, then these tasks regroup into a single job before departing the system.

Examples of Fork-Join Networks
In Fig. 1.1 we see the process progressing from the Arrest of "alleged" criminals until getting them to trial (arraignment). As shown, the process consists of three simultaneous paths—the path of the arrestee, the path of the arresting officer, and the path of the arrestee information through the system. This example is taken from Larson's article [16] on "Improving the N.Y.C A-to-A System".
In Fig. 1.2 we see the process from the arrival of an order to build a house until the completion of the numerous tasks required in the construction plan. In the graph, the construction orders "split" and "join" throughout the system till all the tasks are completed; the precedence constraints take the form of a flow chart.

Fork-Join networks are natural models for a variety of processes including communication and computer systems, manufacturing and project management (as introduced in Fig.

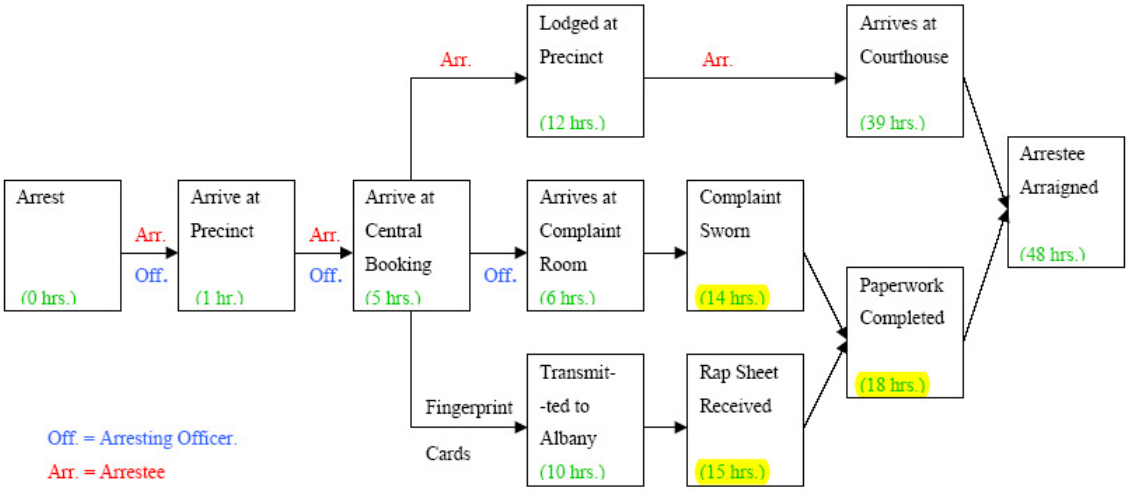


Figure 1.1: Arrest-to-Arraignment process

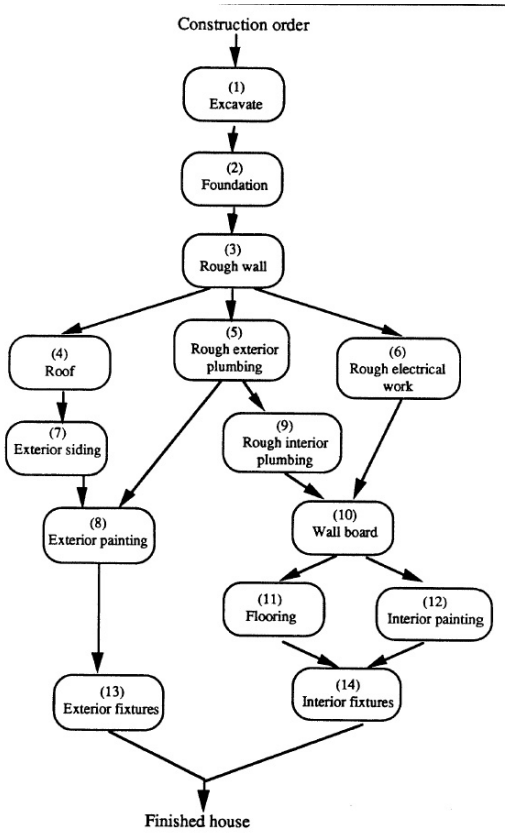


Figure 1.2: Construction of a House

1.2) and service systems (as introduced in Fig. 1.1). A fork-join computer or telecommunication network typically represents the processing of computer programs, data packets, etc., which involve parallel multitasking and the splitting and joining of information. In manufacturing, prevalent fork-join networks are assembly networks, which represents the assembly of a product or system that requires several parts which are processed simultaneously at separate workstations or plant locations.

Fork-join networks can be found frequently in the health-care system in general, and hospitals in particular (see Fig. 1.3), in which patients and their medical files, test results and insurance policy may split and join in different parts of the process, in order to get to the final task, that may be admitting a patient to the wards, starting an operation, etc. Another reason for the need of a fork-join network in hospitals is the necessity to join and synchronize many separate resources—doctors, nurses, room/bed, special equipment—in order to perform one integrated operation. In this research we develop and implement some mathematical approximations that support practical aspects in the analysis and control of hospital processes.

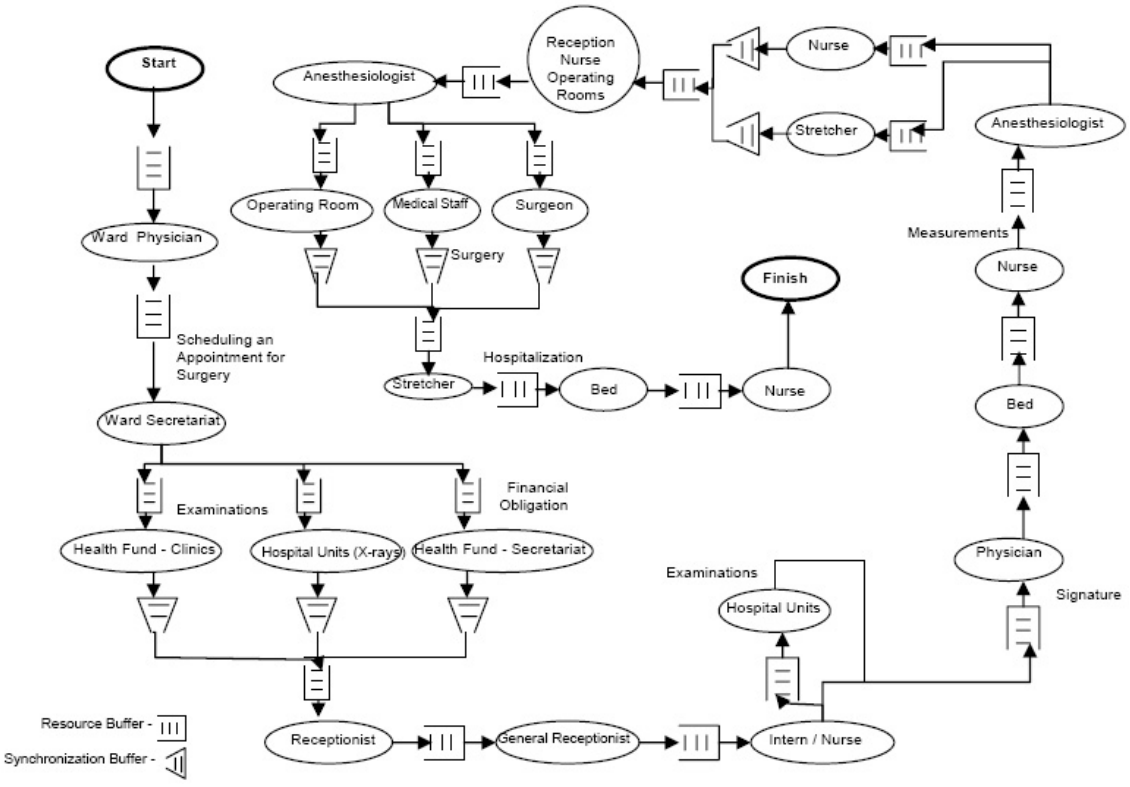


Figure 1.3: Fork-join Networks in Hospitals—Preparation to Surgery

1.1.2 Heavy Traffic Analysis of Queueing Networks

Consider single-station queueing systems with multiple arrival streams and multiple servers, operating under the FIFO discipline, Iglehart and Whitt [15] showed that for a sequence of such systems, with the traffic intensities approaching one, normalized versions of the queue length process converge to a one-dimensional reflected Brownian motion. The parameters of the RBM are completely specified in terms of the first and second moments of the distributions of the interarrival times and service times. Related results are also obtained for the departure, workload, and virtual waiting time processes. In [37], Whitt generalizes this result to a single queue with several priority classes of customers and a preemptive-resume discipline. Whitt obtained heavy traffic limit theorems for the queue length and unfinished work processes associated with each priority class. A significant contribution of this paper, in comparison with earlier work, is that Whitt obtains joint convergence for the various processes of interest.

This analysis was carried out by Harrison in [12], in which he treats the case of two tandem queues, the simplest example of a feedforward queueing network. In [12] Harrison was the first to characterize the heavy traffic limit of certain processes associated with a queueing network as a multi-dimensional RBM on an orthant, describing the directions of reflection as well as the drift vector and covariance matrix. This characterization enables Harrison to derive partial differential equations associated with various system performance measures. Although no general solution is available, he was able to compute the solutions for certain special cases.

Peterson [28] generalizes this treatment to a network with deterministic feedforward routing and multiple customer types. At each station in the network, customer types are partitioned into two classes, one of which has preemptive-resume priority over the other. The designation of priority class for each type is allowed to vary over stations, and within each priority class customers are served in a FIFO manner. The main result in [28] is a heavy traffic limit theorem showing convergence of the workload and queue length processes to an RBM in an orthant. The representation obtained by Peterson enables him to deduce a simple expression for sojourn time processes in terms of the workload processes. The analysis in [28] begins by generalizing the results of Iglehart-Whitt for a single-station network to the setting of multiple customer types. In an "inductive" manner, the analysis is then extended to all stations in the network.

So far we have restricted our discussion to the class of networks with deterministic and feedforward routing. The first significant heavy traffic approximation result for an open queueing network with feedback was obtained by Reiman [7]. In [7], he investigates a network model with Markovian switching between stations, whose customers have general

interarrival time and service time distributions. This network has been called a generalized Jackson network [20]. Reiman proves a heavy traffic limit theorem which shows that the workload and queue length processes converge to an RBM whose state space is an orthant. He also shows that the limiting process for sojourn times is given by a simple transformation of the workload processes.

The analysis was extended by Chen and Mandelbaum [17], who analyzed generalized Jackson networks in which not all stations operate in heavy traffic. A station is said to be a bottleneck if the traffic intensity at that station is greater than or equal to one, and is said to be a non-bottleneck otherwise. A bottleneck stations is said to be balanced if the traffic intensity is equal to one, and it is called a strict bottleneck if the traffic intensity exceeds one. Roughly speaking, the results of Chen and Mandelbaum quantify the common belief that bottleneck stations dominate the performance of the network. They show that in the heavy traffic scaling, queue lengths and workloads vanish at non-bottleneck stations, and they tend to infinity at the strict bottleneck stations. When properly centered, however, the queue lengths and workloads at strict bottleneck stations converge to a simple functional of Brownian motion. The limit of the subnetwork composed of the balanced stations, as one may expect, corresponds to an RBM of the appropriate dimension.

The analog of Reiman's model with multiple customer types presents significant difficulties. Reiman [8] has completed analysis of a single-station network with feedback which is populated by several types of customers. In the work of Mandelbaum and Stolyar [23], they treat the parallel server models (J nonidentical servers working in parallel and I customer classes) and convex cost functions. Finally, they conjectured about the asymptotically optimal solution for model with feedback.

Recent Work by Katsuda [32] study the multiclass queueing networks (MQNs). In the first part of his work he consider the MQNs for which the fluid stability is valid, state-space collapse is exhibited under suitable initial conditions and a heavy traffic limit theorem holds. For such MQNs we establish that, under the assumption of the tightness of a sequence of stationary scaled workloads, the sequence converges to the stationary distribution of semimartingale reflecting Brownian motion in the heavy-traffic regime. In the second part, using the result obtained, it is shown that such a convergence of stationary workload holds for a multiclass single-server queue with feedback routing.

Finally, for a recent general framework on Brownian approximations, one may refer to Harrison [18].

1.1.3 Scheduling and Routing of Queueing Networks

Optimal scheduling and routing present the most interesting and difficult challenges in the management of queueing networks. The routing problem is to determine, upon an arrival, which of the available servers, if any, should we assign to serve a customer. The scheduling problem is to indicate, upon service completion, which of the available waiting customers, if any, should be served.

The earliest control work for single server queues was the exact analysis by Cox and Smith [14]. They looked at a multi-class single-station network (M/G/1) with linear waiting cost, i.e. one pays $c_i\tau$ units for each job of class i that waits for service τ units of time. This is equivalent to looking at $\int_o^t \sum_i c_i Q_i(s)ds$ - the integral over a linear combination of the queue lengths. They proved the classical $c\mu$ rule, which can be described as follows. With each class of jobs we associate an index $c_i\mu_i$ (with μ_i being its service rate) and at a decision point one always serves the highest index. See Walrand [9] for various extensions.

A similar setting was considered in the conventional heavy traffic asymptotic regime by Van Mieghem [11], with his generalized $c\mu$ rule. This culminated in the work of Mandelbaum and Stolyar [23]. They treat the parallel server models (J nonidentical servers working in parallel and I customer classes) and convex cost functions $C_i(\cdot), i = 1, \ldots, I$. Optimal scheduling corresponds to the following: at each time t, when server j becomes idle, it chooses for a service a type i customer with the largest $C_i'(Q_i(t))\mu_i j$. Note however that the cost functions $C(\cdot)$ in [23] were restricted to convex functions with $C(0) = 0$ and $C'(0) = 0$. This excludes a direct application to linear costs.

For linear delay costs, one is referred to Williams [10] and to Harrison [13] and Harrison and Lopez [18]. In [18] it is proved that for the parallel server models, the diffusion control problem exhibits a massive state-space collapse and is reduced from multi-dimension to one-dimension, which is much easier to solve. This is done by the striking "equivalent workload formulation". See also Harrison and Van Meighem [24].

The difficulty in [24] is that the asymptotic solution does not have a clear interpretation within the prelimit model. Bell and Williams [25] proved that for the 2-parallel servers model the asymptotically optimal policy is a threshold policy; i.e., the priority of service depends on whether the queue lengths are below or above certain levels - thresholds. The subsequent paper [26] of the same authors deals with extending the threshold strategy to parallel server systems.

Results are less established for networks with many-server stations. By taking the QED diffusion scaling (taking the number of servers N to infinity in an appropriate manner),

Armony and Maglaras [2] model and analyze rational customers in equilibrium; they treat jointly the problem of optimal control and staffing. Harrison and Zeevi [27] analyze the diffusion control problem associated with a single pool (multiple customer classes) model with linear costs. Specifically, they show that this control problem has an optimal Markov control policy (cf. [22]) which is characterized in terms of its underlying Hamilton- Jacobi-Bellman (HJB) equation.

The works of Atar, Mandelbaum and Reiman [19] and Atar [29] and [29] established asymptotic optimality of policies in the QED regime, for treelike models (the J nodes, which correspond to the J multi-server stations and the I nodes, corresponding to the I classes, jointly constitute a tree). The scaling then enforces convergence of the prelimit control problem to a diffusion control problem which can be dealt with by stochastic control methods, namely via the HJB equations. Then a method is provided on how to translate the obtained solutions into prelimit policies. The diffusion limit problem arises as a formal weak limit of a preemptive network scheduling problem, i.e. one where a service to a customer can be stopped at any moment and resumed at a later time, possibly in a different station. In the prelimit, the behavior is clearly different for non-preemptive networks, but it is proved that this difference vanishes asymptotically.

Few papers have dealt with Scheduling and Routing of parallel processing systems and fork-join networks. The work of Avi-Itzhak and Halfin [30] introduces the question of non-preemptive priority assignment in multi-class fork-join queue. The model considered in their work consist of a fork-join queue (simple fork-join) with m single-servers in parallel and m classes of customers. A customer belongs to class $k = 1, 2, \ldots, m$ if it is a k-split customer, i.e., it forks into k out of the m servers. They showed that there is an advantage in granting priority to non-splitting jobs (class-1) unless their service times are relatively large. Their result established exact optimality based on a simple coupling argument. An interesting observation may be using the m classes model to represent customers in different phases of service. By this analogy, the non-preemptive priority to class-1 may be equivalently described as: At each route, assign non-preemptive priority to customers whose service was completed in all other routes. This is the same policy we propose and analyze for a broader class of networks.

Cohen, Mandelbaum and Shtub [1] examined control mechanisms for project management in a multi-project environment. They surveyed a variety of buffer management techniques in open and closed systems. Our work examine the control problem which arises in their work by means of optimal control techniques.

1.1.4 Heavy Traffic Analysis of Fork-Join Networks

Although heavy traffic analysis has been applied successfully to conventional open queueing networks, not much progress has been reported regarding its applicability to the study of networks with fork and join constructs. An analysis of a fork-join model was carried out by Varma [31]. He considers both heavy traffic and light traffic approximations for this class of networks. Varma proves a limit theorem showing that in heavy traffic the workload levels converge to a process which is given by a functional of a multi-dimensional Brownian motion. For the special case of a fork-join queue, he obtains a characterization for the invariant distribution of the throughput time in the limiting process. When the fork-join queue consists of two symmetric queues, he was able to solve for all moments of the throughput time. However, he was not able to characterize the invariant measure of the general fork-join network.
This analysis was expanded by Nguyen [34], in which she provided characterization of certain limiting processes for a fork-join network as reflected Brownian motions. Unlike the RBM described in Section 1.1.2, the RBM which arises from this analysis lives in a state space that is a polyhedral cone in an orthant. This compact and elegant representation enables her to obtain an analytical characterization of the stationary distributions associated with the limiting process. In particular, she derives conditions which guarantee the stability of the limiting RBM.

1.2 Preliminaries and Notations

1.2.1 Organization of The Thesis

In this thesis, we address the problem of analysis and control of *Fork-Join Networks* in the conventional Heavy Traffic Regime.
In Section 2 we introduce the problem of task synchronization in Fork-Join networks with *non-exchangeable* customers. i.e., customers which are uniquely labeled in the sense that one can not join tasks associated with different customers. We show that, in this setting, there may be dependencies that degrades system's performance. We then formulate a control problem and criteria for maximum throughput in *Exact and Asymptotic optimality problems* via an analogy to *Assembly networks* and proved them be efficient for a general class of networks. Finally, some simple examples are considered.
Section 3 introduces a model with multi-server stations, which is the simplest setting in which Exact optimality seems intractable. Under this setting, we prove an *asymptotic*

optimality of the FCFS policy in the conventional Heavy Traffic regime.
Section 4 introduces a model with feedback. In this setting, FCFS is no longer asymptotically optimal, thus solving for optimal scheduling is hard. Under this setting, a new control policy is proposed and *asymptotic optimality* is proven in conventional Heavy Traffic. Finally, in Section 5 we summarize the contribution of the Thesis, and propose some worthy directions for future work.

1.2.2 Notations

x^+	$= max\{x, 0\},\ x \in \mathbb{R}$.
x^-	$= max\{-x, 0\},\ x \in \mathbb{R}$.
$x \wedge y$	$= min\{x, y\},\ x, y \in \mathbb{R}$.
$x \vee y$	$= max\{x, y\},\ x, y \in \mathbb{R}$.
$\|x\|_T^*$	$= sup_{0 \le u \le T}\|x(u)\|$. A norm of an $\mathbb{R} - valued$ function.
P	Probability measure.
$\sigma\{\mathcal{A}\}$	The sigma-field generated by a collection $\mathcal{A}$ of random variables.
i	Route index.
j	Station index.
s	Server index.
m	Customer index.
α, γ	Policies index.
n	The Heavy Traffic index, construction of a "sequence of systems".
$\bar{X}^n(t) = \frac{X(nt)}{n}$	Fluid scaling.
$\hat{X}^n(t) = \frac{X(nt) - \lambda \cdot nt}{\sqrt{n}}$	Diffusion scaling, when λ denotes the process average rate.
$\Rightarrow$	Weak convergence of probability measures.

2 Problem Definition

In this section we introduce the problem of tasks synchronization in Fork-Join networks. We shall first introduce the network class considered and then the central concept of *non-exchangeable* customers. We then continue with a rigorous definition of the control problem. Finally, some simple examples are introduced for the proposed problem.

2.1 Network Models

This work addresses the problem of analysis and control of fork-join networks in the conventional Heavy-Traffic diffusion regime. In our work we shall consider two classes of network models: multi-server networks and single-server networks with feedback.

Model 1: Networks with Multi-Server Stations. Single-Server Fork-Join Networks are relatively easy to control. We thus allow multi-server networks, which turns the control problem into a challenging one. We shall represent that model with the following test case:

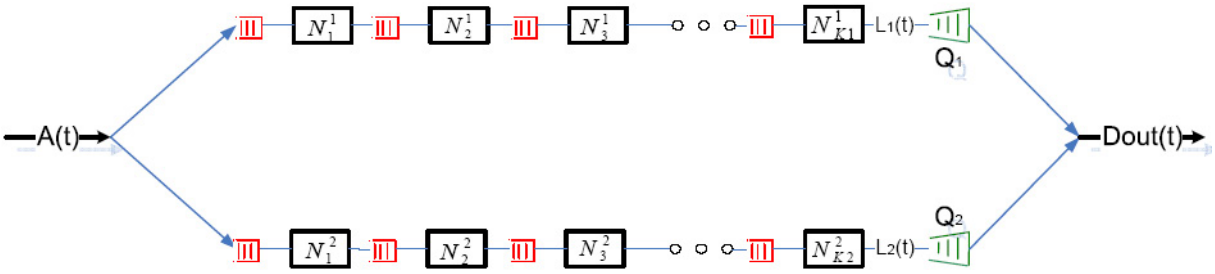


Figure 2.1: Multi-Server Simple Fork-Join

In the network of Figure 2.1, a job arriving to the system "forks" into tasks processed simultaneously in two parallel processing routes. Each route contains multiple service stations (queues) in tandem with multiple servers in each station. Complete jobs depart from the system only after the completion of the tasks associated with it. The completed task in each route waits in the *Synchronization Queues* (Q_1&Q_2) until tasks of both routes are completed and departure is permitted. In the context of this Example, a "join" node may be viewed as the merge of the completed tasks and the departure of the complete

job. In a broader sense, the join node may correspond to a service station where service may not be initiated until all the predecessor tasks have been completed.

Notations

- $A(t)$ - Arrival process: number of customers (jobs) arriving to the system till time t;
- $D_{out}(t)$ - Departure process: number of departures of (complete) customers till time t;
- $L_i(t)$ - Route departure process: number of departures of complete tasks from route i till time t;
- K_i - Number of service stations in route i;
- N_i^j - Number of servers in station j on route i;
- $Q_i(t)$ - Number of customers in the *synchronization queue* in route i at time t;

Model 2: Networks with Feedback. Standard fork-join networks are feed-forward. We shall consider a broader class of models that allow probabilistic feedback, as represented by the following test case:

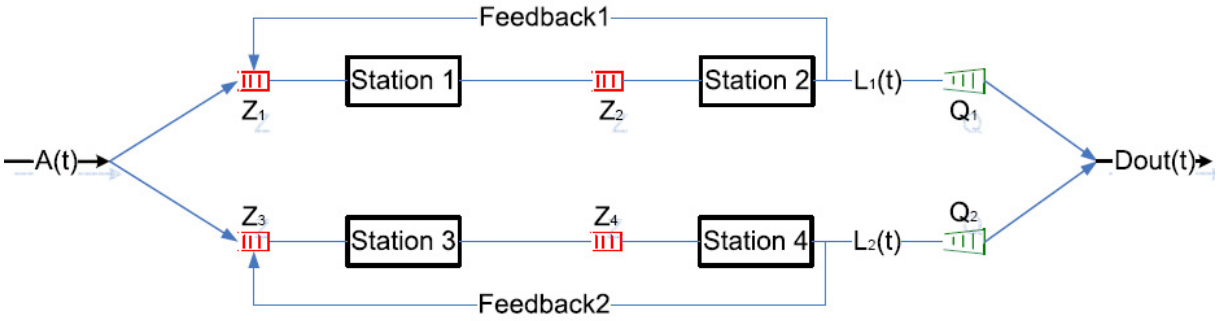


Figure 2.2: Fork-Join with Feedback

In this network, a job arriving to the system "forks" to tasks processed simultaneously in two parallel processing routes, each route consisting of two service stations in tandem, followed by a probabilistic feedback. The feedback may be viewed as a *quality check* at

the end of the processing line, in which unsatisfying tasks are sent back to be processed again.

Notations

- $A(t)$ - Arrival process: number of customers (jobs) arriving to the system till time t;
- $D_{out}(t)$ - Departure process: number of departures of (complete) customers till time t;
- $L_i(t)$ - Route departure process: number of departures of complete tasks from route i till time t;
- $D_j(t)$ - Station departure process: number of departures of complete tasks from station j till time t;
- $Z_j(t)$ - Number of customers in the *resource queue* preceding station j, at time t;
- $Q_i(t)$ - Number of customers in the synchronization queue in route i, at time t.

The model in Figure 4.1 seems to be the simplest setting of a Fork-Join network where solving for optimal scheduling is challenging.

We shall assume the following for both models throughout our work.

Model Assumptions:

- Poisson arrival process of homogeneous customers, with arrival rate λ.
- Exponential service durations with rate μ_j, for the iid servers in station j

2.2 Customers and Tasks

In our model, *tasks* are associated uniquely with *customers*. They are hence *non-exchangeable* in the sense that one can not join tasks associated with different customers. This property can be viewed as though arriving customers receive a characterizing ID code which accompanies them through the system. This property is natural for a variety of applications

such as communication in which data packets are labeled, Project management, service systems and especially health-care. Indeed, in the health-care case, mixing a patient with another patient's blood test or medical file may have severe consequences.
The counter example to *non-exchangeable* networks are *Assembly Networks*, in which all tasks are *exchangeable*. Some basic differences between these two types of networks are now demonstrated in the following example

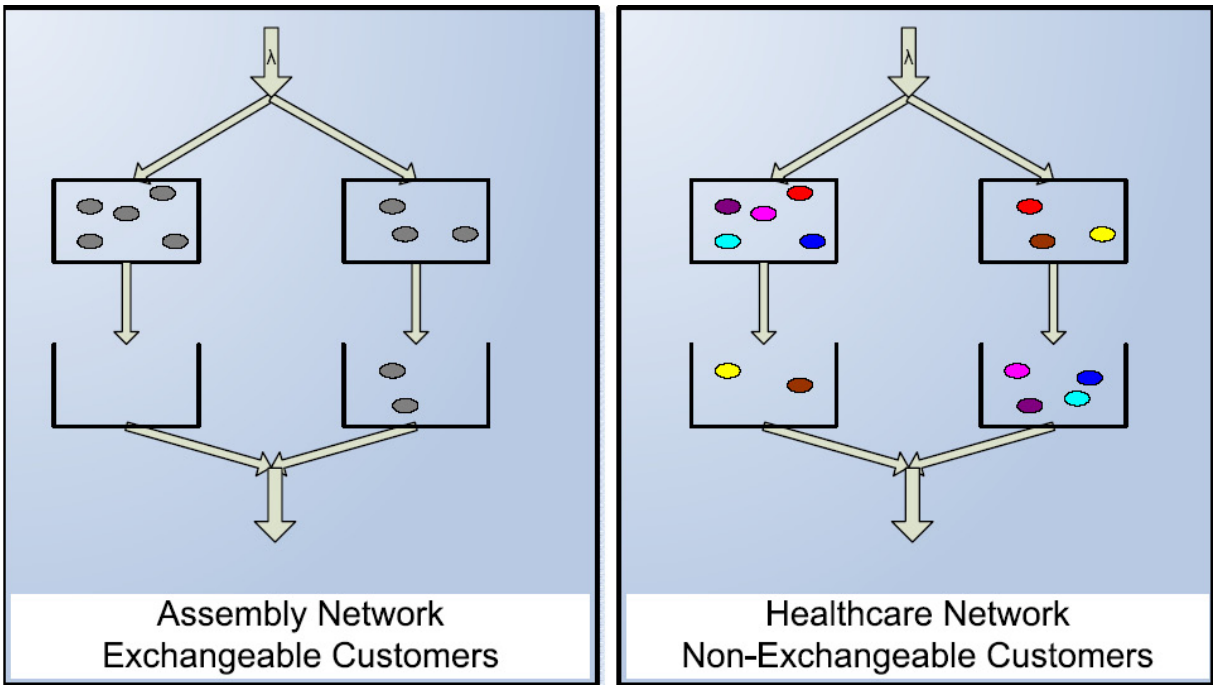


Figure 2.3: Assembly Networks vs. Health-care Networks

On the **left Figure** 2.3, we see an *Assembly Network* example in which an arriving job "forks" into two simultaneous manufacturing tasks. The waiting tasks are represented as the gray balls in the closed squares which represents the processing routes. The complete tasks then depart to the open squares which represent synchronization queues. The gray uniform color indicates that the complete tasks are identical and thus *exchangeable*, hence every complete pair of tasks from both routes can immediately "join" and depart the join node. One can see that this property can be expressed in terms of the following constraint: one of the synchronization queues must be empty at all times or, equivalently, the minimal synchronization queue equals zero at all time.
Thus,

$$Q_1(t) \cdot Q_2(t) = 0, \quad \text{or equivalently} \quad Q_1(t) \wedge Q_2(t) = 0. \tag{2.1}$$

We shall refer to Condition (2.1) as the **Complementarity Condition**.

On the **right figure** 2.3, we have a *Health-Care Network* example in which an arriving patient "forks" into two simultaneous tasks, for example x-ray and blood-test. The waiting tasks are represented as the colored balls in the closed squares such that every patient is uniquely labeled by a unique color. The complete tasks then depart to the open squares, which represents the synchronization queues. One notes that patients can be divided into two categories:

- Customers whose service is still incomplete in both routes, such as the red patient.
- Customers whose service was completed in one of the routes, but is yet incomplete in the other, such as the yellow and blue patients.

In addition, the customers' processing orders were not synchronized, causing the synchronization queues to be occupied by waiting patients while the join node is Idle. Indeed the waiting tasks in the synchronization queues are not associated with the same patient hence they are not allowed to join and depart the system.
Note that the number of waiting customers in the *Health-Care Network* is larger than the number of waiting customers in the *Assembly Network* even though in both networks the stations served the same number of tasks.

The scenario above has a positive probability (larger than zero) to occur in networks with multi-server station and / or probabilistic feedback. The reason for that behavior is due to customers disorder caused by the processing dynamic. In multi-server stations, customers bypass (overtake) each other within the stations due to the servers random service times. In the probabilistic feedback case, the feedback shuffles the customers order of departures due to the random decision to depart or return back to the start of the processing routes.

We deduce from the above discussion the important observation that Customers' *disorder* may cause increase of Idle-time in the join nodes and hence lower throughput.

2.3 Control Problem Formulation

A natural conclusion of the previous section is that some control mechanism is needed in order to help synchronize customers departure order. This control mechanism should have global information of customers status throughout the system and the ability to control customers priorities at each of the service stations in the system.
Heuristically, the ideal performance can be deduced through an analogy to *Assembly Network*, were the network is unrestricted by join constraints derived from customers uniqueness. Thus one may characterize the optimal performance by the **Complementarity condition** defined in Equation (2.1).

In the following section, we shall introduce a control problem that formulates the above discussion rigorously.

General Model Definition Let a complete probability space $(\Omega, \mathcal{F}, \mathbb{P})$ be given, supporting all random variables and stochastic processes defined below.
Consider the following general definition for a simple Fork-Join network:

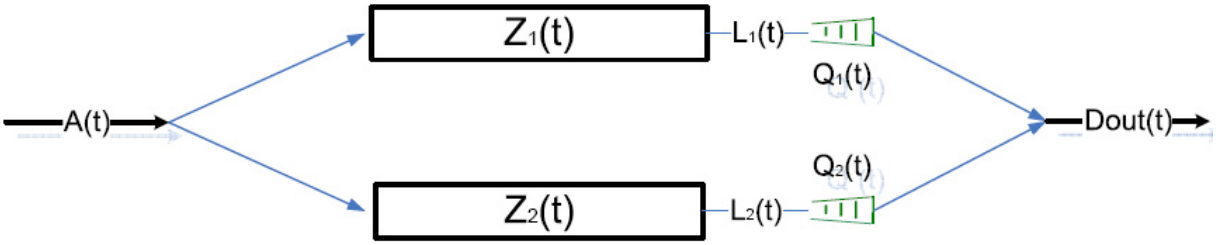


Figure 2.4: General Simple Fork-Join Network

The model consists of a stream of customers, a pair of routes, and two synchronization queues. Upon arrival, a customer "forks" into two tasks, each task being served in one of the routes, and each route representing a *General Jackson Network* with the restriction of exponential service times and markovian routing. A complete task is then queued at the synchronization queue, and customers leave the system only when both of their tasks have completed their service.

Notations

- $Z_i(t)$ − Total number of customers in the processing phase of route i at time t. i.e., customers who are still within the *Jackson Network*;
- $Q_i(t)$ − Number of customers in the synchronization queue of route i at time t;
- $A(t)$ − System's *External Arrival Process*, assumed to be a general renewal process. i.e., iid interarrival times;
- $L_i(t)$ − Route i's *Departure process*, which is the *Arrival Process* for the corresponding synchronization queue;
- $D_{out}(t)$ − System's *Departure process*.

System equations

We assume an empty system at $t = 0$, namely $Z_i(0) = Q_i(0) = 0$, $\forall i$.
Then

$$\begin{cases} Z_i(t) = A(t) - L_i(t); \\ \\ Q_i(t) = L_i(t) - D_{out}(t); \end{cases} \tag{2.2}$$

Optimal Control

Assume that the control policy is *Nonanticipating*, i.e., priority decisions are adapted to the natural filtration containing all customers processing times.[1] Let us define $\mathbb{I}$ as the set of all *admissible control policies* under the assumption above. Let $\gamma \in \mathbb{I}$ denote a specific control policy.

Definition 1. *We define exact optimality as achieving the maximum throughput, in the sense of maximum achievable number of departures over any finite time-interval* $[0, T]$. *i.e.,* γ *is the optimal policy over* $[0, T]$ *if* $\gamma = argmax_{\alpha \in \mathbb{I}}(D_{out}(T))$, *a.s.*

Proposition 2.1. *Each of the following conditions is equivalent to maximum throughput:*

- *Complementarity condition:* $Q_1(T) \wedge Q_2(T) = 0$, *a.s.,*
- *Minimal queues size:* $Q_1(T) + Q_2(T) = |L_1(T) - L_2(T)|$, *a.s.,*

for any fixed T.

[1] In particular, the decisions can not depend on future processing times.

Proposition 2.1 relies on the following relation

$$Q_1(t) + Q_2(t) = |L_1(t) - L_2(t)| + 2 \cdot Q_1(t) \wedge Q_2(t),$$

which is proved in the following.

Proof of Proposition 2.1.
The number of customers in the system at time t can be calculated by $N(t) = A(t) - D_{\text{out}}(t)$. Since $A(t)$ is primitive and thus uncontrollable, we have

$$argmax_{\gamma \in \mathbb{I}}(D_{out}(t)) = argmin_{\gamma \in \mathbb{I}}(N(t)).$$

Notice that a customer joins all routes simultaneously when arriving, and finishes service in the system only after finishing service in all routes. Therefore
$N(t) = Z_1(t) + Q_1(t) = Z_2(t) + Q_2(t)$, or equivalently $2 \cdot N(t) = \sum_j Z_j(t) + \sum_j Q_j(t)$. Hence

$$argmin_{\gamma \in \mathbb{I}}(N(t)) = argmin_{\gamma \in \mathbb{I}}(\sum_j Z_j(t) + \sum_j Q_j(t)).$$

We shall use the following lemma to establish our claim. The proof of the lemma will be provided following the proof of the proposition.

Lemma 2.1. *Assuming a homogeneous customer population and a work conserving policy, the processes $Z_i(t)$ and $L_i(t)$ do not depend on the control policy, for every route i.*

Remark- A policy is said to be *work conserving* if, for every t, a server can not be idle when the preceding queue is not empty. One can verify that any policy which is not work conserving can only decrease the number of departures over any finite time-interval (refer to the proof of the Lemma 2.1). Thus, in the following, we shall only consider work conserving policies.

The conclusion from the Lemma is that

$$argmax_{\gamma \in \mathbb{I}}(D_{out}(T)) = \text{ argmin }_{\gamma \in \mathbb{I}}(\sum_j Q_j(T)).$$

We now claim that $\text{ argmin }_{\gamma \in \mathbb{I}}(Q_1(T) + Q_2(T)) = \text{ argmin }_{\gamma \in \mathbb{I}}(Q_1(T) \wedge Q_2(T))$

As seen before, in Section (2.2), the customers may be divided into two categories, or classes:

- Class A: Customers whose service is still incomplete in both routes.
- Class B: Customers whose service has completed in one of the routes but is still incomplete in the other.

Let us use the following notations for this part:

- $Z_i^B(t)$ − The number of all Class B customers in the processing phase of route i at time t, i.e., customers whose service is still incomplete in route i but was completed in the opposite route;
- $Z_i^T(t)$ − The number of all customers in the processing phase of route i at time t;
- $Z_i^A(t)$ − The number of all Class A customers in the processing phase of route i at time t, i.e., customers whose service is still incomplete in both routes;

Notice that $Z_1^A(t) = Z_2^A(t) \leq Z_1^T(t) \wedge Z_2^T(t) = A(t) - L_1(t) \vee L_2(t)$, i.e., the set of Class A customers is identical in both routes and therefore must be smaller than $Z_1^T(t) \wedge Z_2^T(t)$. In addition, under the assumption of an empty system at $t = 0$, one can verify that $Q_1(t) = Z_2^B(t)$, which means that $Q_1(t) = Z_2^T(t) - Z_2^A(t)$. Now if we use the relations: $Z_2^T(t) = A(t) - L_2(t)$ and $Z_2^A(t) \leq A(t) - L_1(t) \vee L_2(t)$, we get

$$Q_1(t) \geq A(t) - L_2(t) - A(t) + L_1(t) \vee L_2(t) = L_1(t) \vee L_2(t) - L_2(t).$$

This yields the following relations:

$$\begin{cases} Q_1(t) \geq (L_1(t) - L_2(t))^+; \\ \\ Q_2(t) \geq (L_2(t) - L_1(t))^+; \end{cases} \tag{2.3}$$

$$Q_1(t) + Q_2(t) \geq |L_1(t) - L_2(t)|.$$

As we mentioned above, the number of total customers in both routes is always equal, meaning that $N(t) = A(t) - L_1(t) + Q_1(t) = A(t) - L_2(t) + Q_2(t)$. Therefore $Q_1(t) - Q_2(t) = L_1(t) - L_2(t)$. We conclude that

$$\begin{cases} Q_1(t) + Q_2(t) \geq |L_1(t) - L_2(t)|; \\ \\ |Q_1(t) - Q_2(t)| = |L_1(t) - L_2(t)|; \end{cases}$$

But $Q_1(t) + Q_2(t) = Q_1(t) \vee Q_2(t) - Q_1(t) \wedge Q_2(t) + 2 \cdot Q_1(t) \wedge Q_2(t)$.
Therefore we get the following relation:

$$Q_1(t) + Q_2(t) = |L_1(t) - L_2(t)| + 2 \cdot Q_1(t) \wedge Q_2(t),$$

from which we conclude

$Q_1(t) + Q_2(t) = |L_1(t) - L_2(t)|$, which is minimal value, if and only if $Q_1(t) \wedge Q_2(t) = 0$

Now recall that

$$argmax_{\gamma \in \mathbb{I}}(D_{out}(T)) = \text{ argmin }_{\gamma \in \mathbb{I}}(\sum_j Q_j(T)).$$

We get that a sufficient and necessary condition for maximum departures is minimum customers in the synchronization queues and, specifically, a policy is optimal if

- Complementarity condition: $Q_1(T) \wedge Q_2(T) = 0$, a.s.,
- Minimal queues size: $Q_1(T) + Q_2(T) = |L_1(T) - L_2(T)|$, a.s.,

for any fixed T. This completes the proof of Proposition 2.1. □

Proof of Lemma 2.1. Let us focus on a specific route i (the following can be applied to each route separately). The service system on route i can be represented as a *Jackson Network* with K stations, each with multi-servers and exponential service durations. Assume that $Z_i^k(t)$ is the total number of customers in station k at time t, i.e., the number of customers in resource queues and customers who receives service at t. Let us define the following sequences of standard Poisson processes $\{S_i^{k,s}(t),\ k \in (1,\dots,K),\ s \in (1,\dots,N^k)\}$, and the associated busyness processes $\{B_i^{k,s}(t)), k \in (1,\dots,K), s \in (1,\dots,N^k)\}$, for the separate servers in the stations. Additionally define $\{S_i^k(t),\ k \in (1,\dots,K)\}$, and the associated busyness processes $\{B_i^k(t)),\ k \in (1,\dots,K)\}$ for the complete stations. We assume that all underlying Poisson processes are mutually independent.
Therefore the system equations are

$$\begin{cases} Z_i^k(t) = Z_i^k(0) + A_i^k(t) - D_i^k(t) + \sum_{j=1}^K X_i^{j,k}(D_i^j(t)), \quad \text{for all } k \in \{1,\dots,K\}; \\ \\ L_i(t) = \sum_{j=1}^K X_i^{j,out}(D_i^j(t)); \end{cases} \tag{2.4}$$

When

- $A_i^k(t)$, External arrival process to station k.
- $B_i^{k,s}(t) = \int_0^t \mathbb{I}_{\{\text{ server s in station k is busy at time s }\}} ds$, Busyness process for server s in station k.
- $B_i^k(t) = \sum_{s=1}^{N_k} B_i^{k,s}(t)$, Cumulative busyness process for station k.
- $D_i^k(t) = \sum_{i=1}^{N^k} S_i^{k,s}(\mu_i^k B_i^{k,s}(t)) = S_i^k(\mu_i^k B_i^k(t))$, Departure process for station k.
- $X_i^{j,k}(M) = \sum_{\alpha=1}^{M} \xi_{i,\alpha}^{j,k}$, Internal arrival process to station k.
- $\xi_{i,\alpha}^{j,k}$, $\alpha \in \mathbb{N}$, defines a sequence of i.i.d random variables with Bernoulli-distribution (taking values 0/1), which denotes the feedback decision process from station j to station k.

Assume a work-conserving policy, we have

$$B_i^k(t) = \sum_{j=1}^{N_k} B_i^{k,j}(t) = \int_0^t (Z_i^k(s) \wedge N^k) ds.$$

Assume homogeneous customer population and *Nonanticipating* policy, then $S_i^k(t)$ do not depend on the priority decisions, i.e., the service process do not depend on customers identity. Also, the elements $Z_i^k(0)$, $A_i^k(t)$ and $\xi_{i,\alpha}^{j,k}$ are primitives, therefore uncontrollable.

Proposition 2.2. *Under the assumptions above,* $C_t \equiv \{Z_i^k(t), D_i^k(t), B_i^k(t), \ \ k \in \{1, \ldots, K\}\}$ *is a unique solution to the system equations (2.4).*

Proof of Proposition 2.2. Let us assume that C_t is not unique and prove a contradiction.
Assuming that there is a second solution, $C_t' \equiv \{Z_i'^k(t),\ D_i'^k(t),\ B_i'^k(t),\ \ k \in \{1, \ldots, K\}\}$, there exist $\tau < \infty$ s.t $\tau = \ \inf\{t : C_t \neq C_t'\}$. so $\exists t_n \downarrow \tau$ s.t $\ B_i^k(t_n) \neq B_i'^k(t_n)$ for at least one $k \in \{1, \ldots, K\}$. From the definition of τ we have $\forall t_n \uparrow \tau \ \ B_i^k(t_n) = B_i'^k(t_n)$ for all $k \in \{1, \ldots, K\}$. Thus from continuity of $B_i^k(t)$ we get $B_i^k(\tau) = B_i'^k(\tau)$. Thus

$$D_i^k(\tau) = S_i^k(\mu_i^k B_i^k(\tau)) = S_i^k(\mu_i^k B_i'^k(\tau)) = D_i'^k(\tau) \ \ \forall k.$$

But $S_i^k(t)$ is càdlàg (right continuous with left limit) function. It follows that

$$\exists \epsilon \ \text{s.t} \quad D_i^k(t) = D_i'^k(t) \quad \forall t \in [\tau, \tau + \epsilon), \ \forall k.$$

Therefore

$$\begin{cases} Z_i^k(t) = Z_i'^k(t), \quad \forall t \in [0,\tau),\ \forall k; \quad (\text{ by the definition of } \tau) \\ \\ Z_i^k(t) - Z_i'^k(t) = D_i^{k\prime}(t) - D_i^k(t) + \sum_{j=1}^{K} \sum_{\alpha=D_i^{j'}(t)\wedge D_i^j(t)}^{D_i^{j'}(t)\vee D_i^j(t)} \xi_{i,\alpha}^{j,k} = 0, \quad \forall t \in [\tau,\tau+\epsilon),\ \forall k; \end{cases}$$

Or equivalently

$$Z_i^k(t) = Z_i'^k(t), \quad \forall t \in [0,\tau+\epsilon),\ \forall k.$$

Hence we get

$$B_i^k(t) = \int_0^t (Z_i^k(s) \wedge N^k)ds = \int_0^t (Z_i'^k(t) \wedge N^k)ds = B_i'^k(t), \quad \forall t \in [0,\tau+\epsilon),\ \forall k.$$

Finally we may conclude that

$$(Z_i^k(t),\ D_i^k(t),\ B_i^k(t)) = (Z_i'^k(t),\ D_i'^k(t),\ B_i'^k(t)), \quad \forall t \in [0,\tau+\epsilon),\ \forall k \in \{1,\dots,K\}.$$

This is a contradiction to the existence of a second solution , and therefore completes the proof for Claim 2.2. □

Recall that $L_i(t)$, the route departure process, is defined by the following equation $L_i(t) = \sum_{j=1}^{K} X_i^{j,out}(D_i^j(t))$, hence it is also a unique solution. Note that by Proposition 2.2 the processes $Z_i(t)$ and $L_i(t)$ are unique solutions for the system equations, hence they are invariant to the control policy.
This completes the proof of Lemma 2.1. □

In addition, one can verify for policies which are not work-conserving, that

$$B_i^k(t) \le \int_0^t (Z_i^k(s) \wedge N^k)ds,$$

hence, by the proof above, it can be seen that the network throughput can only decrease by not work-conserving policies.

Asymptotically Optimal Control

We shall assume now that the system is in *Heavy Traffic* in a sense to be now defined. The precise formulation of our *Heavy Traffic* limit requires the construction of a "sequence of systems", indexed by n. It is assumed that the following relations hold:

- $\lambda^n = \lambda \cdot n + \hat{\lambda} \cdot \sqrt{n} + o(\sqrt{n})$.
- $\mu_j^n = \mu_j \cdot n + \hat{\mu}_j \cdot \sqrt{n} + o(\sqrt{n})$.
- Heavy Traffic Condition - Define the traffic intensity at station j to be ρ_j^n; it is assumed that there exists a deterministic number $-\infty < \theta_j < \infty$, such that $n^{\frac{1}{2}}(\rho_j^n - 1) \longrightarrow_n \theta_j$, as $n \longrightarrow \infty$, for each station j.

Here λ denotes the average external arrival rate, and μ_j denote the average departure rate from each station j, respectively.

Definition 2. *Policy $\gamma \in \mathbb{I}$ is asymptotically optimal if, given any other policy $\beta \in \mathbb{I}$ and a fixed T,*

$$\hat{D}_{out}^{n,\gamma}(T) \geq \hat{D}_{out}^{n,\beta}(T) - \epsilon(n), \quad \textit{with } \epsilon(n) \to 0 \textit{ in probability}.$$

Proposition 2.3. *Each of the following conditions is equivalent to the asymptotic optimality condition:*

- $\hat{Q}_1^n(T) \wedge \hat{Q}_2^n(T) \to 0$, *in probability ;*
- $\hat{Q}_1^{n,\gamma}(T) + \hat{Q}_2^{n,\gamma}(T) \leq \hat{Q}_1^{n,\beta}(T) + \hat{Q}_2^{n,\beta}(T) + \epsilon(n)$, $\epsilon(n) \to 0$ *in probability;*

for any fixed T.

Proof of Proposition 2.3.
Given that the first condition defined in Proposition 2.3 is valid, we shall prove that the asymptotic optimality condition defined above applies.
For any fixed T, and every $\epsilon > 0$

$$\begin{cases} \mathrm{P}(\hat{Q}_1^{n,\gamma}(T) \wedge \hat{Q}_2^{n,\gamma}(T) \leq \epsilon) \longrightarrow_n 1; \\ \\ Q_1(t) + Q_2(t) = |L_1(t) - L_2(t)| + 2 \cdot Q_1(t) \wedge Q_2(t); \end{cases}$$

Now, for any other policy $\beta \in \mathbb{I}$ we proved that $Q_1^\beta(t) + Q_2^\beta(t) \geq |L_1(t) - L_2(t)|$. Hence

$$Q_1^\gamma(T) + Q_2^\gamma(T) \leq Q_1^\beta(T) + Q_2^\beta(T) + 2 \cdot Q_1^\gamma(t) \wedge Q_2^\gamma(t).$$

Scaling by $(\sqrt{n})^{-1}$ and using the convention $\hat{Q}_i^n(t) = \frac{Q_i^n(t)}{\sqrt{n}}$ we get

$$\mathrm{P}(\hat{Q}_1^{n,\gamma}(T) + \hat{Q}_2^{n,\gamma}(T) \leq \hat{Q}_1^{n,\beta}(T) + \hat{Q}_2^{n,\beta}(T) + 2\epsilon) \longrightarrow_n 1.$$

Now using Lemma 2.1 we have $Z_1^{\gamma}(T) + Z_2^{\gamma}(T) = Z_1^{\beta}(T) + Z_2^{\beta}(T)$ for all $\gamma, \beta \in \mathbb{I}$

$$\mathrm{P}(\sum_k \hat{Q}_k^{n,\gamma}(T) + \sum_k \hat{Z}_k^{n,\gamma}(T) \leq \sum_k \hat{Q}_k^{n,\beta}(T) + \sum_k \hat{Z}_k^{n,\beta}(T) + 2\epsilon) \longrightarrow_n 1.$$

Using $2 \cdot N(t) = \sum_j Z_j(t) + \sum_j Q_j(t)$ and $\hat{N}^n(t) = \frac{N^n(t)}{\sqrt{n}}$

$$\mathrm{P}(2\hat{N}^{n,\gamma}(T) \leq 2\hat{N}^{n,\beta}(T) + 2\epsilon) \longrightarrow_n 1.$$

Now, using $N(T) = A(T) - D_{out}(T)$ and the scaling notations

$$\frac{N^n(t)}{\sqrt{n}} = \frac{A^n(t)-\lambda^n t}{\sqrt{n}} - \frac{D_{out}^n(t)-\mu^n t}{\sqrt{n}} + \frac{(\lambda^n-\mu^n)t}{\sqrt{n}};$$

$$\begin{cases} \hat{N}^n(t) = \frac{N^n(t)}{\sqrt{n}}; \\ \hat{A}^n(t) = \frac{A^n(t)-\lambda^n t}{\sqrt{n}}; \\ \hat{D}_{out}^n(t) = \frac{D_{out}^n(t)-\mu^n t}{\sqrt{n}}; \\ \hat{\lambda}^n - \hat{\mu}^n = \frac{(\lambda^n-\mu^n)}{\sqrt{n}}; \end{cases}$$

we get $\hat{N}^n(t) = \hat{A}^n(t) - \hat{D}_{out}^n(t) + (\hat{\lambda}^n - \hat{\mu}^n) \cdot t$.
Hence

$$\mathrm{P}([\hat{A}^n(T) - \hat{D}_{out}^n(T) + (\hat{\lambda}^n - \hat{\mu}^n)T]^{\gamma} \leq [\hat{A}^n(T) - \hat{D}_{out}^n(T) + (\hat{\lambda}^n - \hat{\mu}^n)T]^{\beta} + \epsilon) \longrightarrow_n 1.$$

But $\hat{A}^{n,\gamma}(T) = \hat{A}^{n,\beta}(T)$ and $[(\hat{\lambda}^n - \hat{\mu}^n)T]^{\gamma} = [(\hat{\lambda}^n - \hat{\mu}^n)T]^{\beta}$, and therefore

$$\mathrm{P}(\hat{D}_{out}^{n,\gamma}(T) \geq \hat{D}_{out}^{n,\beta}(T) - \epsilon) \longrightarrow_n 1.$$

This completes the proof of Proposition 2.3. □

2.4 Comment About Generalizations

Note that both the exact and asymptotic conditions may be generalized to any number of parallel processing routes. For M processing routes, the proof outline will consist of the following steps:

- $argmax_{\gamma\in\mathbb{I}}(D_{out}(t)) = argmin_{\gamma\in\mathbb{I}}(N(t))$;
- $M \cdot N(t) = \sum_j (Z_j(t)) + \sum_{i=1}^{M}(Q_i(t))$;
- Lemma 2.1 - Assuming homogeneous customer population, the processes $Z_i(t)$ and $L_i(t)$ do not depend on the priority policy, for every route i.
- $\sum_{i=1}^{M} Q_i(t) = \sum_{i=1}^{M} (L_i(t) - \bigwedge_{i\in\{1,..,M\}}(L_i(t))) + M \cdot \bigwedge_{i\in\{1,..,M\}}(Q_i(t))$;

The appropriate conditions in this case are

- Exact optimality: $\bigwedge_{i\in\{1,..,M\}}(Q_i(T)) = 0$, a.s.,
- Asymptotic optimality: $\bigwedge_{i\in\{1,..,M\}}(\hat{Q}_i^n(T)) \to 0$, in probability ,

for any fixed T.

Also note that the proofs for Propositions 2.1 and 2.3 can be generalized to general service time distribution. The only part of the proofs that needs to be generalized is the proof for Lemma 2.1.

2.5 Some Simple Examples

2.5.1 Single-Server Fork-Join Network with FCFS Discipline

Model definition In this section we shall consider the class of feed-forward fork-join networks consisting of K single server stations; the size K is insignificant to the analysis. We consider here an arbitrary topology, e.g., Figure 1.3. This section adopts the model definition presented in Neguyen [34].
Jobs arriving to the system are considered to be homogeneous in the sense that all jobs have the same route through the network. We further assume that tasks compete for resources at each station in a FCFS manner. At nodes that do not involve a joining of tasks, this simply means that the tasks enter service in the order of their arrival. At join nodes, the arrival time of a task is defined to be the arrival of the complete job, or equivalently, the arrival time of the last task associated with the specific job. Such a service discipline can be characterized as a local policy since it considers only station-level information. However, in the proposed model, the FCFS discipline does in fact preserve the ordering of the jobs throughout the system.

Is FCFS discipline optimal under Definition (2.1)? For any join node (station) j in the system, define $\beta((j))$ as the set of queues preceding j. Additionally we define $s(k)$ to be the source of queue k, that is the station whose output feeds into queue k.
Hence we claim: For every join node (station) j we have :

$$\min_{k\in\beta((j))} Q_k(t) \equiv 0.$$

Proof. Assume that the customers order of arrival is preserved throughout the system (see model definition). One can see that the following relations prevail, for any join node j:

$$\begin{cases} A_j(t) = \min\limits_{k\in\beta((j))} D_{s(k)}(t) & \text{Customers' ordering is preserved ;} \\ \\ Q_k(t) = D_{s(k)}(t) - A_j(t) = D_{s(k)}(t) - \min\limits_{k\in\beta((j))} D_{s(k)}(t), & \forall k \in \beta((j)); \end{cases}$$

Therefore, $\min\limits_{k\in\beta((j))} Q_k(t) = \min\limits_{k\in\beta((j))} D_{s(k)}(t) - \min\limits_{k\in\beta((j))} D_{s(k)}(t) \equiv 0$, by definition. This implies that the optimality criterion for *Exact Optimality* is fulfilled (see Section 2.3).

2.5.2 Single-Station Fork-Join with Feedback

Model Definition

Let us consider a specific model of a *Simple Fork-Join* network with probabilistic feedback. The network scheme is as follows:

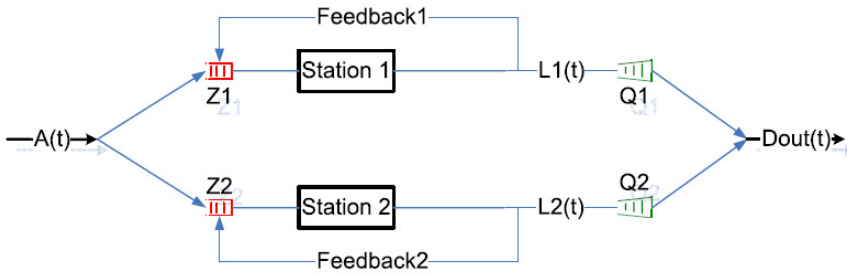


Figure 2.5: Simple Fork-Join with Feedback

Notations:

- $Z_i(t)$ – The number of tasks on route i at time t, waiting in queue i or being processed by server i.
- $Q_i(t)$ – The number of tasks in the synchronization queue i at time t.
- $L_i(t)$ – The number of departing tasks from route i until time t;
- $D_{\text{out}}(t)$ – The number of complete jobs departed from the system until time t;

Every task whose service is finished in route i has a probability p_i to depart to the *Synchronization Queue*, and probability $1 - p_i$ to be sent back to wait at the *Resource Queue*. This property may be viewed as a *quality check* at the end of the processing line, in which unsatisfying tasks are sent back to be processed again. In this model, we shall consider general arrival and service processes.

Optimal Control

The probabilistic feedback has a disordering effect on the customers departure order and may cause synchronization problems at the join node, which may cause $Q_1(t) \wedge Q_2(t)$ to be greater then zero. Thus the conventional FCFS discipline is not efficient in such systems. In conventional FCFS we refer to a priority policy in which tasks compete for resources according to their local arrival time to the station. It is easy to verify that,

with this discipline, every task that feedbacks joins the end of the *Resource Queue* and the customers order is not preserved.

Therefore, we suggest the following **control policy: Exhaustive Service.** Tasks will compete for resources according to their arrival time to the system (global arrival time), meaning that upon arrival every customer receives a unique ID number which determines the priority order of his tasks at every station in the system.
According to our policy, every task which feedbacks to the start of the processing route will enter service immediately without waiting. Thus according to this policy, the server and feedback block act jointly as a $G/G/1$ server, hence the system can be viewed as a *Single-Server Fork-Join Network with FCFS Discipline*, which has optimal performance as seen in the previous example.

2.5.3 Single-Station Fork-Join with Multi-Server

In this section we consider a network in which an arriving job "forks" into tasks processed simultaneously in two parallel processing routes, each route containing one service station with multiple servers. All completed tasks are waiting in the *Synchronization Buffers* until all tasks are completed and departure is then permitted. The network scheme is as follows:

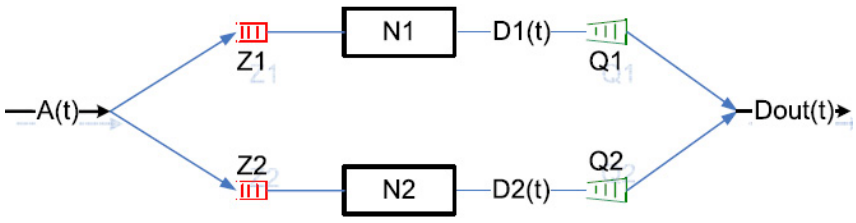


Figure 2.6: Simple Fork-Join with Multi-Servers

Notations:

- $A(t)$ - Arrival process: number of customers arriving to the system till time t;
- $D_{out}(t)$ - Departure process: number of departures of complete customers till time t;
- $D_i(t)$ - Route departure process: number of departures of complete tasks from route i till time t;

- N_i - Number of servers in route i;
- $Q_i(t)$ - Number of customers in the *synchronization queue* in route i on time t;

The two parallel multi-server stations seems to be the simplest setting where *Exact Optimality* is unreachable, meaning that $\mathrm{P}(Q_1(T) \wedge Q_2(T) > 0) > 0$, for any fixed T and any control policy. The reason is that customers overtake each other via the multi-server processing.

Is $Q_1(T) \wedge Q_2(T)$ at least bounded under the FCFS discipline?
We shall provide a positive answer to this question in a more general case in Section 3. But here we will give some intuition for the problem and its solution.
Assume that both routes have the same number of servers, meaning that $N_1 = N_2 = N$. Also assume that the system is under *Heavy Traffic* in the sense that all servers are busy all the time (Heuristic definition). Define $\tilde{V}_i(t)$ as the vector of all customers indices who enter service in station j till time t; the index vector is arranged according to the service initiation times. One can see that $|\tilde{V}_i(t)| = D_i(t) + N_i(t)$, assuming all servers are busy at time t, as defined before. But $D_i(t) = Q_i(t) + D_{out}(t), \quad \forall i$, therefore

$$|\tilde{V}_i(t)| = Q_i(t) + D_{out}(t) + N_i(t), \quad \forall i,$$

meaning that

$$|\tilde{V}_1(t)| - |\tilde{V}_2(t)| = (Q_1(t) - Q_2(t)) + (N_1(t) - N_2(t)) = Q_1(t) - Q_2(t).$$

Now assume that, for some fixed T, $Q_2(T) < Q_1(T)$, which means that $|\tilde{V}_2(T)| < |\tilde{V}_1(T)|$. Then, according to the property $|\tilde{V}_2(T)| < |\tilde{V}_1(T)|$ and the FCFS discipline, we can conclude the following statement:
Each customer index m whose task waits in Q_2 at time T (his service has completed in route 2) must be already in service in route 1; or, equivalently, if $m \in \tilde{Q}_2(T)$ then $m \in \tilde{V}_1(t)$. Now, since the last customer who enters service in route 1 is $|\tilde{V}_1(T)|$ (by the FCFS discipline) and $|\tilde{V}_2(T)| < |\tilde{V}_1(T)|$ then the same customer still waits at the resource queue in route 2. This implies that the maximum number of customers who are still in service in route 1, but their service have completed in route 2 (customers who are in Q_2), is bounded by $N - 1$. The same logic can be used for $Q_1(T) < Q_2(T)$ in order to get $Q_1(T) \leq N - 1$, for any fixed T.

Thus, under our model assumption, for any fixed T we have

$$\mathrm{P}(Q_1(T) \wedge Q_2(T) \geq N) = 0.$$

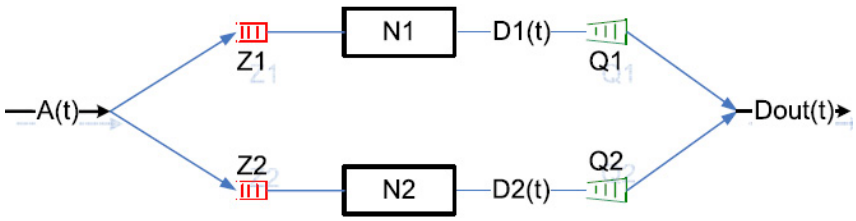
In Section 3 we shall use the same method to prove asymptotic optimality of the FCFS discipline in a more general class of networks.

2.5.4 Unsynchronized Policies in Fork-Join Networks

So far we described the problem of customers' synchronization in Fork-Join networks with *non-exchangeable* customers. The problem was then formulated as a stochastic control problem, and some Simple examples of networks and optimal policies were presented.
In the following example we shall address the question

Can a priority policy hurt system performance and how bad can it get?

We shall consider the following model



Assume that there are only two priorities, with high priority (H) tasks having preemptive priority over low priority (L)tasks. Additionally, we assume that all customer tasks have a unique ID number received upon arrival, but priority decision is local for each server and it is unsynchronized. By this we mean, for example, that server 1 assigns high priority to all tasks whose arrival ID is an **even** number, while server 2 ascribe high priority to all tasks whose arrival ID is an **odd** number.

Notations:

- $Z_i(t)$ – The number of customers on route i at time t, waiting in Buffer i or processing by server i;
- $Z_i^{H/L}(t)$ – The number of High / Low priority customers in the resource queue on route i at time t, respectively;

- $Q_i(t)$ – The number of customers in the synchronization queue preceding the join node at time t;
- $Q_i^{H/L}(t)$ – The number of all High / Low priority customers in the synchronization queue, respectively;
- $N(t)$ – Total number of customers in the system at time t;

One can verify the following relations for any control policy:

$$N(t) = Z_1(t) + Q_1(t) = Z_2(t) + Q_2(t);$$

$$\begin{cases} Q_1(t) \leq Z_2(t); \\ \\ Q_2(t) \leq Z_1(t). \end{cases} \tag{2.5}$$

Here, the first equation states that the number of customers is equal in both routes preceding the join node, and the two inequalities are derived from the definition of the synchronization queues: tasks waiting in the synchronization queues are tasks associated with a customer whose service was completed in one of the routes but is still incomplete in the other.
Using the *Collapse of High-Priority Processes* notion described in Reiman and Simon paper ([21]) for conventional Heavy Traffic, one can argue heuristically that $\hat{Q}_i^{H,(n)}(T) \Rightarrow 0$, for all i and any fixed T, meaning that the normalized queue length of high priority customers converge weakly to zero, as $n \longrightarrow \infty$. The intuition for that is clear since the high priority, under preemptive discipline, see a queue that is not entering heavy traffic ($\rho_H^n < 1$ for all n).
Let us denote by $\tilde{Z}_i^n(T)$ the set of all customer IDs at time T in route i. Then the previous result can be expressed as $\tilde{Z}_1^n(T) \cap \tilde{Z}_2^n(T) \longrightarrow_n \emptyset$. Therefore, for any fixed T,

$$\begin{cases} \hat{Q}_1^n(T) \Rightarrow \hat{Z}_2^n(T); \\ \\ \hat{Q}_2^n(T) \Rightarrow \hat{Z}_1^n(T); \end{cases} \tag{2.6}$$

$$\hat{N}^n(T) \Rightarrow \hat{Z}_1^n(T) + \hat{Z}_2^n(T);$$

One can verify by Eq (2.5) that $\hat{Z}_1^n(T) + \hat{Z}_2^n(T)$ is the maximum number of customers achievable in the system. It follows that we have achieved the worst performance possible.

Note - At this point we can point out the difference between optimal performance $Q_1(t) \wedge Q_2(t) \equiv 0$ or $N(t) = Z_1(t) \vee Z_2(t)$, to the worst performance $\tilde{Z}_1^n(T) \cap \tilde{Z}_2^n(T) \longrightarrow_n \emptyset$ or $\hat{N}^n(T) \Rightarrow \hat{Z}_1^n(T) + \hat{Z}_2^n(T)$.

3 Fork-Join Network with Multi-Servers

In this section we shall consider a generalized version of the problem introduced in Section 2.5.3. Under this multi-server settings, one can verify that *Exact Optimality* is unreachable. We thus consider the question

Is the FCFS discipline asymptotically optimal for the model in Figure 2.1?

We will show that the answer is **positive**.

3.1 Model Definition

Let a complete probability space $(\Omega, \mathcal{F}, \mathbb{P})$ be given supporting all random variables and stochastic processes defined below.
Let us consider the following network

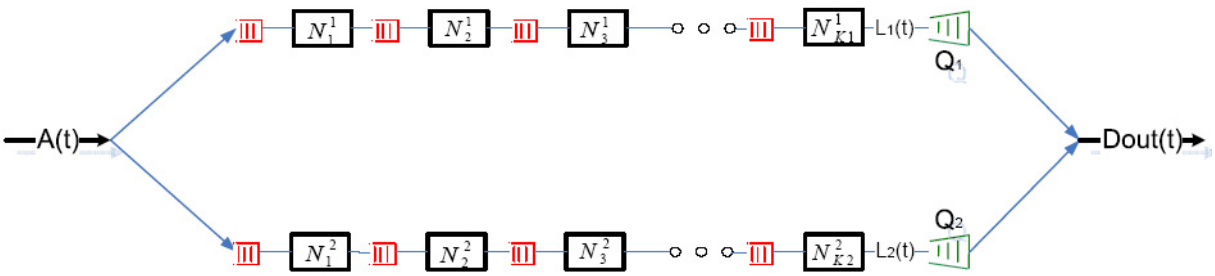


Figure 3.1: Multi-Server Simple Fork-Join Network

In this network a job arriving to the system "forks" to tasks processed simultaneously in the two parallel processing routes, each route contains multiple service stations with multiple servers in each station. In each route the task is performed sequentially according to the stations' order. The job departs the system only after the completion of the two tasks associated with it. All tasks completed in one of the routes wait in the *Synchronization buffers* (Fig 2.1, green buffers) until the corresponding task completes in the other route.
Notations

- $A(t)$ - Arrival process: number of customers arriving the system till time t;
- $D_{out}(t)$ - Departure process: number of departures from the system till time t;

- $Z_i^j(t)$ - Number of customers in the *resource buffer* preceding station j in route i on time t;
- $D_i^j(t)$ - Route i station j departure process: number of departures of complete tasks from route i station j till time t;
- $L_i(t)$ - Route departure process: number of departures of completed tasks from route i till time t;
- K_i - Number of service stations in route i;
- N_i^j - Number of servers in station j in route i;
- $Q_i(t)$ - Number of customers in the *synchronization buffer* in route i at time t;

Let $\tilde{Q}_i(t)$ be s set-valued process with values in the set of subsets of $\mathbb{N}$, representing the set of customers' index who are waiting in the *synchronization buffer* in route i at time t. Note that $|\tilde{Q}_i(t)| \equiv Q_i(t)$.
Let $V_i^j(t)$ be s set-valued process with values in the set of subsets of $\mathbb{N}$, representing the set of customers' index whose service have completed in station j (route i) till time t. The set is arranged according to the customers service completion times. Note that $|V_i^j(t)| \equiv D_i^j(t), \ \forall j \geq 1$.
Let $U_i^j(t)$ be s set-valued process with values in the set of subsets of $\mathbb{N}$, representing the set of customers' index whose service have initiated in station j (route i) till time t. the set is arranged according to the times in which the customers enter service.
Under the definition above, we define additional notations

- $V_i^0(t)$, the set of all customers' index who enter route i till time t, the set is arranged according to the arrival times. Note that $V_1^0(t) \equiv V_2^0(t)$, due to the common arriving process to the fork node.
- $V_i^{K_i}(t)$, the set of all customers' index whose service have completed in route i till time t. Note that $|V_i^{K_i}(t)| \equiv L_i(t)$;

System equations
The system primitive processes are the following

- $Z_i^1(0) = 0, \ \forall i, j$, assuming empty system on $t = 0$;

- $Q_i(0) = 0, \ \ \forall i \in \{1, 2\}$, assuming empty system on $t = 0$;
- $A(t)$, *External Arrival Process*: general distribution renewal process with average arrival rate of λ ;
- $S_i^j(t), \ \ \forall i, j$, *Potential service process.* Where $\{S_1^j(t), j \in (1, \ldots, K_1)\}$ and $\{S_2^j(t), j \in (1, \ldots, K_2)\}$ are assumed to be $K_1 + K_2$ mutually independent standard Poisson processes, independent of A(t).

Now we shall describe the system equations

$$\begin{cases} Z_i^1(t) = A(t) - D_i^1(t); & \forall i \in \{1, 2\} \\ Z_i^j(t) = D_i^{j-1}(t) - D_i^j(t); & \forall i \in \{1, 2\}, \ \ \forall j > 1 \\ L_i(t) = D_i^{K_i}(t); & \forall i \in \{1, 2\} \\ Q_i(t) = D_{out}(t) - L_i(t); & \\ D_i^j(t) = S_i^j(\mu_i^j B_i^j(t)); & \end{cases} \tag{3.1}$$

When $\mu_i^j, B_i^j(t)$ are the average service rate and busyness process associated with station j in route i.

We shall assume from now on that the system is in *Heavy Traffic* in the following sense **Heavy Traffic conditions** We shall use a construction of a sequence of systems, indexed by n. It is assumed that the following relations hold

- arrival rate: $\lambda^n = \lambda \cdot n + \hat{\lambda} \cdot \sqrt{n} + o(\sqrt{n})$.
- service rate: $\mu_j^n = \mu_j \cdot n + \hat{\mu}_j \cdot \sqrt{n} + o(\sqrt{n})$.
- $\lambda = N_j \cdot \mu_j$, for all stations j.

The following definition will be used in order to quantify the customers' disorder.

Definition 3. *Define a distance function for the position of a customer in two separate departure index vectors*

$$d^m(V_1^j(t), V_2^k(t)) = |d^m(V_1^j(t)) - d^m(V_2^k(t))| \cdot I\{m \in V_1^j(t) \cap V_2^k(t)\}; \tag{3.2}$$

here $d^m(V_1^j(t))$ ($d^m(V_2^k(t))$) is the position of the m'th customer in the departure index vector of station j in route 1 (station k route 2), respectively. From the following distance function, we can define a semi-metric for the distance between two index vectors

$$d(V_1^j(t), V_2^k(t)) = \sum_{m \in V_1^j(t) \cap V_2^j(t)} |d^m(V_1^j(t)) - d^m(V_2^k(t))|; \tag{3.3}$$

We will see that this function can measure disorder in the customers order between two vectors.

Control definition

We shall assume that tasks compete for resources at each station in a FCFS manner. At stations that do not involve a joining of tasks, this simply means that the tasks enter service in the order of their arrival to the station. Such a service discipline can be characterized as a local policy since it considers only station-level information. However we aim to show that this policy is asymptotically optimal in the conventional Heavy-Traffic.

3.2 Asymptotically Optimal Control

Following the condition for asymptotically optimal control defined in Section 2.3, for any fixed T

Theorem 3.1.

$$\begin{cases} \mathrm{P}(Q_1^n(T) \wedge Q_2^n(T) > K) \longrightarrow_n 0; \\ \\ \textit{When } K = [\sum_{j=1}^{K_1} (N_1^j - 1)] \vee [\sum_{j=1}^{K_2} (N_2^j - 1)]; \end{cases} \tag{3.4}$$

K is a deterministic number defined by the network structure, which is a generalized version of the result in Section 2.5.3.

We shall use three lemmas to prove the theorem, the lemmas' proofs can be found in Section 3.3.

Lemma 3.1. *For any fixed T and $\delta \in (0, 1/4)$*

$$\mathrm{P}(||V_1^{K_1}(T)| - |V_2^{K_2}(T)|| \leq n^{\frac{1}{2}-\delta}) \longrightarrow_n 0. \tag{3.5}$$

in Heavy Traffic.

Lemma 3.2. *Let us consider a single M/M/N station, we shall define $R(t)$ as the set of all customers' indexes whose service have completed until time t. For any fixed T and $\delta \in (0, 1/4)$, we claim that under the FCFS discipline one has*

$$\mathrm{P}(\max_{m \in R^n(T)} [d^m(V_A(T), V_D(T))] \geq n^\delta) \longrightarrow_n 0. \tag{3.6}$$

Here $V_A(T)$ denotes the index vector for the arrival process until time T, and $V_D(T)$ denotes the index vector for the departure process, respectively.

Lemma 3.3. *Triangle inequality for the distance function in tandem systems, for any customer m:*

$$d^m(V_1^{K_1}(t), V_2^{K_2}(t)) \leq d^m(V_1^{K_1-1}(t), V_2^{K_2-1}(t)) + d^m(V_1^{K_1}(t), V_1^{K_1-1}(t)) + d^m(V_2^{K_2}(t), V_2^{K_2-1}(t)).$$

Note that $d^m(V_1^0(t), V_2^0(t)) \equiv 0$, due to the common arriving process to the fork node, along with defining $d^m(V_1^0(t), V_1^{-1}(t)) = d^m(V_1^0(t), V_1^0(t)) \equiv 0$, $d^m(V_2^0(t), V_2^{-1}(t)) = d^m(V_2^0(t), V_2^0(t)) \equiv 0$. We may use induction to conclude the following inequality

$$d^m(V_1^{K_1}(t), V_2^{K_2}(t)) \leq \sum_{j=1}^{K_1} d^m(V_1^j(t), V_1^{j-1}(t)) + \sum_{j=1}^{K_2} d^m(V_2^j(t), V_2^{j-1}(t)), \tag{3.7}$$

where K_1 and K_2 denote the number of stations in each route, respectively.

Proof of Theorem 3.1.
Let $E_{n,T} = \{Q_1^n(T) \wedge Q_2^n(T) > K\}$.
Note that

$$\mathrm{P}(E_{n,T}) = \mathrm{P}(Q_1(T) \leq Q_2(T), Q_1(T) > K) + \mathrm{P}(Q_2(T) < Q_1(T), Q_2(T) > K).$$

Let us define the event on the first term of the r.h.s as E_n, we shall prove $\mathrm{P}(E_n) \longrightarrow_n 0$. One can see that the proof for the second term on the r.h.s is similar with opposite route indexes.
Recall that $U_i^j(t)$ is the set of customers' index whose service have initiated in station j (route i) till time t. Note that $|U_2^j(t)| = |V_2^j(t)| + dB_2^j(t)$ when $dB_2^j(t)$ is the number of busy servers in station j at time t ($B_2^j(t) = \int_0^t dB_2^j(t)$ is the busy time process).
Let us assume that there are more than K customers whose service was completed at route 1 but is still incomplete in route 2, which means more than K customers in $Q_1(T)$. We may divide the event into two cases

1. There exist a station $j \in \{1,\ldots,K_2\}$ in which all servers are busy (N_2^j servers) and all the customers in the station j belongs also to $Q_1(T)$. In this case, we shall consider m to be the last customer to enter service in station j. By the property of the FCFS priority discipline, we get

   $$\exists m \in Q_1(T),\ \exists j \in \{1,\ldots,K_2\} \text{ s.t } \ d^m(V_2^{j-1})(T) = |U_2^j(T)| = |V_2^j(T)| + N_2^j > |V_2^j(T)|.$$

2. There exist a station $j \in \{1,\ldots,K_2\}$ and a customer $m \in Q_1$ s.t customer m is waiting in the resource buffer preceding station j. By the property of the FCFS priority discipline, we get

   $$\exists m \in Q_1(T),\ \exists j \in \{1,\ldots,K_2\} \text{ s.t } \ d^m(V_2^{j-1})(T) > |U_2^j(T)| = |V_2^j(T)| + dB_2^j(t) > |V_2^j(T)|.$$

One see that $\mathrm{P}(E_n) \leq \mathrm{P}(\text{ case 1 }) + \mathrm{P}(\text{ case 2 })$, i.e., if there are more than K customers in $Q_1(T)$ at least one of the cases above must be true.
Let us define the event

$$H_n = \{\exists m \in Q_1(T),\ \exists j \in \{1,\ldots,K_2\} \text{ s.t } \ d^m(V_2^{j-1})(T) > |V_2^j(T)|\}.$$

Hence, $\mathrm{P}(E_n) \leq \mathrm{P}(H_n)$.
Notice that $|V_2^{K_2}(T)| \leq |V_2^{K_2-1}(T)| \leq \ldots \leq |V_2^j(T)|$, due to that the arrival process at any time T is always larger or equal to the departure process (assuming empty system at $t = 0$).
On the new event H_n, there exist a customer m and $\exists j \in [1,\ldots,K_2]$ s.t $\ d^m(V_2^{j-1})(T) > |V_2^j(T)| \geq |V_2^{K_2}(T)|$, but $m \in Q_1$ which means $d^m(V_1^{K_1})(T) \leq |V_1^{K_1}(T)|$ by definition. In addition $Q_1(T) \leq Q_2(T)$ which can be interpreted as $|V_2^{K_2}(T)| \geq |V_1^{K_1}(T)|$, due to the relation $|V_i^{K_i}(T)| = D_{\text{out}}(T) + Q_i(T)$. $D_{\text{out}}(T)$ is the system's departure process which include customers whose service was completed in both routes. In conclusion we get the following relation- $d^m(V_2^{j-1})(T) \geq |V_2^{K_2}(T)| \geq |V_1^{K_1}(T)| \geq d^m(V_1^{K_1})(T)$, which can be interpreted as

$$|d^m(V_2^{j-1})(T) - d^m(V_1^{K_1})(T)| \geq ||V_2^{K_2}(T)| - |V_1^{K_1}(T)||.$$

On the event $\{||V_2^{K_2}(T)| - |V_1^{K_1}(T)|| > n^{\frac{1}{2}-\delta}\}$, we get

$$d^m(V_2^{j-1}, V_1^{K_1})(T) = |d^m(V_2^{j-1})(T) - d^m(V_1^{K_1})(T)| \geq ||V_2^{K_2}(T)| - |V_1^{K_1}(T)|| > n^{\frac{1}{2}-\delta}.$$

Let us define the event

$$\tilde{H}_n = \{\exists m,\ \exists j \in \{1,\ldots,K_2\} \text{ s.t } \ d^m(V_2^{j-1}, V_1^{K_1})(T) > n^{\frac{1}{2}-\delta}\}.$$

Hence, by Lemma 3.1 we get $\mathrm{P}(H_n) \leq \mathrm{P}(\tilde{H}_n) + \alpha_n, \;\; \alpha_n \longrightarrow_n 0.$

By Lemma 3.3 we get

$$d^m(V_2^{j-1}, V_1^{K_1})(T) \leq \sum_{k=1}^{j-1} d^m(V_2^k(T), V_2^{k-1}(T)) + \sum_{k=1}^{K_1} d^m(V_1^k(T), V_1^{k-1}(T))$$

$$\leq \sum_{k=1}^{K_2} d^m(V_2^k(T), V_2^{k-1}(T)) + \sum_{k=1}^{K_1} d^m(V_1^k(T), V_1^{k-1}(T)) \tag{3.8}$$

When $\;\; d^m(V_2^k(T), V_2^{k-1}(T)) = 0 \;\; \forall k > j-1 \;\;$ by definition .

Therefore

$$\mathrm{P}(H_n) \leq_{\{\text{Lemma 1}\}} \mathrm{P}(\exists m, \; \exists j \in \{1, \ldots, K_2\} \text{ s.t } \;\; d^m(V_2^{j-1}, V_1^{K_1})(T) > n^{\frac{1}{2}-\delta}) + \alpha_n, \;\; \alpha_n \longrightarrow_n 0$$

$$\leq_{\{\text{Lemma 3}\}} \mathrm{P}(\exists m, \;\; \exists k \in [1, \ldots, K_2] \text{ s.t } \;\; d^m(V_2^k(T), V_2^{k-1}(T)) > \tfrac{n^{\frac{1}{2}-\delta}}{2 \cdot K_2})$$

$$+\mathrm{P}(\exists m, \;\; \exists k \in [1, \ldots, K_1] \text{ s.t } \;\; d^m(V_1^k(T), V_1^{k-1}(T)) > \tfrac{n^{\frac{1}{2}-\delta}}{2 \cdot K_1})$$

Recall that K_1, K_2 are deterministic numbers, and $\delta \in (0, 1/4)$

one can see that for any fixed $\delta' \in (0, 1/4)$

$$\exists N \;\; \text{s.t} \;\; \forall n > N \;\; \{n^{\delta'} \leq \tfrac{n^{\frac{1}{2}-\delta}}{2 \cdot K_1}\} \text{ and } \{n^{\delta'} \leq \tfrac{n^{\frac{1}{2}-\delta}}{2 \cdot K_2}\};$$

$$\leq \mathrm{P}(\exists m, \;\; \exists k \;\; \text{s.t} \;\; d^m(V_2^k(T), V_2^{k-1}(T)) > n^{\delta'})$$

$$+\mathrm{P}(\exists m, \;\; \exists k \;\; \text{s.t} \;\; d^m(V_1^k(T), V_1^{k-1}(T)) > n^{\delta'}) \tag{3.9}$$

Finally from Lemma 3.2 we conclude $\mathrm{P}(E_n) \leq \mathrm{P}(H_n) \longrightarrow_n 0$. That completes the proof of Theorem 3.1. □

Remark- notice that the customer index m analyzed here is a random variable taking value from the set of all served customers on route 1. That is why we use uniform convergence over all customers in the definition of Lemma 3.2.

Using the *diffusion scaling* $\hat{Q}_i^n(t) = \frac{Q_i^n(t)}{\sqrt{n}}$ with Claim 3.1 we get that for any fixed T

$$\hat{Q}_1^n(T) \wedge \hat{Q}_2^n(T) \leq \frac{K}{n^{\frac{1}{2}}} \;\; \text{w.p. converging to 1 .}$$

We may conclude that $\hat{Q}_1^n(T) \wedge \hat{Q}_2^n(T) \to 0$ in probability, which is asymptotically optimal under the definition above (2.3).
Note that the proof above also determines the rate of convergence, which is $(\sqrt{n})^{-1}$.

3.3 Proof of Lemmas

In this section we provides proofs for the three lemmas used in Section 3.2.
Fix T, and define $\delta \in (0, 1/4)$.

Proof of Lemma 3.1. Let us define the event $H_n = \{||V_1^{K_1}(T)| - |V_2^{K_2}(T)|| \leq n^{\frac{1}{2}-\delta}\}$. When $|V_1^{K_1}(t)|$ and $|V_2^{K_2}(t)|$ denotes the departure processes from route 1 and route 2, respectively (see Section 3.1).
Let us define the following sequences of standard Poisson processes $\{S^s_{K_1}(t), s \in (1, \ldots, N_1^{K_1})\}$, $\{S^s_{K_2}(t),\ s \in (1, \ldots, N_2^{K_2})\}$, and the associated busyness processes of the separate servers in the departure station- $\{B^s_{K_1}(t))\}$, $s \in (1, \ldots, N_1^{K_1})\}$, $\{B^s_{K_2}(t),\ s \in (1, \ldots, N_2^{K_2})\}$, where s refers to the server index. We assume that the $K_1 + K_2$ Poisson processes are mutually independent. In addition we shall define $S_1(t)$, $S_2(t)$ as two standard Poisson processes mutual independent to all other Poisson processes
We use the following notations

- $V_i^{K_i}(t)$, the set of all customers' index whose service have completed in route i till time t;
- $S^s_{K_i}(t)$- Standard Poisson process with rate 1, associated with the server s in the departure station of route i;
- N_i^j - Number of servers on station j in route i;

The relevant system equations for fixed T, are

$$\begin{cases} |V_1^{K_1}(T)| = \sum_{s=1}^{N_1^{K_1}} S_{K_1}^s(\mu_{K_1}^n \cdot B_{K_1}^s(T)) = S_1(\mu_{K_1}^n \cdot \sum_{s=1}^{N_1^{K_1}} B_{K_1}^s(T)) = S_1(\mu_{K_1}^n \cdot B_{K_1}(T)); \\ \\ |V_2^{K_2}(T)| = \sum_{s=1}^{N_2^{K_2}} S_{K_2}^s(\mu_{K_2}^n \cdot B_{K_2}^s(T)) = S_2(\mu_{K_2}^n \cdot \sum_{s=1}^{N_2^{K_2}} B_{K_2}^s(T)) = S_2(\mu_{K_2}^n \cdot B_{K_2}(T)); \end{cases}$$

$$\begin{cases} B_j^s(t) = \int_0^t \mathbb{I}_{\{\text{ server s in station j is busy at time u }\}} du, \quad j \in \{K_1, K_2\}; \\ \\ B_j(t) = \sum_{s=1}^{N_j} B_j^s(t), \quad j \in \{K_1, K_2\}; \end{cases} \tag{3.10}$$

Recall that the system approaches Heavy-Traffic in the sense (see Section 3.1)

- $\lambda^n = \lambda \cdot n + \hat{\lambda} \cdot \sqrt{n} + o(\sqrt{n})$.
- $\mu_j^n = \mu_j \cdot n + \hat{\mu}_j \cdot \sqrt{n} + o(\sqrt{n}), \quad j \in \{K_1, K_2\}$.
- $\lambda = N_j \cdot \mu_j$, for all stations j.

Therefore

$$\begin{cases} |\mu_{K_1} \cdot N_1^{K_1} - \mu_{K_2} \cdot N_2^{K_2}| = 0. \quad \text{Due to the common arrival process .} \\ \\ |B_{K_1}(T) - N_1^{K_1} \cdot T| \longrightarrow_n 0 \quad \text{for any fixed T. Since } \rho_1^{K_1,n} \longrightarrow_n 1 \text{ as } n \longrightarrow \infty. \\ \\ |B_{K_2}(T) - N_2^{K_2} \cdot T| \longrightarrow_n 0 \quad \text{for any fixed T. Since } \rho_2^{K_2,n} \longrightarrow_n 1 \text{ as } n \longrightarrow \infty. \end{cases}$$

Above $\mu_{K_1}^n, \mu_{K_2}^n$ are the average service rates for the servers in the departure station of route 1 and 2, respectively.

Now let us define the event

$$\tilde{H}_n = \left\{ \begin{array}{l} |S_1(\mu_{K_1}^n \cdot N_1^{K_1} T) - S_2(\mu_{K_2}^n \cdot N_2^{K_2} T)| \leq n^{\frac{1}{2}-\delta}, \\ \\ \mu_{K_1}^n \cdot N_1^{K_1} T - \mu_{K_2}^n \cdot N_2^{K_2} T = (\hat{\mu}_{K_1} \cdot N_1^{K_1} - \hat{\mu}_{K_2}^n \cdot N_2^{K_2}) \cdot \sqrt{n} \cdot T \end{array} \right\}$$

Then $\mathrm{P}(H_n) \leq \mathrm{P}(\tilde{H}_n) + \alpha_n$, when $\alpha_n \longrightarrow_n 0$.
On the new event $\tilde{H}_n$,

We shall "scale" $|S_1(\mu^n_{K_1}\cdot N_1^{K_1}T)-S_2(\mu^n_{K_2}\cdot N_2^{K_2}T)|\leq n^{\frac{1}{2}-\delta}$ by $n^{-\frac{1}{2}}$ and center the random variables to get

$$|\frac{S_1(\mu^n_{K_1}\cdot N_1^{K_1}T)-\mu^n_{K_1}\cdot N_1^{K_1}T}{n^{\frac{1}{2}}}-\frac{S_2(\mu^n_{K_2}\cdot N_2^{K_2}T)-\mu^n_{K_2}\cdot N_2^{K_2}T}{n^{\frac{1}{2}}}+\frac{(\mu^n_{K_1}\cdot N_1^{K_1}-\mu^n_{K_2}\cdot N_2^{K_2})\cdot T}{n^{\frac{1}{2}}}|\leq n^{-\delta}.$$

Using the diffusion scaling $\hat{S}^n_k(t)=\frac{S_k(\mu^n_k t)-\mu^n_k t}{\sqrt{n}}$, and from the event $\tilde{H}_n$ we get the relation $\hat{\mu}_{K_1}\cdot N_1^{K_1}-\hat{\mu}^n_{K_2}\cdot N_2^{K_2}=\frac{\mu^n_{K_1}\cdot N_1^{K_1}-\mu^n_{K_2}\cdot N_2^{K_2}}{\sqrt{n}}$.

Therefore

$$|\hat{S}^n_{K_1}(T)-\hat{S}^n_{K_2}(T)+(\hat{\mu}_{K_1}\cdot N_1^{K_1}-\hat{\mu}^n_{K_2}\cdot N_2^{K_2})\cdot T|\leq n^{-\delta}.$$

When $\hat{S}^n_{K_1}(T)$ and $\hat{S}^n_{K_2}(T)$ are scaled Poisson random variables which converge weakly to Normal random variables, by the central limit theorem.
Let us define - $\hat{w}^n(T)=\hat{S}^n_{K_1}(T)-\hat{S}^n_{K_2}(T)+(\hat{\mu}_{K_1}\cdot N_1^{K_1}-\hat{\mu}^n_{K_2}\cdot N_2^{K_2})\cdot T$, then $\hat{w}^n(T)$ converge weakly to Normal random variable. Therefore $\mathrm{P}(\tilde{H}_n)=\mathrm{P}(|\hat{w}^n(T)|\leq n^{-\delta})$.
However, since $\hat{w}^n(T)$ converge to a continuous R.V., the probability measure to be bounded by $n^{-\delta}$ tends to zero with n. Thus $\mathrm{P}(\tilde{H}_n)=\mathrm{P}(|\hat{w}^n(T)|\leq n^{-\delta})\longrightarrow_n 0$.
We conclude that $\mathrm{P}(H_n)\longrightarrow_n 0$, which completes the proof of Lemma 3.1. □

Proof of Lemma 3.2. Let us consider a single M/M/N station in which customers compete for resources according to FCFS discipline. Define $R(t)$ as the set of all customers' indexes whose service have completed until time t, and define the event $H^n_m=\{d^m(V_A(T),V_D(T))\geq n^\delta\}$ for a specific $m\in R^n(T)$ and fixed T.
One may see that,

$$\mathrm{P}(\max_{m\in R^n(T)}[d^m(V^n_A(T),V^n_D(T))]\geq n^\delta)=\mathrm{P}(\cup_{m\in R^n(T)}H^n_m)\leq\sum_{m\in R^n(T)}\mathrm{P}(H^n_m).$$

Define $\{S^j(t),\ j\in(1,\ldots,N)\}$ and $S(t)$ to be $N+1$ mutually independent standard Poisson processes. The system equations are

$$
\begin{cases}
|R^n(T)| = \sum_{j=1}^{N} S^j(\mu_j^n \cdot B^j(T)) = S(\sum_{j=1}^{N} \mu_j^n \cdot B^j(T)) = \\
\qquad = S\Big(\mu \cdot n \cdot B(T) + \sum_{j=1}^{N} [(\hat{\mu_j} \cdot \sqrt{n} + o(\sqrt{n})) \cdot B^j(T)]\Big); \\
B^j(t) = \int_0^t \mathbb{I}_{\{\text{ server j is busy at time s }\}} ds; \\
B(t) = \sum_{j=1}^{N} B^j(t); \\
\mu_j^n = \mu \cdot n + \hat{\mu_j} \cdot \sqrt{n} + o(\sqrt{n}), \quad \forall j \in \{1, \ldots, N\}, \\
\qquad \text{i.e., assuming identical servers' avarage rates ;}
\end{cases}
$$

Step 1- Upper bound for $|R^n(T)|$.

Claim 3.1. *For fixed* $\delta \in (0, 1/4)$, $\mathrm{P}(|R^n(T)| > n^{1+\delta}) \longrightarrow_n 0$.

Proof of Claim 3.1.
$B(t)$ is a non-decreasing function from $[0, T]$ to $[0, T]$ which is Lipschitz continuous and $0 \leq \frac{B(T)}{NT} \leq 1$. Therefore

$$\mathrm{P}(|R^n(T)| > n^{1+\delta}) \leq \mathrm{P}(|S(\mu \cdot n \cdot NT)| > n^{1+\delta}) + \alpha_n, \;\; \alpha_n \longrightarrow_n 0.$$

By Markov inequality we get-
$\mathrm{P}(|S(\mu \cdot n \cdot NT)| > n^{1+\delta}) \leq \frac{\mu \cdot n \cdot NT}{n^{1+\delta}} = \frac{\mu \cdot NT}{n^{\delta}} \longrightarrow_n 0.$
That completes the proof of Claim 3.1. □

Therefore

$$\mathrm{P}(\max_{m \in R^n(T)} [d^m(V_A^n(T), V_D^n(T))] \geq n^{\delta}) \leq \sum_{m \in R^n(T)} \mathrm{P}(H_m^n) \leq n^{1+\delta} \cdot \mathrm{P}(H_m^n) + \beta_n, \;\; \beta_n \longrightarrow_n 0.$$

Hence it is enough to prove that $\mathrm{P}(H_m^n)$ converge to zero with exponential rate.

Under the FCFS discipline the event H_m^n occurs if and only if while customer m is served by a server j, the other $N-1$ servers complete the service of another n^{δ} customers. Which means that in this event there exist a server j who serves 1 customer and server k who serves at least $\frac{n^{\delta}}{N-1}$ at the same time period.

Let

$\tau_m = \{t \ : \ \text{customer m enters service in server j }\}$;

$\sigma_m = \{t \ : \ \text{customer m departures from server j }\}$;

Note that on H_m^n there exist j s.t. $0 \leq \tau_m \leq \sigma_m \leq T$, and τ_m, σ_m are stoping times adapted to the natural filtration containing all customers' arrivals and departures.

Step 2- Upper bound for $|\sigma_m - \tau_m|$.

Let us define the event,

$E_n = \{\exists j,\ \tau_m, \sigma_m \ \text{as defined} \ , \ |\sigma_m - \tau_m| > n^{-1+\delta}\}$.

Claim 3.2. $\mathrm{P}(E_n) \longrightarrow_n 0$.

Proof of Claim 3.2.

τ_m, σ_m are stoping times, thus

$$\triangle S_j[\tau_m, \sigma_m] = S_j(\mu_j^n \cdot \triangle B^j[\tau_m, \sigma_m]) = S_j(\mu_j^n \cdot |\sigma_m - \tau_m|).$$

The last equality is derived from the definition of $[\tau_m, \sigma_m]$, during this period of time server j is busy serving customer m. Therefore

$$\mathrm{P}(E_n) = \mathrm{P}(S_j(\mu \cdot n^\delta) = 1).$$

Using a *large deviation* technique with fixed $u = 1$, we get

$$\mathrm{P}(S_j(\mu \cdot n^\delta) = 1) = \mathrm{P}(e^{S_j(\mu \cdot n^\delta)} = e^1) \overset{\text{Markov}}{\leq} \frac{E(e^{S_j(\mu \cdot n^\delta)})}{e^1} = (*);$$

Now using the Poisson moment generating function,

$$E(e^{S_j(\mu \cdot n^\delta)}) = e^{-(\mu \cdot n^\delta)\cdot(1-e)} \propto e^{-n^\delta}.$$

Hence

$$(*) \propto e^{-n^\delta - 1} \propto e^{-n^\delta}.$$

From above we conclude that $\mathrm{P}(E_n) \longrightarrow_n 0$, which completes the proof of Claim 3.2. □

Step 3- We prove that $\mathrm{P}(H_m^n)$ converges to zero with exponential rate.

We calculate the probability of the event, in which there exists a server j who serves 1 customer and server k who serves at least $\frac{n^\delta}{N-1}$ at the same time period. Define $[\tau_m, \sigma_m]$ as before, let us define the event -

$$\tilde{H}^n = \{\exists j, k \ \text{ s.t } \ \triangle(S_k - S_j)[\tau_m, \sigma_m] \geq \frac{n^\delta}{N-1} - 1 \geq c \cdot n^\delta, \ \ |\sigma_m - \tau_m| \leq n^{-1+\delta}\}.$$

One can see that $\mathrm{P}(H_m^n) \leq \mathrm{P}(\tilde{H}^n) + \beta_n$, when $\beta_n \propto e^{-n^\delta}$ converge to zero with exponential rate [2].

Claim 3.3. $\mathrm{P}(\tilde{H}^n) \longrightarrow_n 0$ *with exponential rate.*

Proof of Claim 3.3.
$\triangle B^j[\tau_m, \sigma_m] = [\tau_m, \sigma_m]$ follows from the definition of $[\tau_m, \sigma_m]$, i.e., during this period of time server j is busy serving customer m. In addition, $\triangle B^k[\tau_m, \sigma_m] \leq [\tau_m, \sigma_m]$. Now on the event $\tilde{H}^n$ we get [3]

$$\mathrm{P}(\triangle(S_k - S_j)[\tau_m, \sigma_m] \geq c \cdot n^\delta) \leq \mathrm{P}(S[|\mu_k^n - \mu_j^n| \cdot |\sigma_m - \tau_m|] \geq c \cdot n^\delta)$$

$$\leq \mathrm{P}(S[(\hat{\mu}_j - \hat{\mu}_k) \cdot n^{-\frac{1}{2}+\delta}] \geq c \cdot n^\delta) + \beta_n, \quad \beta_n \longrightarrow_n 0;$$

When $S(t)$ is standard Poisson processes with rate 1.

Using a *large deviation* technique with fixed $u = 1$, we get

$$\mathrm{P}(S[|\hat{\mu}_j - \hat{\mu}_k| \cdot n^{-\frac{1}{2}+\delta}] \geq c \cdot n^\delta) = \mathrm{P}(e^{S[|\hat{\mu}_j - \hat{\mu}_k| \cdot n^{-\frac{1}{2}+\delta}]} > e^{c \cdot n^\delta}) \overset{\text{Markov}}{\leq}$$

$$\overset{\text{Markov}}{\leq} \frac{E(e^{S(|\hat{\mu}_j - \hat{\mu}_k| \cdot n^{-\frac{1}{2}+\delta})})}{e^{c \cdot n^\delta}} = (*);$$

Now using the Poisson moment generating function,

$$E(e^{S(|\hat{\mu}_j - \hat{\mu}_k| \cdot n^{-\frac{1}{2}+\delta})}) = e^{-|\hat{\mu}_j - \hat{\mu}_k| \cdot n^{-\frac{1}{2}+\delta} \cdot (1-e)} = e^{-\tilde{c} \cdot n^{-\frac{1}{2}+\delta}}.$$

Hence

$$(*) = e^{-\tilde{c} \cdot n^{\delta - \frac{1}{2}} - c \cdot n^\delta} = e^{-c \cdot n^\delta \cdot (\frac{\tilde{c}}{c} n^{-\frac{1}{2}} + 1)} = e^{-c \cdot n^\delta} \cdot e^{(\frac{\tilde{c}}{c} n^{-\frac{1}{2}} + 1)} \propto e^{-n^\delta}.$$

Therefore $\mathrm{P}(\tilde{H}^n) \longrightarrow_n 0$ with exponential rate, which completes the proof of Claim 3.3. □

In conclusion we get

$$\mathrm{P}(H_m^n) \leq \mathrm{P}(\tilde{H}^n) + \beta_n \leq e^{-n^\delta} + \beta_n, \quad \beta_n \propto e^{-n^\delta}.$$

Which means that $\mathrm{P}(H_m^n)$ converge to zero with exponential rate, or

$$\mathrm{P}(\max_{m \in R^n(T)} [d^m(V_A^n(T), V_D^n(T))] \geq n^\delta) \leq n^{1+\delta} \cdot \mathrm{P}(H_m^n) \leq n^{1+\delta} \cdot e^{-n^\delta} \longrightarrow_n 0.$$

This completes the proof of Lemma 3.2. □

[2] according to step 2

[3] τ_m, σ_m are stopping times

Proof of Lemma 3.3.(Triangle inequality for the distance function in tandem systems) Let us define $V_i^j(t)$ as the vector of all customers' indexes whose service have completed in station j (route i) till time t, the index vector is arranged according to the service completion times (see Section 3.1). We shall prove the claim for any customer m
$d^m(V_1^{K_1}(t), V_2^{K_2}(t)) \leq d^m(V_1^{K_1-1}(t), V_2^{K_2-2}(t)) + d^m(V_1^{K_1}(t), V_1^{K_1-1}(t)) + d^m(V_2^{K_2}(t), V_2^{K_2-1}(t))$.
The rest can be checked by using simple induction.
By the definition in Section 3.1 we set $d^m(V_1^{K_1}(t), V_2^{K_2}(t))$ to be

$$d^m(V_1^{K_1}(t), V_2^{K_2}(t)) = |d^m(V_1^{K_1}(t)) - d^m(V_2^{K_2}(t))| \cdot I\{m \in V_1^{K_1}(t) \cap V_2^{K_2}(t)\}.$$

Now let us check the validity of the inequality,

- if $m \in V_1^{K_1}(t) \cap V_2^{K_2}(t)$: surely $m \in V_1^{K_1-1}(t)$ and $m \in V_2^{K_2-1}(t)$ because these are the arrival processes for $V_1^{K_1}(t)$ and $V_2^{K_2}(t)$, respectively. Then $d^m(V_1^{K_1-1}(t))$ and $d^m(V_2^{K_2-1}(t))$ are well defined. Now
$d^m(V_1^{K_1}(t), V_2^{K_2}(t)) = |d^m(V_1^{K_1}(t)) - d^m(V_2^{K_2}(t))|$

$$= |(d^m(V_1^{K_1}(t)) - d^m(V_1^{K_1-1}(t))) + (d^m(V_2^{K_2-1}(t)) - d^m(V_2^{K_2}(t)))$$

$$+(d^m(V_1^{K_1-1}(t)) - d^m(V_2^{K_2-1}(t)))|$$

$$\leq |d^m(V_1^{K_1}(t)) - d^m(V_1^{K_1-1}(t))| + |d^m(V_2^{K_2}(t)) - d^m(V_2^{K_2-1}(t))|$$

$$+|d^m(V_1^{K_1-1}(t)) - d^m(V_2^{K_2-1}(t))|$$

- if not: $d^m(V_1^{K_1}(t), V_2^{K_2}(t)) \equiv 0$, which means that the inequality is valid. Due to the fact that all the distance functions on the right side of the inequality are positive.

This completes the proof of lemma 3.3. □

Although we do not use this in this work, note that

$$d(V_1^n(t), V_2^k(t)) = \sum_{i \in V_1^n(t) \cap V_2^n(t)} |d^i(V_1^n(t)) - d^i(V_2^k(t))|,$$

is a semi-metric in the space of the index subsets $(\mathbb{N}^d)$.

- $d(V_1^n(t), V_2^k(t))$ is symmetrical by definition.

- $d(V_1^n(t), V_2^k(t)) = 0$ if $\{V_1^n(t) \equiv V_2^k(t)\}$ Or $\{V_1^n(t) \bigcap V_2^k(t) = \emptyset\}$.
- The triangle inequality is satisfied (by the proof above).

3.4 Comments About Generalization

The model presented in this section consist of a single join node at the departure end of the system. This model can be extended to a general multi-server feedforward fork-join network with multiple fork and join constructs, such as in Cohen, Mandelbaum and Shtub [1].

Note that the proof for the *Asymptotic Optimality Theorem* (3.1) relies on the property that customers disorder is bounded by the order of n^δ throughout the system. Now using the following intuition, this property can be extended to general fork-join system.

Define $d(j)$ to be the "degree" of the join node j, meaning that $d(j)$ is the number of predecessor join nodes preceding join node j. Starting with the join nodes J such that $d(j) = 0$ for each $j \in J$, each route preceding a join node $j \in J$ is a finite sum of stations with bounded disorder. Hence, focusing on a join node $j \in J$, by using the *Triangle Inequality* (3.3), we get that customers disorder is bounded by the order of n^δ. Also, the customers disorder in the departure set from the join node $j \in J$ is bounded by the maximum disorders in all the preceding routes, which is bounded by the order of n^δ. In an inductive manner, this bound can be extended to all join nodes in the network. Consider a join node j ($d(j) \geq 1$) such that all immediate predecessor join nodes have been "treated", meaning that their disorder bound has been established. Then each route preceding the join node j is a finite sum of stations and join nodes with bounded disorder. Therefore, the customers disorder in the departure set from the join node j ($d(j) \geq 1$) is also bounded by the order of n^δ.

Hence, we expect (but did not prove) that Theorem 3.1 may be generalized to general multi-server feedforward fork-join networks, such as in [1], meaning that FCFS policy is *asymptotically optimal* for this general setting, contradicting the observations made in [1].

4 Fork-Join Network with Feedback

In this section we shall consider a broader class of models which allow probabilistic feedback. This setting is a generalized version of the problem introduced in Section 2.5.2. Under this settings one may see that is *Exact Optimality* is unreachable for any definition of FCFS. In fact, this seems to be the simplest setting of a Fork-Join network where solving for optimal scheduling is hard. We shall propose a new control policy and prove asymptotic optimality for the specific network introduced here.

4.1 Model Definition

Let a complete probability space $(\Omega, \mathcal{F}, \mathbb{P})$ be given supporting all random variables and stochastic processes defined below.
Let us consider the following network

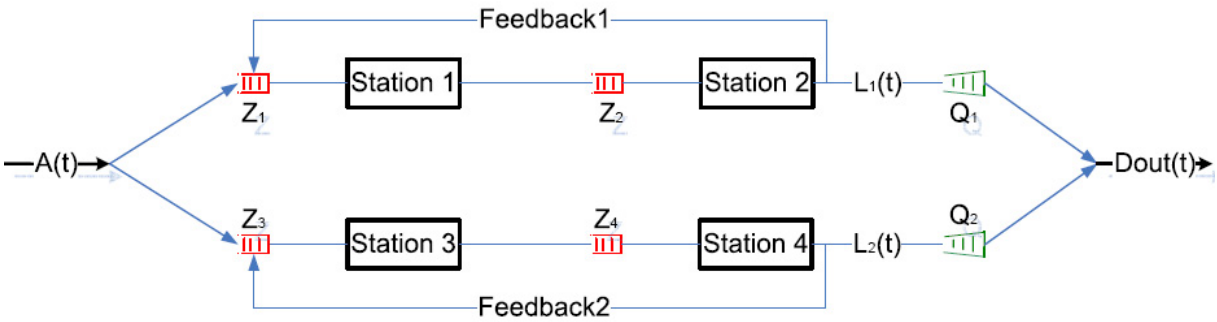


Figure 4.1: Double Station Simple Fork-Join with Feedback

In this Network a job arriving to the system "fork" to tasks processed simultaneously in the two parallel processing routes, each routes contains two service stations and a probabilistic feedback. The feedback may be viewed has a has a *quality check* at the end of the processing line, in which unsatisfying tasks are sent back to be processed again.
Notations

- $A(t)$ - Arrival process: number of customers arriving to the system till time t;
- $D_{out}(t)$ - Departure process: number of departures of complete customers till time t;

- $L_i(t)$ - Route departure process: number of departures of complete tasks from route i till time t;
- $D_j(t)$ - Station departure process: number of departures of complete tasks from station j till time t;
- Z_j - Number of customers in the *resource buffer* preceding station j on time t;
- $Q_i(t)$ - Number of customers in the synchronization buffer in route i;

System equations

The system primitive processes are the following

- $Z_j(0) = 0 \ \ \forall j \in \{1, \ldots, 4\}$, Initial condition, assuming empty system on $t = 0$;
- $Q_i(0) = 0 \ \ \forall i \in \{1, 2\}$, Initial condition, assuming empty system on $t = 0$;
- $A(t)$ System's *External Arrival Process*, general distribution renewal process with average arrival rate of λ ;
- $S_j(t) \ \ \forall j \in \{1, \ldots, 4\}$, System's *Potential service process*. Where $\{S_j(t),\ j \in (1, \ldots, 4)\}$ are assumed to be mutually independent standard Poisson processes.
- $\xi_k^i \ \ \forall i \in \{1, 2\},\ k \in \mathbb{N}=$ defines a sequence of i.i.d random variables with Bernoulli-distribution (taking values 0/1), which denote the feedback decision process. Let us define the probability of feedback on route i to be $1 - p_i$;

The primitive processes are assumed to be mutually independent.

Now we shall describe the System's equation

$$\begin{cases} Z_1(t) = A(t) - D_1(t) + R_1(t); \\ Z_2(t) = D_1(t) - D_2(t); \\ Z_3(t) = A(t) - D_3(t) + R_2(t); \\ Z_4(t) = D_3(t) - D_4(t); \\ L_1(t) = D_2(t) - R_1(t); \\ L_2(t) = D_4(t) - R_2(t); \\ B_j(t) = \int_0^t \mathbb{I}_{\{Z_j(s)>0\}} ds; \\ I_j(t) = t - B_j(t) = \int_0^t \mathbb{I}_{\{Z_j(s)=0\}} ds; \\ D_j(t) = S_j(\mu_j B_j(t)); \\ R_1(t) = \sum_{k=1}^{D_2(t)} \xi_k^1; \\ R_2(t) = \sum_{k=1}^{D_4(t)} \xi_k^2; \end{cases} \tag{4.1}$$

Where μ_j is the average service rate associated with station j, and $B_j(t), I_j(t)$ are the station's busyness, Idleness processes,respectively.

We shall assume from now on that the system is in *Heavy Traffic* in the following sense
Heavy Traffic definition
Let us define the *Heavy Traffic* condition for the upper route (for the lower route the

result is the same). According to ([3])

$$\lambda = \alpha + \mathrm{P}'(\lambda \wedge \mu)$$

$$\text{Thus } \begin{pmatrix} \lambda_1 \\ \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} \alpha \\ \\ 0 \end{pmatrix} + \begin{pmatrix} 0 & p \\ \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} \lambda_1 \wedge \mu_1 \\ \\ \lambda_2 \wedge \mu_2 \end{pmatrix}$$

$$\text{Thus } \begin{cases} \lambda_1 = \alpha + p \cdot (\lambda_2 \wedge \mu_2); \\ \\ \lambda_2 = \lambda_1 \wedge \mu_1; \end{cases} \tag{4.2}$$

Let us assume *Heavy Traffic* on both servers, meaning $\lambda_1 = \mu_1, \ \lambda_2 = \mu_2;$

$$\text{Thus } \textit{Heavy Traffic} \text{ condition - } \begin{cases} \mu_1 = \mu_2 = \mu; \\ \\ \alpha = \mu \cdot (1 - p); \end{cases}$$

When α denote the average external arrival rate to server 1, p denote the feedback probability, and λ_i denote the average arrival rate to each server, respectively. Note that we relay on the Lemma from [3] that the equation solution exist and unique.
The precise formulation of our *Heavy Traffic* limit theorem requires the construction of a "sequence of systems", indexed by n. It is assumed that the following relations hold-

- $\alpha^n = \alpha \cdot n + \hat{\alpha} \cdot \sqrt{n} + o(\sqrt{n})$.
- $\mu_j^n = \mu_j \cdot n + \hat{\mu}_j \cdot \sqrt{n} + o(\sqrt{n})$.
- *Heavy Traffic Condition*:

$$\begin{cases} \mu_1 = \mu_2; \\ \\ \mu_3 = \mu_4; \\ \\ \alpha = \mu_1 \cdot (1 - p_1) = \mu_3 \cdot (1 - p_2); \end{cases}$$

4.2 Proposed Control

As noted before FCFS is no longer even asymptotically optimal in this setting. Therefore we shall define a new control policy which is assumed to be asymptotically optimal, which

we will prove later on. Let us assume that there are only 2 priorities, with high priority (H-P) tasks having preemptive priority over low priority (L-P) tasks, i.e., a service to a low priority customer can be interrupted and resumed at a later time. Let us define the **Control policy** as follows: At each route, assign preemptive priority to customers whose service was completed in the other route. Preemptive priority, i.e., a service to a customer can be interrupted and resumed at a later time.

Note that this policy requires information flow between servers and central coordination (global control). We shall assume immediate awareness of each station to the status in all other stations.

One may see that the definition of the policy creates an artificial division of the customers into two classes:

- LP (Low Priority) Customers: Customers whose service is still incomplete in both routes.
- HP (High Priority) Customers: Customers whose service was completed in one of the routes but is still incomplete in the other.

Define FCFS priority policy within each class, the policy is fully defined.

Let us define the following notation- Consider a generic process $G_j(t)$, we shall refer to

- $G_j^T(t)$ as the process associated with the total amount of customers in station j on time t, hence high priority customers and low priority customers together;
- $G_j^H(t)$ as the process associated with the amount of high priority customers in station j on time t;
- $G_j^L(t)$ as the process associated with the amount of low priority customers in station j on time t;

Using the above notations we get

$$\begin{cases} Z_j^T(t) = Z_j^H(t) + Z_j^L(t); \\ \\ D_j^T(t) = D_j^H(t) + D_j^L(t); \\ \\ L_i^T(t) = L_i^H(t) + L_i^L(t); \end{cases}$$

Note that the generation of high priority (H-P) customer in route 1 is caused by a departure of low priority (L-P) customer from route 2, and the opposite around. Meaning that $L_2^L(t) = D_4^L(t) - \sum_{k=1}^{D_4^L(t)} \xi_k^{2,L} = A_1^H(t) + A_2^H(t)$, when $A_j^H(t)$ denote the generation ("arrival") process of high-priority customers to station j. Under this context $\xi_k^{2,L}$ means the appropriate sub-sequence of ξ_k^2, which is used to draw Low-Priority feedbacks.
In the same way, we may define $\xi_k^{i,L}$ and $\xi_k^{i,H}$ as the appropriate sub-sequence of ξ_k^i, which is used to draw Low-Priority and High-Priority feedbacks, respectively. One may see that each is a sequence of i.i.d Bernoulli random variables with the probability $1 - p_i$;

Note that the *Heavy Traffic* definition above is referred to the total customers amount process. The high priority and low priority behavior is not yet defined under the system's *Heavy Traffic*.

4.3 Asymptotically Optimal Control

Following the condition for asymptotically optimal control defined in Section 2.3, we aim to prove that the proposed policy is asymptotically optimal.
Fix T and $\epsilon > 0$,

Theorem 4.1.

$$\mathrm{P}(max_{t\in[0,T]}\{\hat{Z}_{1,2}^{n,H}(t) \wedge \hat{Z}_{3,4}^{n,H}(t)\} > \epsilon) \longrightarrow_n 0;$$

$$When \begin{cases} \hat{Z}_{1,2}^{n,H}(t) = \hat{Z}_1^{n,H}(t) + \hat{Z}_2^{n,H}(t); \\ \\ \hat{Z}_{3,4}^{n,H}(t) = \hat{Z}_3^{n,H}(t) + \hat{Z}_4^{n,H}(t); \end{cases} \tag{4.3}$$

Proof of Theorem 4.1. Let $E_{n,T} = \{max_{t\in[0,T]}\{\hat{Z}_{1,2}^{n,H}(t) \wedge \hat{Z}_{3,4}^{n,H}(t)\} > \epsilon\}$.
Let

$$\sigma = inf\{t \ : \ \hat{Z}_{1,2}^{n,H}(t) \wedge \hat{Z}_{3,4}^{n,H}(t) > \epsilon\};$$

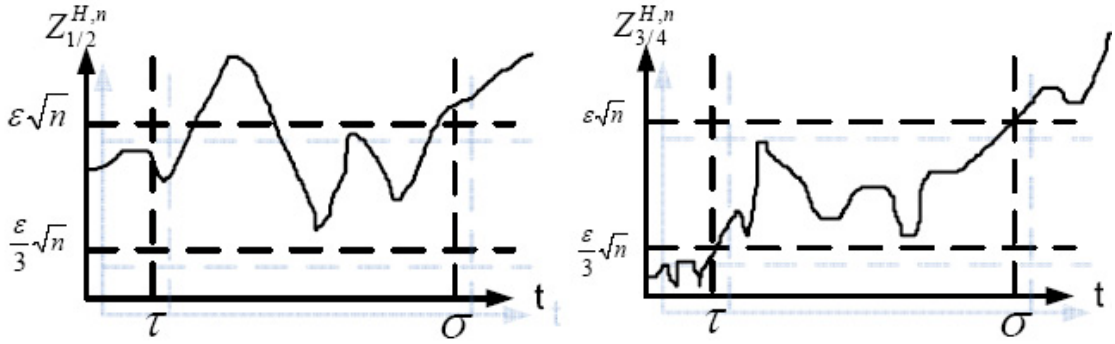
$$\tau = sup\{t < \sigma \ : \ \hat{Z}_{1,2}^{n,H}(t) \wedge \hat{Z}_{3,4}^{n,H}(t) \leq \tfrac{\epsilon}{3}\};$$

Note that on $E_{n,T}$ one has $0 \leq \tau \leq \sigma \leq T$.
Define

$$E_n = E_{n,T} \cap \{\hat{Z}_{3,4}^{n,H}(\tau) \leq \hat{Z}_{1,2}^{n,H}(\tau)\}$$

Then it is sufficient to prove $\mathrm{P}(E_n) \longrightarrow_n 0$.

On the event E_n



Note that on the event E_n one has

- $\hat{Z}^{n,H}_{1,2}(s) \geq \frac{\epsilon}{3} \ \ \forall s \in [\tau, \sigma]$;
- $\hat{Z}^{n,H}_{3,4}(s) \geq \frac{\epsilon}{3} \ \ \forall s \in [\tau, \sigma]$;
- $\hat{Z}^{n,H}_{3,4}(\sigma) - \hat{Z}^{n,H}_{3,4}(\tau) \geq \frac{\epsilon}{2}$;

We shall use three lemmas to prove that $\mathrm{P}(E_n) \longrightarrow_n 0$, the lemmas' proofs can be seen in Section 4.4.

Lemma 4.1. *Using the defined interval* $[\tau, \sigma]$ *and fixed* $\delta \in (0, 1/4)$

$$\begin{aligned} \mathrm{P}(I^{n,H}_2[\tau, \sigma] > n^{-\frac{1}{2}+\delta}) &\longrightarrow_n 0; \\ \mathrm{P}(I^{n,H}_4[\tau, \sigma] > n^{-\frac{1}{2}+\delta}) &\longrightarrow_n 0; \end{aligned} \tag{4.4}$$

Upper bound on the cumulative Idleness *time for H-P customers on interval* $[\tau, \sigma]$.

Lemma 4.2. *Using the defined interval* $[\tau, \sigma]$ *and fixed* $\delta \in (0, 1/4)$

$$\mathrm{P}(D^{n,L}_2(\sigma) - D^{n,L}_2(\tau) > n^{\frac{1}{2}+\delta}) \longrightarrow_n 0; \tag{4.5}$$

Upper bound on the cumulative amount of L-P customers' departure from route 1 on interval $[\tau, \sigma]$.

Lemma 4.3. *Using the defined interval* $[\tau, \sigma]$ *and fixed* $\delta \in (0, 1/4)$

$$\mathrm{P}(|\sigma - \tau| < n^{-\delta}, \ \ A^{n,H}_{3,4}[\tau, \sigma] \geq \frac{\epsilon}{2} \cdot \sqrt{n}) \longrightarrow_n 0; \tag{4.6}$$

Lower bound on the length of time of interval $[\tau, \sigma]$.

Note that $A^{n,H}_{3,4}[\tau,\sigma] \leq D^{n,L}_2(\sigma) - D^{n,L}_2(\tau)$, since the generation of high priority (H-P) customers in route 2 is caused by a departure of low priority (L-P) customer from route 1. Meaning that $A^{n,H}_{3,4}(t) = D^{n,L}_2(t) - \sum_{k=1}^{D^{n,L}_2(t)} \xi^{1,L}_k$.
Let us define the event

$$\tilde{E}_n = \{A^{n,H}_{3,4}[\tau,\sigma] \leq n^{\frac{1}{2}+\delta},\ \ \hat{Z}^{n,H}_{1,2}(s) \geq \frac{\epsilon}{2}\ \forall s \in [\tau,\sigma],\ \ I^{n,H}_4[\tau,\sigma] \leq n^{-\frac{1}{2}+\delta},\ \ \hat{Z}^{n,H}_{3,4}(\sigma) - \hat{Z}^{n,H}_{3,4}(\tau) \geq \frac{\epsilon}{2}\}.$$

Note that by using Lemma 4.1&4.2 one get $\mathrm{P}(E_n) \leq \mathrm{P}(\tilde{E}_n) + \alpha_n, \quad \alpha_n \longrightarrow_n 0$.
Hence it is enough to prove that $\mathrm{P}(\tilde{E}_n) \longrightarrow_n 0$.
$\mathrm{P}(\tilde{E}_n) = \mathrm{P}(\tilde{E}_n,\ \ A^{n,H}_{3,4}[\tau,\sigma] < \frac{\epsilon}{2} \cdot \sqrt{n}) + \mathrm{P}(\tilde{E}_n,\ \ A^{n,H}_{3,4}[\tau,\sigma] \geq \frac{\epsilon}{2} \cdot \sqrt{n})$;

But on the event $\tilde{E}_n$

$$\Delta Z^{n,H}_{3,4}[\tau,\sigma] = A^{n,H}_{3,4}[\tau,\sigma] - \Delta L^{n,H}_2[\tau,\sigma] > \tfrac{\epsilon}{2} \cdot \sqrt{n};$$

$$\Delta L^{n,H}_2[\tau,\sigma] \geq 0 \ \text{ Thus } A^{n,H}_{3,4}[\tau,\sigma] \geq \tfrac{\epsilon}{2} \cdot \sqrt{n};$$

Thus

$$\mathrm{P}(\tilde{E}_n) = 0 + \mathrm{P}(\tilde{E}_n,\ \ A^{n,H}_{3,4}[\tau,\sigma] \geq \frac{\epsilon}{2} \cdot \sqrt{n}).$$

Hence on the event $\tilde{E}_n$, by using Lemma 4.3 we get
$B^{n,H}_4[\tau,\sigma] = |\sigma - \tau| - I^{n,H}_4[\tau,\sigma] \geq n^{-\delta} - n^{-\frac{1}{2}+\delta} \geq n^{-\delta}(1 - n^{-\frac{1}{2}+2\delta}) = (*)$

$\exists N$ s.t $\forall n > N \ \ 1 - n^{-\frac{1}{2}+2\delta} > \frac{1}{2}$ Thus $(*) \geq \frac{n^{-\delta}}{2} \ \ \forall n > N$;
Now let us define the event
$H_n = \{\exists \sigma, \tau \in [0,T] \ \text{ s.t } \ A^{n,H}_{3,4}[\tau,\sigma] \leq n^{\delta+\frac{1}{2}},\ B^{n,H}_4[\tau,\sigma] \geq \frac{n^{-\delta}}{2},\ \Delta Z^{n,H}_{3,4}[\tau,\sigma] > 0\}$.
We have shown that $\mathrm{P}(\tilde{E}_n) \leq \mathrm{P}(H_n) + \alpha^{'}_n, \quad \alpha^{'}_n \longrightarrow_n 0$.
Hence it is enough to prove that $\mathrm{P}(H_n) \longrightarrow_n 0$.
In order to proceed let us write the model equation

$$Z^{n,H}_{3,4}(t) = Z^{n,H}_{3,4}(0) + A^{n,H}_{3,4}(t) - S^H_4(\mu^n_4 B^{n,H}_4(t)) + \tilde{\mathrm{P}}_2(S^H_4(\mu^n_4 B^{n,H}_4(t)));$$

$$\text{Where } \begin{cases} S^H_4 - \ \text{Poisson process with rate 1 ;} \\ \mu^n_4 = \mu_4 \cdot n + \hat{\mu}_4 \cdot \sqrt{n} + o(\sqrt{n}); \\ B^{n,H}_4(t) = \int_0^t \mathbb{I}_{\{Z^{n,H}_4(s) > 0\}} ds; \\ \tilde{\mathrm{P}}_2(N) = \sum_{k=1}^N \xi^2_k; \end{cases} \tag{4.7}$$

Note that $L_2^{n,H}(t) = S_4^H(\mu_4^n B_4^{n,H}(t)) - \tilde{\mathrm{P}}_2(S_4^H(\mu_4^n B_4^{n,H}(t)))$, the departure process from route 2. Therefore on the event H_n

$$\Delta Z_{3,4}^{n,H}[\tau,\sigma] > 0 \text{ equal to } L_2^{n,H}(\sigma) - L_2^{n,H}(\tau) < A_{3,4}^{n,H}[\tau,\sigma].$$

But the route departure process for H-P customers ($L_2^{n,H}(t)$) is the outcome of a thinning procedure on process $S_4^H(\mu_4^n B_4^{n,H}(t))$. One may see that $L_2^{n,H}(t)$ is determined from $S_4^H(\mu_4^n B_4^{n,H}(t))$ by random sampling, i.e., $L_2^{n,H}(t) = \sum_{k=1}^{D_4^H(t)} 1 - \xi_k^{2,H}$. Hence each departure H-P customer tosses a coin with a probability p_2 to leave and hence be counted in $L_2^{n,H}(t)$. The thinning is done by the random toss represented by $\xi_k^{4,H}$ which are iid Bernoulli R.V with probability $1-p$. Hence we may define $L_2^{n,H}(t) = S(p \cdot \mu_4^n B_4^{n,H}(t))$, where S is a standard Poisson process.

Let us define the Event

$$\tilde{H}_n = \{\bar{\sigma} = p\cdot\mu_4 n B_4^{n,H}(\sigma), \bar{\tau} = p\cdot\mu_4 n B_4^{n,H}(\tau) \;\; S(\bar{\sigma}) - S(\bar{\tau}) < n^{\delta+\frac{1}{2}}, \;\; |\bar{\sigma}-\bar{\tau}| > \frac{p\mu_4 \cdot n^{1-\delta}}{2}\}.$$

Hence

$$\mathrm{P}(H_n) \le \mathrm{P}(\tilde{H}_n) + \beta_n, \quad \beta_n \longrightarrow_n 0$$

Divide $[0, nT]$ into K intervals with length $\frac{p\cdot\mu_4 n^{1-\delta}}{4}$, when $K = \sharp$ intervals $= c\cdot n^{\delta}$. Let us define $\Delta_k S = \Delta S(J_k)$, when J_k denote time interval k. On the event $\tilde{H}_n$ there is at least one interval on which $\Delta_k S < n^{\delta+\frac{1}{2}}$.
$\mathrm{P}(\tilde{H}_n) \le \mathrm{P}(\exists k \in \{1,\dots,K\} \text{ s.t } \;\; \Delta_k S < n^{\delta+\frac{1}{2}}, |J_k| \approx n^{1-\delta}) = 1 - (1 - \mathrm{P}(\Delta_1 S < n^{\delta+\frac{1}{2}}))^K$.
Let us focus on
$\mathrm{P}(\Delta_1 S < n^{\delta+\frac{1}{2}}) = \mathrm{P}(S(n^{1-\delta}) < n^{\delta+\frac{1}{2}}) = (*)$;
$S -$ Standard Poisson process with rate 1 ;

$(*) = \mathrm{P}(\frac{S(n^{1-\delta})}{n^{1-\delta}} < n^{2\delta-\frac{1}{2}})$

$\{\delta < 1/4\}$ Thus $\exists N$ s.t $\forall n > N \;\; n^{2\delta-\frac{1}{2}} < \frac{1}{2}$; .

$(*) \le \mathrm{P}(\frac{S(n^{1-\delta})}{n^{1-\delta}} < \frac{1}{2}) \le \mathrm{P}(|\frac{S(n^{1-\delta})}{n^{1-\delta}} - 1| \ge \frac{1}{2}) \overset{\text{chebysheb}}{\le} 4\cdot \text{ Var } (\frac{S(n^{1-\delta})}{n^{1-\delta}} - 1)$;

$= \frac{4}{n^{2-2\delta}} \cdot \text{ Var } (S(n^{1-\delta})) = \frac{4n^{1-\delta}}{n^{2-2\delta}} = 4\cdot n^{\delta-1}$;

Therefore

$$\mathrm{P}(H_n) \leq 1 - [1 - 4 \cdot n^{\delta-1}]^{n^\delta} \longrightarrow_n 0;$$

$$\begin{cases} \lim (1-x_n)^{y_n} = e^{-\lim x_n y_n} \longrightarrow_n 1; \\ \\ \lim x_n y_n = \lim (n^{\delta-1} \cdot n^{\delta}) = \lim (n^{-1+2\delta}) \longrightarrow_n 0; \quad (\text{ since } \quad 2\delta \leq \frac{1}{2}) \end{cases} .$$

Which mean $\mathrm{P}(E_n) \longrightarrow_n 0$. That completes the proof of Theorem 4.1. □

Now by the *asymptotic optimality* criterion defined in 2.3, one may see that the proposed control policy is asymptotically optimal.

Note- The High-Priority Birth-Process is hard to define in the sense of probabilistic distribution, recall that any departure of Low-Priority customer in one of the routes will cause a birth of High-Priority in the other route in one of the servers (in which one? hard to define). Hence, the lemmas can not be proven by "simple" fluid and diffusion limits. Thus, a central ingredient in the proof of the Lemmas is the use of a down-crossings technique on the random process of H-P customers' queue length (see proofs on Section 4.4).

4.4 Proof of Lemmas

Now we shall prove the three Lemmas which were used in the previous Section (4.3).
Fix T, and define $\delta \in (0, 1/4)$, $\epsilon > 0$.

Proof of Lemma 4.1.

We shall prove the claim for $I_2^{n,H}[\tau, \sigma]$, the proof for $I_4^{n,H}[\tau, \sigma]$ is similar. Let us define the event

$E_n = \{\hat{Z}_1^{n,H}(s) + \hat{Z}_2^{n,H}(s) \geq \frac{\epsilon}{2} \ \ \forall s \in [\tau, \sigma], \ \ I_2^{n,H}[\tau, \sigma] > n^{-\frac{1}{2}+\delta}\}$;
We need to show that $\mathrm{P}(E_n) \longrightarrow_n 0$.

Consider the relevant system equation-

$$Z_1^{n,H}(t) = Z_1^{n,H}(0) + A_1^{n,H}(t) - S_1^H(\mu_1^n B_1^{n,H}(t)) + \tilde{\mathrm{P}}_1(S_2^{n,H}(\mu_2^n B_2^{n,H}(t)));$$

$$Z_2^{n,H}(t) = Z_2^{n,H}(0) + A_2^{n,H}(t) + S_1^H(\mu_1^n B_1^{n,H}(t)) - S_2^H(\mu_2^n B_2^{n,H}(t));$$

$$\text{Where } \begin{cases} A_1^{n,H}(t) + A_2^{n,H}(t) = L_2^{n,L}(t) = S_4^L(\mu_4^n B_4^{n,L}(t)) - \tilde{\mathrm{P}}_2(S_4^L(\mu_4^n B_4^{n,L}(t))); \\ S_1^H(t), S_2^H(t), S_4^L(t) - \text{ Are Poisson processes with rate 1 ;} \\ \mu_i^n = \mu_i \cdot n + \hat{\mu}_i \cdot \sqrt{n} + o(\sqrt{n}) \quad \forall i, \quad \mu_1 = \mu_2; \\ B_i^{n,H}(t) = \int_0^t \mathbb{I}_{\{Z_i^{n,H}(s)>0\}} ds; \\ I_j(t) = t - B_j(t) = \int_0^t \mathbb{I}_{\{Z_i^{n,H}(s)=0\}} ds; \\ \tilde{\mathrm{P}}_1(N) = \sum_{k=1}^N \xi_k^{1,H}; \\ \tilde{\mathrm{P}}_2(N) = \sum_{k=1}^N \xi_k^{2,L}; \end{cases} \tag{4.8}$$

Let us scale $Z_2^{n,H}(t)$ by $\sqrt{n}$ and get

$$\frac{Z_2^{n,H}(t)}{\sqrt{n}} = \frac{Z_2^{n,H}(0)}{\sqrt{n}} + \text{ increasing process } + \frac{S_1^H(\mu_1^n B_1^{n,H}(t)) - \mu_1^n B_1^{n,H}(t)}{\sqrt{n}} - \frac{S_2^H(\mu_2^n B_2^{n,H}(t)) - \mu_2^n B_2^{n,H}(t)}{\sqrt{n}} +$$

$$+ \frac{\mu_1^n B_1^{n,H}(t)}{\sqrt{n}} - \frac{\mu_2^n B_2^{n,H}(t)}{\sqrt{n}};$$

We shall use the identity $B_i^{n,H}(t) = t - I_i^{n,H}(t)$, and the notation
$\hat{Z}_2^{n,H}(t) = \frac{Z_2^{n,H}(t)}{\sqrt{n}}$, $\hat{Z}_2^{n,H}(0) = \frac{Z_2^{n,H}(0)}{\sqrt{n}}$, $\hat{S}_i^{n,H}(t) = \frac{S_i^H(\mu_i^n t) - \mu_i^n t}{\sqrt{n}}$, $\hat{I}_i^{n,H}(t) = \frac{I_i^{n,H}(t)}{\sqrt{n}}$, $\hat{\mu}_1 - \hat{\mu}_2 = \frac{\mu_1^n - \mu_2^n}{\sqrt{n}}$.
Thus

$$\hat{Z}_2^{n,H}(t) = \hat{Z}_2^{n,H}(0) + \text{ increasing process } + \hat{S}_1^{n,H}(B_1^H(t)) - \hat{S}_2^{n,H}(B_2^H(t)) + (\hat{\mu}_1 - \hat{\mu}_2) \cdot t +$$

$$+ \mu_2^n \cdot \hat{I}_2^{n,H}(t) - \mu_1^n \cdot \hat{I}_1^{n,H}(t);$$

Let us define

$$\hat{X}_2^n(t) = \hat{S}_1^{n,H}(B_1^H(t)) - \hat{S}_2^{n,H}(B_2^H(t)) + (\hat{\mu}_1 - \hat{\mu}_2) \cdot t.$$

Thus the following relations holds on the event E_n-

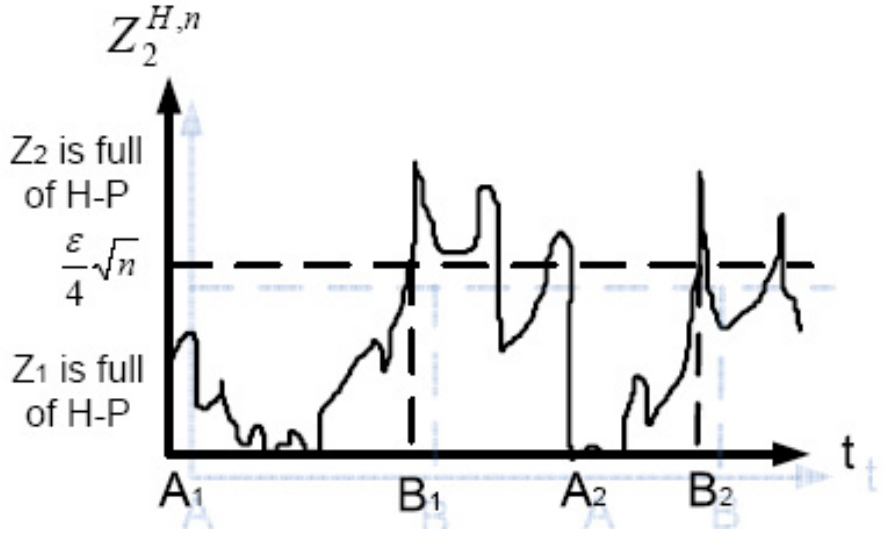
$$\hat{Z}_2^{n,H}(t) = \hat{Z}_2^{n,H}(0) + \text{ increasing process } + \hat{X}_2^n(t) + \mu_2^n \cdot \hat{I}_2^{n,H}(t) - \mu_1^n \cdot \hat{I}_1^{n,H}(t);$$

$$\begin{cases} \int_0^t \mathbb{I}_{\{\hat{Z}_2^{n,H}(s)>0\}} d\hat{I}_2^{n,H} = 0; \\ \\ \int_0^t \mathbb{I}_{\{\hat{Z}_2^{n,H}(s)<\frac{\epsilon}{4}\}} d\hat{I}_1^{n,H} = 0; \end{cases} \tag{4.9}$$

The above is derived from work conservation and from $\hat{Z}_1^{n,H}(s) + \hat{Z}_2^{n,H}(s) \geq \frac{\epsilon}{2}$ on the event E_n;

In other words, any time t s.t $\hat{Z}_2^{n,H}(t) \in (0, \frac{\epsilon}{4})$ we get from work conservation that $d\hat{I}_1^{n,H}(t) = d\hat{I}_2^{n,H}(t) = 0$.
Let us define the following random times-

$$\begin{cases} A_1 = \inf\{\tau \leq t \leq \sigma : \hat{Z}_2^{n,H}(s) = 0\}; \\ \\ B_1 = \inf\{A_1 < t \leq \sigma : \hat{Z}_2^{n,H}(s) \geq \frac{\epsilon}{4}\}; \\ \\ \text{continue in inductive manner ;} \\ \\ A_{i+1} = \inf\{B_i < t \leq \sigma : \hat{Z}_2^{n,H}(s) = 0\}; \\ \\ B_{i+1} = \inf\{A_{i+1} < t \leq \sigma : \hat{Z}_2^{n,H}(s) \geq \frac{\epsilon}{4}\}; \end{cases}$$



Notice that on the event E_n one has at least one time point- $\tau \leq A_1 \leq \sigma$, otherwise

$\hat{Z}_2^{n,H}(s) > 0 \, \forall s \in [\tau, \sigma]$ meaning that $I_2^{n,H}[\tau, \sigma] = 0$ which contradicts the event.
Note that the above construction divides $[\tau, \sigma]$ into regions, with the following property

- $[A_i, B_i)$: $d\hat{I}_1^{n,H}(s) = 0 \;\; \forall s \in [A_i, B_i), \;\; \hat{I}_2^{n,H}(s)$ increase on $\{s \in [A_i, B_i) : \;\; \hat{Z}_2^{n,H}(s) = 0\}$.
- $[B_i, A_{i+1})$: $d\hat{I}_2^{n,H}(s) = 0 \;\; \forall s \in [A_i, B_i), \;\; \hat{I}_1^{n,H}(s)$ increase on $\{s \in [B_i, A_{i+1}) : \{\hat{Z}_1^{n,H}(s) = 0\} \cap \{\hat{Z}_2^{n,H}(s) \geq \frac{\epsilon}{4}\}\}$.
- $[\tau, A_1)$: $d\hat{I}_2^{n,H}(s) = 0 \;\; \forall s \in [A_i, B_i), \;\; \hat{I}_1^{n,H}(s)$ increase on $\{s \in [B_i, A_{i+1}) : \{\hat{Z}_1^{n,H}(s) = 0\} \cap \{\hat{Z}_2^{n,H}(s) \geq \frac{\epsilon}{4}\}\}$.

Notice that on every $[B_i, A_{i+1})$ interval there is a unique time $C_i = \sup\{t \in [B_i, A_{i+1}) : \hat{Z}_2^{n,H}(s) \geq \frac{\epsilon}{4}\}$, when i is the interval index. By the definitions above one can see that on the intervals $[C_i, A_{i+1}) \;\; \hat{Z}_2^{n,H}$ starts at $\frac{\epsilon}{4}$ and ends at zero without exiting $(0, \frac{\epsilon}{4}]$. We shall call these intervals *Down Crossings*.

Claim 4.1. *Let us denote by* $R^n[\tau, \sigma]$ *the number of* Down Crossings *on* $[\tau, \sigma]$. *Then* $R^n[\tau, \sigma]$ *are tight on* E_n.
Equivalently - letting
$H_K = \{\sigma, \tau \in [0, T] \;\; \textit{as defined}\,, \;\; \hat{Z}_1^{n,H}(s) + \hat{Z}_2^{n,H}(s) \geq \frac{\epsilon}{2} \;\; \forall s \in [\tau, \sigma], \;\; R^n[\tau, \sigma] > K\}$.
Thus $\forall \eta > 0 \;\; \exists K \in \mathbb{N} \;\; s.t \; \mathrm{P}(H_K) \leq \eta$.

Proof of Claim. Notice that on $s \in [C_i, A_{i+1}) \;\; \hat{Z}_2^{n,H}(s) \in (0, \frac{\epsilon}{4})$ and therefore $\hat{I}_1^{n,H}$ and $\hat{I}_2^{n,H}$ are constant in the interval.
Hence

$$\hat{Z}_2^{n,H}(A_{i+1}) - \hat{Z}_2^{n,H}(C_i) = \text{ Positive R.V. } + \hat{X}_2^n(C_i) - \hat{X}_2^n(A_{i+1}) = -\frac{\epsilon}{4}.$$

Then on H_K using positivity of the arrival process, we have $|\Delta \hat{X}_2^n[C_i, A_{i+1})| > \frac{\epsilon}{4}$. Hence

$$H_K \subseteq \{\exists 0 \leq s_1 \leq t1 \leq s2 \leq \ldots \leq t_K \leq T \;\; \text{s.t } \; |\Delta \hat{X}_2^n[s_i, t_i)| \geq \frac{\epsilon}{4} \;\; \forall i \in \{1, \ldots, K\}\}.$$

Using the notion of modulus of continuity, we have

$$\mathrm{P}(H_K) \leq \mathrm{P}(\exists i \;\; \text{s.t } \; |\Delta \hat{X}_2^n[s_i, t_i)| \geq \frac{\epsilon}{4}, \;\; 0 \leq t_i - s_i \leq \frac{T}{K}) = \mathrm{P}(mod_T(\hat{X}_2^n, \frac{T}{K}) \geq \frac{\epsilon}{4})$$

Where $mod_T(X, \delta)$ is the modulus of continuity defined in the following way -

$$mod_T(X, \delta) = \sup \left\{ \begin{array}{l} s, t \in [0, T] \\ |s - t| < \delta \end{array} \right. |X(s) - X(t)| \tag{4.10}$$

But

$$\mathrm{P}(H_K) \leq \mathrm{P}(mod_T(\hat{X}_2^n, \tfrac{T}{K}) \geq \tfrac{\epsilon}{4}) = \mathrm{P}(mod_T\Big(\hat{S}_1^{n,H}(B_1^H(t)) - \hat{S}_2^{n,H}(B_2^H(t)) + (\hat{\mu}_1 - \hat{\mu}_2)\cdot t, \tfrac{T}{K}\Big) \geq \tfrac{\epsilon}{4})$$

$$\leq \mathrm{P}(mod_T(\hat{S}_1^{n,H} \circ B_1^H, \tfrac{T}{K}) + mod_T(\hat{S}_2^{n,H} \circ B_2^H, \tfrac{T}{K}) + mod_T((\hat{\mu}_1 - \hat{\mu}_2)\cdot t, \tfrac{T}{K}) \geq \tfrac{\epsilon}{4})$$

$$\leq \mathrm{P}(mod_T(\hat{S}_1^{n,H} \circ B_1^H, \tfrac{T}{K}) \geq \tfrac{\epsilon}{12}) + \mathrm{P}(mod_T(\hat{S}_2^{n,H} \circ B_2^H, \tfrac{T}{K}) \geq \tfrac{\epsilon}{12}) + \mathrm{P}(mod_T((\hat{\mu}_1 - \hat{\mu}_2)t, \tfrac{T}{K}) \geq \tfrac{\epsilon}{12}) \tag{4.11}$$

First term in the r.h.s of (4.11) $B_1^H(t)$ is a non-decreasing function from $[0,T]$ to $[0,T]$ which is Lipschitz continuous, in fact

$$0 \leq \frac{B_1^H(t) - B_1^H(s)}{t-s} \leq 1, \quad \forall t < s.$$

Therefore $mod_T(\hat{S}_1^{n,H} \circ B_1^H, \frac{T}{K}) \leq mod_T(\hat{S}_1^{n,H}, \frac{T}{K})$. Note that $\hat{S}_1^{n,H}(t)$ is a scaled Poisson process which converges weakly to BM process. Therefore it converges to a continuous sample-path process. As a result, it is C-tight. Hence
For each positive η_1, there exist $K_1 > 1$, $\delta = \frac{T}{K_1}$, and an integer n_1, s.t.-

$$\mathrm{P}(mod_T(\hat{S}_1^{n,H} \circ B_1^H, \frac{T}{K}) \geq \frac{\epsilon}{12}) \leq \mathrm{P}(mod_T(\hat{S}_1^{n,H}, \frac{T}{K}) \geq \frac{\epsilon}{12}) \leq \eta_1, \quad \forall n > n_1.$$

Second term in the r.h.s of(4.11) By the same way, we get
For each positive η_2, there exist $K_2 > 1$, $\delta = \frac{T}{K_2}$, and an integer n_2, s.t.-

$$\mathrm{P}(mod_T(\hat{S}_2^{n,H} \circ B_2^H, \frac{T}{K}) \geq \frac{\epsilon}{12}) \leq \mathrm{P}(mod_T(\hat{S}_2^{n,H}, \frac{T}{K}) \geq \frac{\epsilon}{12}) \leq \eta_2, \quad \forall n > n_2.$$

Third term in the r.h.s of(4.11) $(\hat{\mu}_1 - \hat{\mu}_2)t$ is a continuous linear function,

$$mod_T((\hat{\mu}_1 - \hat{\mu}_2)\cdot t, \frac{T}{K}) < (\hat{\mu}_1 - \hat{\mu}_2)\cdot \frac{T}{K}.$$

Which means that for each fixed ϵ, there exist $K > 1$, s.t.-

$$\mathrm{P}(mod_T((\hat{\mu}_1 - \hat{\mu}_2)\cdot t, \frac{T}{K}) \geq \frac{\epsilon}{8}) < \mathrm{P}((\hat{\mu}_1 - \hat{\mu}_2)\cdot \frac{T}{K} \geq \frac{\epsilon}{8}) = 0, \quad \forall K > \frac{8T}{\epsilon}\cdot(\hat{\mu}_1 - \hat{\mu}_2).$$

Concluding that For each positive η, there exist $K = max\{K_1, K_2, K_3\}$, $\delta = \frac{T}{K}$, and an integer n, s.t.

$$\mathrm{P}(H_K) \leq \eta/2 + \eta/2 + 0 = \eta$$

This proves Claim 4.1. □

Back to the proof of Lemma 4.1. Fix K, on the event $\{E_n\} \cap \{H_K^c\}$ we get
There exist $R^n \leq K+1$ intervals of $[A_i, B_i)$ on $[\tau, \sigma]$, and only on those intervals $\hat{I}_2^{n,H}(s)$ increases by $n^{\delta-1}$. Therefore there exist an interval j on which $\hat{I}_2^{n,H}[A_j, B_j)$ increases by $\frac{n^{\delta-1}}{K+1}$.
Let us define the event

$$\tilde{E}_K = \{\exists j \quad \text{s.t} \quad [A_j, B_j) \text{ as defined above }, \quad R^n[\tau, \sigma] \leq K, \quad \hat{I}_2^{n,H}[A_j, B_j) > \frac{n^{\delta-1}}{K+1}\}.$$

One can see that $\mathrm{P}(E_n) = \mathrm{P}(E_n \cap H_K^c) + \mathrm{P}(E_n \cap H_K) \leq \mathrm{P}(\tilde{E}_K) + \mathrm{P}(H_K)$.

Claim 4.2. $\mathrm{P}(\tilde{E}_K) \longrightarrow_n 0$.

In interval $[A_j, B_j)$ we get $\Delta\hat{I}_1^{n,H}[A_j, B_j) = 0$, and $\hat{Z}_2^{n,H}(s)$ is bounded by $\frac{\epsilon}{4}$ for all $s \in [A_j, B_j)$. So by Equation 4.9

$\hat{Z}_2^{n,H}(B_j) - \hat{Z}_2^{n,H}(A_j) =$ Positive R.V. $+ \Delta\hat{X}_2^n[A_j, B_j) + \mu_2^n \cdot \hat{I}_2^{n,H}[A_j, B_j) - \mu_1^n \cdot \hat{I}_1^{n,H}[A_j, B_j)$;

Or $\Delta\hat{X}_2^n[A_j, B_j) = \Delta\hat{Z}_2^n[A_j, B_j) - \mu_2^n \cdot \hat{I}_2^{n,H}[A_j, B_j) -$ Positive R.V. ;

$\mu_2^n = 0_{(n)}$, $\Delta\hat{Z}_2^n[A_j, B_j) < \frac{\epsilon}{4}$, $\Delta\hat{A}[A_j, B_j) > 0$, $\hat{I}_2^{n,H}[A_j, B_j) > \frac{n^{\delta-1}}{K+1}$;

Thus $|\Delta\hat{X}_2^n[A_j, B_j)| > |\frac{n^\delta}{K+1} - \frac{\epsilon}{4}| > |\frac{n^\delta}{K'}|$;

But $\hat{X}_2^n(t) = \hat{S}_1^{n,H}(B_1^H(t)) - \hat{S}_2^{n,H}(B_2^H(t)) + (\hat{\mu}_1 - \hat{\mu}_2) \cdot t$;

Therefore $|\Delta\{\hat{S}_1^{n,H} \circ B_1^H\}[A_j, B_j) - \Delta\{\hat{S}_2^{n,H} \circ B_2^H\}[A_j, B_j)| > |\frac{n^\delta}{K'} - |\hat{\mu}_1 - \hat{\mu}_2| \cdot |B_j - A_j|\,| > |\frac{n^\delta}{K''}|$;

Hence $\mathrm{P}(\tilde{E}_K) = \mathrm{P}(|\Delta\{\hat{S}_1^{n,H} \circ B_1^H\}[A_j, B_j) - \Delta\{\hat{S}_2^{n,H} \circ B_2^H\}[A_j, B_j)| > |\frac{n^\delta}{K''}|)$

$\leq \mathrm{P}(|\Delta\{\hat{S}_1^{n,H} \circ B_1^H\}[A_j, B_j)| > |\frac{n^\delta}{2K''}|) + \mathrm{P}(|\Delta\{\hat{S}_1^{n,H} \circ B_1^H\}[A_j, B_j)| > |\frac{n^\delta}{2K''}|)$;

$\leq \mathrm{P}(|\hat{S}_1^{n,H} \circ B_1^H|_T^* > |\frac{n^\delta}{4K''}|) + \mathrm{P}(|\hat{S}_1^{n,H} \circ B_1^H|_T^* > |\frac{n^\delta}{4K''}|)$;

$\leq \mathrm{P}(|\hat{S}_1^{n,H}|_T^* > |\frac{n^\delta}{4K''}|) + \mathrm{P}(|\hat{S}_1^{n,H}|_T^* > |\frac{n^\delta}{4K''}|) \longrightarrow_n 0$;
Due to the tightness of the scaled Poisson processes.

Now we get the following- For any given η there exist K s.t $\bar{\lim}_n \mathrm{P}(E_n) \leq \eta$, since η is arbitrary we can take it to zero and get $\lim_n \mathrm{P}(E_n) = 0$.
This completes the proof of Lemma 4.1. □

Proof of Lemma 4.2. Let $H_n = \{D_2^{n,L}(\sigma) - D_2^{n,L}(\tau) \geq n^{\frac{1}{2}+\delta}\}$.
Let

$$\alpha = \inf\{t > \tau \ : \ D_2^{n,L}(t) - D_2^{n,L}(\tau) \geq \tfrac{1}{3} \cdot n^{\frac{1}{2}+\delta}\};$$

$$\beta = \inf\{t > \tau \ : \ D_2^{n,L}(t) - D_2^{n,L}(\tau) \geq n^{\frac{1}{2}+\delta}\};$$

Note that on the event H_n one has $\tau \leq \alpha \leq \beta \leq \sigma$.
Define $\delta^{'} = \frac{\delta}{2}$, Thus

$$\mathrm{P}(H_n) = \mathrm{P}(H_n,\ I_2^{n,H}[\alpha,\beta) > n^{\delta^{'}-\frac{1}{2}}) + \mathrm{P}(H_n,\ I_2^{n,H}[\alpha,\beta) \leq n^{\delta^{'}-\frac{1}{2}}); \tag{4.12}$$

Analysis of the first term on (4.12)
$\mathrm{P}(H_n,\ I_2^{n,H}[\alpha,\beta) > n^{\delta^{'}-\frac{1}{2}}) \leq \mathrm{P}(I_2^{n,H}[\alpha,\beta) > n^{\delta^{'}-\frac{1}{2}}) \leq \mathrm{P}(I_2^{n,H}[\tau,\sigma) > n^{\delta^{'}-\frac{1}{2}}) \longrightarrow_n 0$,
according to Lemma 4.1.
Analysis of the second term on (4.12)
Notice that $B_2^{n,L}[\alpha,\beta) \leq I_2^{n,H}[\alpha,\beta)$, because $I_2^{n,H}(t)$ measures time when server 2 is not working on High-Priority customers. This is equal to the amount of time when the server is idle plus time when it is busy serving Low-Priority customers, i.e., $B_2^{n,L}(t)$. Or, equally written $I_2^{n,H}[\alpha,\beta) = I_2^{n,T}[\alpha,\beta) + B_2^{n,L}[\alpha,\beta)$.
Let us define
$\tilde{H}_n = H_n \cap \{B_2^{n,L}[\alpha,\beta] \leq n^{\delta^{'}-\frac{1}{2}}\}$.

Claim 4.3.

$$\mathrm{P}(H_n,\ I_2^{n,H}[\alpha,\beta) \leq n^{\delta^{'}-\frac{1}{2}}) \leq \mathrm{P}(\tilde{H}_n) \longrightarrow_n 0.$$

Proof of Claim 4.3. Let us write the system's equation

$$D_2^{n,L}(t) = S_2^L(\mu_2^n B_2^{n,L}(t));$$
$$\begin{cases} S_2^L - \text{ Standard Poisson process with rate 1 ;} \\ \\ \mu_2^n = \mu_2 \cdot n + \hat{\mu}_2 \cdot \sqrt{n} + o(\sqrt{n}); \end{cases}$$

Define $\bar{\tau} = \mu_2 \cdot n \cdot B_2^{n,L}(\tau)$, $\bar{\sigma} = \mu_2 \cdot n \cdot B_2^{n,L}(\sigma)$, $a_n = \frac{1}{2} \cdot n^{\frac{1}{2}+\delta}$, $b_n = \mu_2 \cdot n^{\delta^{'}+\frac{1}{2}}$. We have
$\mathrm{P}(\tilde{H}_n) \leq \mathrm{P}(S_2^L(\bar{\sigma}) - S_2^L(\bar{\tau}) \geq a_n,\ \ |\bar{\sigma} - \bar{\tau}| \leq b_n) + \alpha_n,\ \ \alpha_n \longrightarrow_n 0$;
Divide $[0, nT]$ into intervals with length b_n, on the event $\tilde{H}_n$ there exist index k s.t on interval J_k we have at least $\frac{1}{2}a_n$ increment on $S_2^L[J_k]$.

$\mathrm{P}(\tilde{H}_n) \le \mathrm{P}(\exists k \text{ s.t } S_2^L[(k+1)b_n] - S_2^L[(k)b_n] > \frac{1}{2}a_n) = 1 - [1 - \mathrm{P}(S_2^L[J_1] > \frac{1}{2}a_n)]^{\sharp \text{ intervals} = T \cdot n^{\frac{1}{2}-\delta'}}$.
When $S_2^L[J_1] = S_2^L(b_n)$. Hence, using a simple large deviation argument,

Fix $u > 0$

$$\mathrm{P}(S_2^L(b_n) > \tfrac{1}{2}a_n) = \mathrm{P}(e^{u \cdot S_2^L(b_n)} > e^{u \cdot \frac{1}{2}a_n}) \overset{\text{Markov}}{\le} \frac{E(e^{u \cdot S_2^L(b_n)})}{e^{u \cdot \frac{1}{2}a_n}} = (*);$$

$S_2^L(T)$ is a Poisson R.V. Therefore $E(e^{u \cdot S_2^L(b_n)}) = e^{-b_n \cdot (1-e^u)}$;

$$(*) = e^{-b_n \cdot (1-e^u) - u \cdot \frac{1}{2}a_n} \overset{\text{fix u=1}}{\le} e^{n^{\frac{1}{2}+\frac{\delta}{2}} \cdot ((e-1) - \frac{1}{4}\frac{\delta}{2})} \overset{\forall n > n_0}{\le} e^{-\frac{1}{2} \cdot n^{\frac{1}{2}+\frac{\delta}{2}}};$$

Hence $\mathrm{P}(\tilde{H}_n) \le 1 - [1 - e^{-\frac{1}{2} \cdot n^{\frac{1}{2}+\frac{\delta}{2}}}]^{n^{-\frac{\delta}{2}+\frac{1}{2}}} \longrightarrow_n 0$;

$$\begin{cases} \lim (1 - x_n)^{y_n} = e^{-\lim x_n y_n} \longrightarrow_n 1; \\ \lim x_n y_n = \lim e^{-\frac{1}{2} \cdot n^{\frac{1}{2}+\frac{\delta}{2}}} n^{-\frac{\delta}{2}+\frac{1}{2}} \longrightarrow_n 0; \end{cases}$$

This shows $\mathrm{P}(\tilde{H}_n) \longrightarrow_n 0$ and completes the proof of Claim 4.3; □

As a result,

$$\mathrm{P}(H_n) = \mathrm{P}(H_n,\ I_2^{n,H}[\alpha, \beta) > n^{\delta' - \frac{1}{2}}) + \mathrm{P}(H_n,\ I_2^{n,H}[\alpha, \beta) \le n^{\delta' - \frac{1}{2}}) \longrightarrow_n 0.$$

Hence the proof of Lemma 4.2 is complete. □

Proof of Lemma 4.3. Note that $\Delta\hat{D}_2^{n,L}[\tau,\sigma] \geq \hat{A}_{3,4}^{n,H}[\tau,\sigma] \geq \frac{\epsilon}{2}$, since the generation of high priority (H-P) customers in route 2 is caused by a departure of low priority (L-P) customer from route 1. Meaning that $A_{3,4}^{n,H}(t) = D_2^{n,L}(t) - \sum_{k=1}^{D_2^{n,L}(t)} \xi_k^{1,L}$.
Let us define the event

$$H_n = \{|\sigma - \tau| < n^{-\delta},\ \ \hat{Z}_1^{n,H}(s) + \hat{Z}_2^{n,H}(s) \geq \frac{\epsilon}{2}\ \ \forall s \in [\tau,\sigma],\ \ \Delta\hat{D}_2^{n,L}[\tau,\sigma] \geq \frac{\epsilon}{2}\}.$$

When $[\tau,\sigma]$ as defined in Section 4.3.
Hence $\mathrm{P}(|\sigma - \tau| < n^{-\delta},\ \ A_{3,4}^{n,H}[\tau,\sigma] \geq \frac{\epsilon}{2} \cdot \sqrt{n}) \leq \mathrm{P}(H_n)$.
Let

$$\alpha = \ \inf\{t \geq \tau\ :\ \hat{Z}_2^{n,H}(t) \geq \tfrac{\epsilon}{4}\};$$

$$\sigma' = \ \inf\{t \geq \tau\ :\ \Delta\hat{D}_2^{n,L}[\tau,t] \geq \tfrac{\epsilon}{2}\};$$

Therefore

$$\mathrm{P}(H_n) = \mathrm{P}(H_n,\ \alpha \leq \sigma') + \mathrm{P}(H_n,\ \alpha > \sigma'); \tag{4.13}$$

Analysis of the first term on (4.13)
Let us define

$$\begin{cases} \beta = \ \inf\{t \geq \alpha\ :\ \hat{Z}_2^{n,H}(t) = 0\}; \\ \\ \gamma = \ \sup\{t \geq \alpha\ :\ \hat{Z}_2^{n,H}(t) \geq \tfrac{\epsilon}{4}\}; \end{cases}$$

Notice that on the event $H_n \cap \{\alpha < \sigma'\}$ one has $\Delta\hat{D}_2^{n,L}[\tau,\alpha] < \frac{\epsilon}{2}$. Also, one can see that for all $s \in [\alpha,\beta)$ we have $\hat{Z}_2^{n,H}(s) > 0$, and by work conservation property $I_2^{n,H}[\alpha,\beta) = 0 \ \Rightarrow\ B_2^{n,L}[\alpha,\beta) = 0 \ \Rightarrow\ D_2^{n,L}[\alpha,\beta) = 0$. Therefore, $\Delta\hat{D}_2^{n,L}[\tau,\beta] < \frac{\epsilon}{2}$ which means $\tau \leq \alpha \leq \gamma \leq \beta \leq \sigma' \leq \sigma$ and $|\sigma - \tau| \geq |\beta - \alpha| \geq |\beta - \gamma|$.
On $[\gamma,\beta)$ the process $\hat{Z}_2^{n,H}(t)$ starts at $\frac{\epsilon}{4}$ and ends at zero without exiting $(0,\frac{\epsilon}{4}]$. Hence we shall refer to $|\beta - \gamma|$ as the down-crossing time of the process $\hat{Z}_2^{n,H}(t)$. Thus on the event $H_n \cap \{\alpha < \sigma'\}$ there is at least one down crossing during $[\tau,\sigma)$.

Claim 4.4. *Since $|\sigma - \tau| \geq |\beta - \gamma|$ we claim that*

$$\mathrm{P}(H_n \cap \{\alpha < \sigma'\}) \leq \mathrm{P}(|\beta - \gamma| < n^{-\delta}) \longrightarrow_n 0.$$

Proof of Claim 4.4.

As was seen before in the proof of Lemma 4.1, we have the following equations

$$\begin{aligned}
&\hat{Z}_2^{n,H}(t) = \hat{Z}_2^{n,H}(0) + \text{ increasing process } + \hat{X}_2^n(t) + \mu_2^n \cdot \hat{I}_2^{n,H}(t) - \mu_1^n \cdot \hat{I}_1^{n,H}(t);\\
&\hat{X}_2^n(t) = \hat{S}_1^{n,H}(B_1^H(t)) - \hat{S}_2^{n,H}(B_2^H(t)) + (\hat{\mu}_1 - \hat{\mu}_2)\cdot t;\\
&\begin{cases} \int_0^t \mathbb{I}_{\{\hat{Z}_2^{n,H}(s)>0\}} d\hat{I}_2^{n,H} = 0; \\ \int_0^t \mathbb{I}_{\{\hat{Z}_2^{n,H}(s)<\frac{\epsilon}{4}\}} d\hat{I}_1^{n,H} = 0; \end{cases}
\end{aligned} \tag{4.14}$$

Since $\hat{Z}_2^{n,H}(s) \in [\frac{\epsilon}{4}, 0)$ for all $s \in [\gamma, \beta)$ we get $\hat{I}_1^{n,H}[\gamma,\beta) = \hat{I}_2^{n,H}[\gamma,\beta) = 0$. In addition $|\Delta\hat{Z}_2^{n,H}[\gamma,\beta)| > \frac{\epsilon}{8}$. Thus $|\Delta\hat{X}_2^n[\gamma,\beta)| > \frac{\epsilon}{8}$.

Now using the notion of modulus of continuity, one may see that

$$\mathrm{P}(|\beta-\gamma| < n^{-\delta}) = \mathrm{P}(mod_T(\hat{X}_2^n, n^{-\delta}) \geq \frac{\epsilon}{8})$$

Where

$$mod_T(X,\delta) = \sup_{\begin{cases} s,t\in[0,T] \\ |s-t|<\delta \end{cases}} |X(s)-X(t)| \tag{4.15}$$

But

$$\begin{aligned}
&\mathrm{P}(mod_T(\hat{X}_2^n, n^{-\delta}) \geq \tfrac{\epsilon}{4}) = \mathrm{P}\Big(mod_T(\hat{S}_1^{n,H}(B_1^H(t)) - \hat{S}_2^{n,H}(B_2^H(t)) + (\hat{\mu}_1-\hat{\mu}_2)\cdot t, n^{-\delta}) \geq \tfrac{\epsilon}{8}\Big)\\
&\leq \mathrm{P}(mod_T(\hat{S}_1^{n,H}\circ B_1^H, n^{-\delta}) + mod_T(\hat{S}_2^{n,H}\circ B_2^H, n^{-\delta}) + mod_T((\hat{\mu}_1-\hat{\mu}_2)\cdot t, n^{-\delta}) \geq \tfrac{\epsilon}{8})\\
&\leq \mathrm{P}(mod_T(\hat{S}_1^{n,H}\circ B_1^H, n^{-\delta}) \geq \tfrac{\epsilon}{24}) + \mathrm{P}(mod_T(\hat{S}_2^{n,H}\circ B_2^H, n^{-\delta}) \geq \tfrac{\epsilon}{24}) + \mathrm{P}(mod_T((\hat{\mu}_1-\hat{\mu}_2)t, n^{-\delta}) \geq \tfrac{\epsilon}{24})
\end{aligned} \tag{4.16}$$

First term in the r.h.s of (4.16) $B_1^H(t)$ is a non-decreasing function from $[0,T]$ to $[0,T]$ which is Lipschitz continuous and, in fact

$$0 \leq \frac{B_1^H(t) - B_1^H(s)}{t-s} \leq 1, \quad \forall t < s.$$

Therefore $mod_T(\hat{S}_1^{n,H} \circ B_1^H, \frac{T}{K}) \leq mod_T(\hat{S}_1^{n,H}, \frac{T}{K})$. Note that $\hat{S}_1^{n,H}(t)$ is a scaled Poisson process which converges weakly to BM process. Therefore it converges to a continuous sample-path process. As a result, it is C-tight. Hence

$$\mathrm{P}(mod_T(\hat{S}_1^{n,H} \circ B_1^H, n^{-\delta}) \geq \frac{\epsilon}{12}) \leq \mathrm{P}(mod_T(\hat{S}_1^{n,H}, n^{-\delta}) \geq \frac{\epsilon}{12}) \longrightarrow_n 0.$$

Second term in the r.h.s of (4.16) By the same reason we get

$$\mathrm{P}(mod_T(\hat{S}_2^{n,H} \circ B_2^H, n^{-\delta}) \geq \frac{\epsilon}{12}) \leq \mathrm{P}(mod_T(\hat{S}_2^{n,H}, n^{-\delta}) \geq \frac{\epsilon}{12}) \longrightarrow_n 0.$$

Third term in the r.h.s of (4.16) $(\hat{\mu}_1 - \hat{\mu}_2)t$ is a continuous function, and therefore

$$\mathrm{P}(mod_T((\hat{\mu}_1 - \hat{\mu}_2)t, n^{-\delta}) \geq \frac{\epsilon}{12}) \longrightarrow_n 0.$$

This completes the proof of Claim 4.4. □

Analysis of the second term on (4.13)

We have shown before that $\tau \leq \sigma^{'} \leq \sigma$, meaning that $|\sigma - \tau| \geq |\sigma^{'} - \tau|$.

Now let us define the following event

$$\tilde{H}_n = \{|\sigma^{'} - \tau| < n^{-\delta}, \ \ \Delta\hat{D}_2^{n,L}[\tau, \sigma^{'}] \geq \frac{\epsilon}{2}\} \cap \{\hat{Z}_2^{n,H}(s) \in (\frac{\epsilon}{4}, 0], \hat{Z}_1^{n,H}(s) \geq \frac{\epsilon}{4} \ \ \forall s \in [\tau, \sigma^{'}]\}.$$

Claim 4.5.

$$\mathrm{P}(H_n, \ \alpha > \sigma^{'}) \leq \mathrm{P}(\tilde{H}_n) \longrightarrow_n 0.$$

Proof of Claim 4.5.

Note that

$$\Delta D_2^{n,T}[\tau, \sigma^{'}] = \Delta D_2^{n,L}[\tau, \sigma^{'}] + \Delta D_2^{n,H}[\tau, \sigma^{'}]; \tag{4.17}$$

Where under the event $\tilde{H}_n$

$\Delta D_2^{n,T}[\tau, \sigma^{'}] = S_2^T(\mu_2^n B_2^{n,T}(\sigma^{'})) - S_2^T(\mu_2^n B_2^{n,T}(\tau))$;

$\Delta D_2^{n,L}[\tau, \sigma^{'}] \geq \frac{\epsilon}{2} \cdot \sqrt{n}$;

Recall that

$$Z_2^{n,H}(t) = Z_2^{n,H}(0) + A_2^{n,H}(t) + S_1^H(\mu_1^n B_1^{n,H}(t)) - S_2^H(\mu_2^n B_2^{n,H}(t)).$$

Therefore

$$Z_2^{n,H}(\sigma^{'}) - Z_2^{n,H}(\tau) = \Delta A_2^{n,H}[\tau, \sigma^{'}] + S_1^H(\mu_1^n B_1^{n,H}(\sigma^{'})) - S_1^H(\mu_1^n B_1^{n,H}(\tau)) - \Delta D_2^{n,H}[\tau, \sigma^{'}] \tag{4.18}$$

Claim 4.6. $Z_2^{n,H}(\sigma^{'}-) = 0$.

Proof of Claim 4.6.
By definition

$$\begin{cases} \sigma^{'} = \inf\{t \geq \tau \ : \ \Delta\hat{D}_2^{n,L}[\tau,t] > \frac{\epsilon}{2}\}; \\ \\ D_2^{n,L}(t) = S_2^L(\mu_2^n \cdot B_2^{n,L}(t)); \end{cases}$$

Hence $D_2^{n,L}(\sigma^{'}) - D_2^{n,L}(\sigma^{'}-) = 1 \ \Rightarrow \ \dot{B}_2^{n,L}(\sigma^{'}-) = 1 \ \Rightarrow \ Z_2^{n,H}(\sigma^{'}-) = 0$ due to the preemptive priority discipline, L-P customers are served only if there are no H-P customers waiting in the resource buffer. That completes the proof of Claim 4.6. □

Therefore, according to (4.18) we get

$$\Delta D_2^{n,H}[\tau,\sigma^{'}] - Z_2^{n,H}(\tau) - \Delta A_2^{n,H}[\tau,\sigma^{'}] = S_1^H(\mu_1^n B_1^{n,H}(\sigma^{'})) - S_1^H(\mu_1^n B_1^{n,H}(\tau))$$

Note that $Z_2^{n,H}(\tau) \geq 0$, and $\Delta A_2^{n,H}[\tau,\sigma^{'}] \geq 0$, therefore

$$\Delta D_2^{n,H}[\tau,\sigma^{'}] \geq S_1^H(\mu_1^n B_1^{n,H}(\sigma^{'})) - S_1^H(\mu_1^n B_1^{n,H}(\tau)).$$

To summarize, we have

$$\begin{cases} \Delta D_2^{n,T}[\tau,\sigma^{'}] = S_2^T(\mu_2^n B_2^{n,T}(\sigma^{'})) - S_2^T(\mu_2^n B_2^{n,T}(\tau)); \\ \\ \Delta D_2^{n,L}[\tau,\sigma^{'}] \geq \frac{\epsilon}{2} \cdot \sqrt{n}; \\ \\ \Delta D_2^{n,H}[\tau,\sigma^{'}] \geq S_1^H(\mu_1^n B_1^{n,H}(\sigma^{'})) - S_1^H(\mu_1^n B_1^{n,H}(\tau)). \end{cases}$$

Hence from Equation (4.17) we get

$$\Delta D_2^{n,L}[\tau,\sigma^{'}] = \Delta D_2^{n,T}[\tau,\sigma^{'}] - \Delta D_2^{n,H}[\tau,\sigma^{'}].$$

Therefore

$$\Delta D_2^{n,L}[\tau,\sigma^{'}] \leq \left[S_2^T(\mu_2^n B_2^{n,T}(\sigma^{'})) - S_1^H(\mu_1^n B_1^{n,H}(\sigma^{'}))\right] - \left[S_2^T(\mu_2^n B_2^{n,T}(\tau)) - S_1^H(\mu_1^n B_1^{n,H}(\tau))\right].$$

Where $\begin{cases} S_1^H, S_2^T - \text{ are mutualy independent Poisson processes with rate 1 ;} \\ \\ \mu_i^n = \mu_i \cdot n + \hat{\mu}_i \cdot \sqrt{n} + o(\sqrt{n}) \quad \forall i, \quad \mu_1 = \mu_2; \end{cases}$

Define $S^{'}(t) = S_2^T(\mu_2^n \cdot B_2^{n,T}(t)) - S_1^H(\mu_1^n \cdot B_1^{n,H}(t))$. Define the event

$$E_n = \{|\sigma^{'}-\tau| < n^{-\delta}, \ \ S^{'}(\sigma^{'}) - S^{'}(\tau) \geq \frac{\epsilon}{2} \cdot \sqrt{n}\} \cap \{\hat{Z}_2^{n,H}(s) \in (\frac{\epsilon}{4}, 0], \hat{Z}_1^{n,H}(s) \geq \frac{\epsilon}{4} \ \ \forall s \in [\tau,\sigma^{'}]\}.$$

One can see that $\mathrm{P}(\tilde{H}_n) \leq \mathrm{P}(E_n)$. Hence it is enough to prove $\mathrm{P}(E_n) \longrightarrow_n 0$.
Divide $[0,T]$ into intervals with length $n^{-\delta}$. Then on the event E_n there exists an interval J_k s.t $\Delta S^{'}[J_k] \geq \frac{\epsilon}{4} \cdot \sqrt{n}$. Thus

$$\mathrm{P}(E_n) \leq \mathrm{P}(\exists k \text{ s.t } \Delta S^{'}[J_k] \geq \frac{\epsilon}{4} \cdot \sqrt{n}) = 1 - [1 - \mathrm{P}(\Delta S^{'}[J_1] \geq \frac{\epsilon}{4} \cdot \sqrt{n})]^{\sharp \text{ intervals} = T \cdot n^{\delta}}.$$

Note that on the event E_n one has $\hat{Z}_1^{n,H}(s) \geq \frac{\epsilon}{4} \quad \forall s \in [\tau, \sigma^{'})$. i.e., server 1 is always busy with H-P customers, meaning that $B_1^{n,H}[J_1] = |J_1|$. Also note that $B_2^{n,T}(t)$ is a non-decreasing function from $[0,T]$ to $[0,T]$ which is Lipschitz continuous and $B_2^{n,T}[J_1] \leq |J_1|$. Therefore

$$\mathrm{P}(\Delta S^{'}[J_1] \geq \frac{\epsilon}{4} \cdot \sqrt{n}) \leq \mathrm{P}\left(S_2^T(\mu_2^n \cdot n^{-\delta}) - S_1^H(\mu_1^n \cdot n^{-\delta}) \geq \frac{\epsilon}{4} \cdot \sqrt{n} \right).$$

We now use an elementary large deviation argument. Fix $u > 0$,
$\mathrm{P}(S_2^T(\mu_2^n \cdot n^{-\delta}) - S_1^H(\mu_1^n \cdot n^{-\delta}) \geq \frac{\epsilon}{4} \cdot \sqrt{n})) = \mathrm{P}(e^{u \cdot [S_2^T(\mu_2^n \cdot n^{-\delta}) - S_1^H(\mu_1^n \cdot n^{-\delta})]} > e^{u \cdot \frac{\epsilon}{4} \cdot \sqrt{n}})$

.

$\overset{\text{Markov}}{\leq} \frac{E(e^{u \cdot [S_2^T(\mu_2^n \cdot n^{-\delta}) - S_1^H(\mu_1^n \cdot n^{-\delta})]})}{e^{u \cdot \frac{\epsilon}{4} \cdot \sqrt{n}}}$

Also,
$E(e^{u \cdot [S_2^T(\mu_2^n \cdot n^{-\delta}) - S_1^H(\mu_1^n \cdot n^{-\delta})]}) = E(e^{u \cdot S_2^T(\mu_2^n \cdot n^{-\delta})}) \cdot E(e^{-u \cdot S_1^H(\mu_1^n \cdot n^{-\delta})]}) = e^{-\mu_2^n \cdot n^{-\delta} \cdot (1 - e^u)} \cdot e^{\mu_1^n \cdot n^{-\delta} \cdot (1 - e^u)}$

.

$= e^{(\hat{\mu}_1 - \hat{\mu}_2)(1 - e^u) \cdot n^{\frac{1}{2} - \delta}} = e^{c \cdot n^{\frac{1}{2} - \delta}}$

Hence, letting $u = 1$,
$\mathrm{P}(\Delta S^{'}[J_1] \geq \frac{\epsilon}{4} \cdot \sqrt{n}) \leq e^{c \cdot n^{\frac{1}{2} - \delta} - \frac{\epsilon}{4} \cdot \sqrt{n}} = e^{-\frac{\epsilon}{4} \cdot \sqrt{n} \cdot (1 - c^{'} n^{-\delta})} \overset{\exists N: \forall n > N}{\leq} e^{-\frac{\epsilon}{8} \cdot \sqrt{n}}; \ .$

Hence, we get

$$\mathrm{P}(E_n) \leq 1 - [1 - e^{-\frac{\epsilon}{8} \cdot \sqrt{n}}]^{T n^{\delta}} \longrightarrow_n 0.$$

Meaning that $\mathrm{P}(\tilde{H}_n) \longrightarrow_n 0$, which completes the proof of Claim 4.5. □

Finally, we conclude that

$$\mathrm{P}(H_n) = \mathrm{P}(H_n, \ \alpha < \sigma^{'}) + \mathrm{P}(H_n, \ \alpha > \sigma^{'}) \longrightarrow_n 0.$$

This completes the proof of Lemma 4.3. □

4.5 Comments About Generalization

The strategy of the proof is designed to extend to general service time distributions and a nonpreemptive discipline. However, these extensions have not been established as of this time.

Recall that

- General service time distribution refers to iid service durations.
- Non-Preemptive is a policy where service to a customer can not be interrupted before it is completed.

4.6 High-Priority Dynamics

Recall that $\hat{Z}^{n,H}_{1,2}(t) = \hat{Q}^n_2(t)$ and $\hat{Z}^{n,H}_{3,4}(t) = \hat{Q}^n_1(t)$. In Section 4.3, we showed that

$$\hat{Q}^n_1(t) \wedge \hat{Q}^n_2(t) \text{ converges uniformly to 0, in probability.}$$

In this section we discuss two important consequences of the result above.

Remark 1. *The state space for the synchronization queues may be reduced to one-dimension in the limit.*

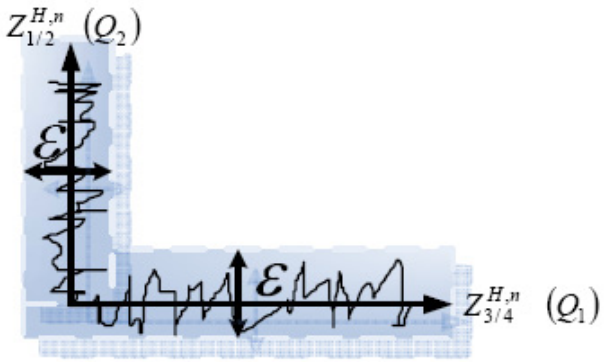


Figure 4.2: Synchronization Queues Dynamic in Heavy Traffic

One may see that the synchronization queues are restricted to an ϵ-environment around the axes, where $\epsilon > 0$ may be arbitrarily small. Hence, even though we have not formulated the limit distributions of the synchronization queues, we are able to determine that these distributions are one dimensional.

Remark 2. *At any given time there is a* critical route *which determines the customers' departure order. The critical route index is a random variable defined by the routes service dynamics.*

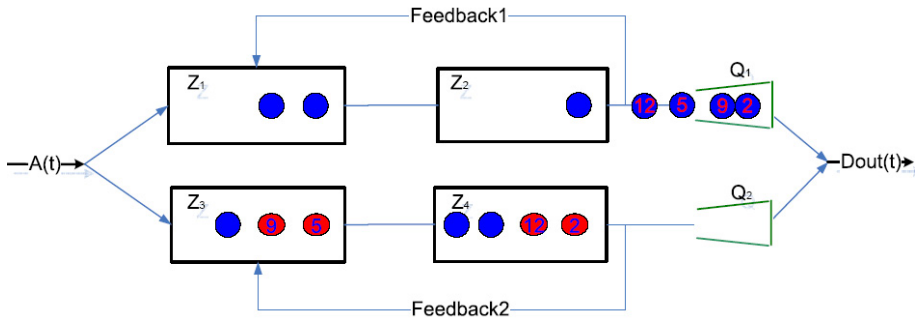


Figure 4.3: Critical Route Dynamic in Heavy Traffic

The following relation is equivalent to the above result

$$\hat{Z}^{n,H}_{1,2}(t) \wedge \hat{Z}^{n,H}_{3,4}(t) \text{ converges uniformly to 0, in probability.}$$

Hence, at any given time there is one processing route (assume this route is indexed 2) which is busy with High-Priority customers, and by the Preemptive property we may say that in this route almost only High-Priority customers are served. While the other route (assume this route is indexed 1) is free from High-Priority customers and therefore its departure process consists of almost only Low-Priority customers. Therefore, route 1 creates a birth process of High-Priority customers for route 2.
We may conclude that at any given time there is a *critical route* which is the route that serve mostly Low-Priority customers, this route determines the customers' departure order in the sense that this route departures order is similar to the system departure order. One may see that the *critical route* index can be formulated by $argmax_{i\in\{1,2\}} D_i(t)$, meaning that the route which has maximum departures till time t is the critical route.

5 Extensions and Concluding Remarks

5.1 Summary

We introduced a natural concept of optimality for a broad class of Fork-Join networks with *non-exchangeable* customers, including networks with multi-server stations and networks with probabilistic feedback. In this setting, a stochastic control problem was formulated (see Section 2.3), an optimality condition was derived via an analogy to *Assembly networks* and proved to be efficient for a general class of networks. In Sections 3 and 4, we present examples for two of the main causes for customers' disorder, i.e., multi-server stations and feedback. In these sections we formulate the specific control problem, propose optimal policies for priority control and prove asymptotic optimality for both models. These proofs introduce a characterization of the synchronization queues dynamics, under Heavy-Traffic, although limit distributions were not achieved. Finally, the results yield an **asymptotic equivalence** between non-exchangeable and exchangeable dynamics in Heavy-Traffic.

The contribution of this thesis from a mathematical perspective is the important observation of the asymptotic equivalence between non-exchangeable and exchangeable dynamics in Heavy-Traffic. As mentioned in Nguyen [34] and [35], any deviation from the simple setting which consists of feedforward network, single-server stations, single-class customers and FCFS discipline, makes the model highly intractable. Our work indicates that under our proposed priority policy, the heavy-traffic limit of the non-exchangeable Fork-Join model may be equivalent to the Assembly model Heavy-Traffic limit, which is clearly easier to analyze.

The contribution of this thesis from a system engineering perspective is the observation of the need for global control in parallel processing systems, with the rigorous formulation of these models as stochastic control problems. The models introduced here are natural models for a variety of applications, one of them is the *Multi-Project Scheduling Problem*. In this field of study there are many Heuristics but few rigourously proven results. In Section 3, we contradict the paper from Cohen, Mandelbaum and Shtub [1], by proving the asymptotic optimality of FCFS priority policy in feedforward Fork-Join model. Additionally, in Section 4, we propose a priority policy for a broader class of networks that allow probabilistic feedback, which proves to be asymptotically optimal in

conventional Heavy Traffic. Note that simulation runs for our model with feedback show improvements of 33% in sojourn time and almost 66% in synchronization time, for the proposed policy vs. FCFS in Heavy-Traffic (approximation).

5.2 Extensions

5.2.1 Modeling Extensions

When presenting the control problem for the two models, we restrict our attention to simple settings that represent the general problem. One may attempt to extend these settings in the following way:

1. Multiple processing routes: (refer to Section 2.4) The models may be generalized to include more than two processing routes. Then, the optimality condition can be expressed by

$$\prod_{i\in\{1,..,M\}} (Q_i(t)) = 0, \quad \text{or equivalently} \quad \bigwedge_{i\in\{1,..,M\}} (Q_i(t)) = 0.$$

   where M is the number of processing routes.
   Note that, in this setting, the proposed control policy should change to: At each route, assign absolute preemptive priority to customers whose service was completed in all other routes.

2. General arrival and service distribution and more general policies: The strategy of the proofs is designed to extend to general service time distributions and a non-preemptive discipline. However, these extensions have not been established as of this time.

3. General Fork-Join networks: (refer to Section 3.4) The models presented in our work consist of a single join node at the departure end of the system. We expect that this model can be extended to a general fork-join network with multiple fork and join constructs, such as in Cohen, Mandelbaum and Shtub [1].

4. Jackson network routes: The models may be combined and extended in such a way that each processing route will represent a *General Jackson Network* with Multi-server stations, as assumed in Section 2.3. It is our belief that the proposed priority policy: *At each route, assign absolute preemptive priority to customers whose service was completed in all other routes*, will be optimal in this broad setting.

5. Heterogeneous customer population: The models presented here assume single type customers, such that all jobs share the same precedence constraints, interarrival time distributions and service time distributions. However, these properties may vary across different job types in the *Heterogeneous case*. As mentioned in Nguyen [35], the setting where multiple customer types traverse possibly different routes through the network, may cause a disordering phenomenon in which tasks overtake each other.

5.2.2 The Halfin-Whitt Regime (QED)

The Halfin-Whitt regime [36] (also known as the QED regime), one increases the number of servers at the rate of N, $N \uparrow \infty$. In this setting, the disordering effect under the diffusion scaling will not be negligible, as in the multi-server case in the conventional Heavy-Traffic. Note also that, in this setting, customers bypass each other within the stations due to the servers random service times, and therefore the disordering phenomenon is uncontrollable. Such models give rise to the question: Do extreme cases of multi-server stations present a disadvantage in parallel processing systems in general and fork-join networks in particular?. This question may arise in wide variety of applications; one example appears in Kaplan [5] and [6] in the context of intelligence processing networks. In this context, the information flow between many intelligence agents (servers) and centers (stations) can be modeled by a Fork-Join network in the Halfin-Whitt regime.

References

[1] Cohen I., Mandelbaum A., Shtub A., *Multi-project scheduling and control: a process-based comparative study of the critical chain methodology and some alternatives*, Project Management Journal **35** (2004), 39–50. 1.1.3, 3.4, 5.1, 3

[2] Armony M., Maglaras C., *On customer contact centers with a callback option: customer decisions, routing rules and system design*, Operatins Research **52** (2004), 271–292. 1.1.3

[3] Chen H., Yao D., *Fundamentals of queueing networks*, Springer-Vrelag New York,Inc (2001). 4.1, 4.1

[4] Shaikhet G., *Control of many-server queueing systems in heavy traffic*, Thesis Dissertation (2007). 1.1

[5] Kaplan E. H., *Intel queues*, To Appear (2011). 5.2.2

[6] ———, *Intelligence operations research*, Oper. Res. (2011). 5.2.2

[7] Reiman M. I., *Open queueing networks in heavy traffic*, Math. Oper. Res. **9** (1984), 441–458. 1.1.2

[8] ———, *A multiclass feedback queues in heavy traffic*, Adv. Appl. Prob **20** (1988), 179–207. 1.1.2

[9] Walrand J., *An introduction to queueing networks*, Englewood Cliffs, N.J. : Prentice-Hall (1988). 1.1.3

[10] Williams R. J., *On dynamic scheduling of a parallel server system with complete resource pooling. analysis of communication networks: call centres, traffic and performance*, Fields Inst. Commun. **28** (2000), 49–71. 1.1.3

[11] Van Mieghem J.A., *Dynamic scheduling with convex delay costs: the generalized $c\mu$ rule*, Ann. Appl. Probab. **5** (1995), 809–833. 1.1.3

[12] Harrison J.M., *The diffusion approximation for tandem queues in heavy traffic*, Adv. Appl. Prob **10** (1978), 886–905. 1.1.2

[13] ———, *Brownian models of open processing networks: canonical representation of workload*, Ann. Appl. Probab. **10** (2000), 75–103. 1.1.3

[14] Cox D.R., Smith W. L., *Queues*, Methuen and Wiley (1961). 1.1.3

[15] Iglehart D. L. and Whitt W., *Multiple channel queues in heavy traffic, i and ii*, Adv. Appl. Prob **2** (1970), 150–177. 1.1.2

[16] Larson R., Cahn M., Shell M., *Improving the* N.Y.C A*-to-*A *system*, Interfaces **23** (1993), 76–96. 1.1.1

[17] Mandelbaum A. , Chen H., *Stochastic discrete flow networks: Diffusion approximations and bottlenecks*, Ann. Prob. **19** (1991), 1463–1519. 1.1.2

[18] Harrison J.M. , Lopez M. J., *Heavy traffic resource pooling in parallel-server systems*, Queueing Systems **33** (1999), 339–368. 1.1.2, 1.1.3

[19] Atar R. , Mandelbaum A. , Reiman M., *Scheduling a multi-class queue with many exponential servers: Asymptotic optimality in heavy-traffic*, Ann. Appl. Probab. **14** (2004), 1083–1134. 1.1.3

[20] Harrison J.M. , Nguyen V., *The qnet method for two-moment analysis of open queuing networks*, Queuing System **6** (1990), 1–32. 1.1.2

[21] Reiman M. I. , Simon B., *A network of proiority queues in heavy traffic: One bottleneck station*, Queueing Systems **6** (1990), 33–58. 2.5.4

[22] Fleming W. H. , Soner H. M., *Controlled markov processes and viscosity solutions*, Springer-Verlag, New York (1993). 1.1.3

[23] Mandelbaum A. , Stolyar A., *Scheduling flexible servers with convex delay costs: heavy traffic optimality of the generalized c-rule*, Oper. Res. **52** (2004), 836–855. 1.1.2, 1.1.3

[24] Harrison J.M. , Van Mieghem J. A., *Dynamic control of brownian networks: state space collapse and equivalent workload formulations*, Ann. Appl. Probab. **7** (1997), 747–771. 1.1.3

[25] Bell S. L. , Williams R. J., *Dynamic scheduling of a system with two parallel servers in heavy trafficc with resource pooling: asymptotic optimality of a threshold policy*, Ann. Appl. Probab. **11** (2001), 608–649. 1.1.3

[26] ———, *Dynamic scheduling of a parallel server system in heavy traffic with complete resource pooling: Asymptotic optimality of a threshold policy*, Electronic J. of Probability **10** (2005), 1044–1115. 1.1.3

[27] Harrison J.M. , Zeevi A., *Dynamic scheduling of a multiclass queue in the halffin-whitt heavy traffic regime*, Oper. Res. **52** (2004), 243–257. 1.1.3

[28] Peterson W. P., *A heavy traffic limit theorem for networks of queues with multiple customer types*, Math. Oper. Res. **16** (1991), 90–105. 1.1.2

[29] Atar R., *A diffusion model of scheduling control in queueing systems with many servers*, Ann. Appl. Probab. **15** (2005), 820–852. 1.1.3

[30] Avi-Itzhak B., Halfin S., *Non-preemptive priorities in simple fork-join queues*, Queueing, Performance and Control in ATM (1991). 1.1.3

[31] Varma S., *Heavy and light traffic approximations for queues with synchronization constraints*, Thesis Dissertation (1990). 1.1.4

[32] Katsuda T., *State-space collapse in stationarity and its application to a multiclass single-server queue in heavy traffic*, Queueing Systems **65** (2010), 237–273. 1.1.2

[33] Nguyen V., *Heavy traffic analysis of processing networks with parallel and sequential tasks*, Thesis Dissertation (1990). 1.1

[34] ———, *Processing networks with parallel and sequential tasks: Heavy traffic analysis and brownian limits*, The Annals of Applied Probability **3** (1993), 28–55. 1.1.4, 2.5.1, 5.1

[35] ———, *The troubles with diversity: Fork-join networks with heterogeneous customer population*, The Annals of Applied Probability **4** (1994), 1–25. 5.1, 5

[36] Halfin S., Whitt W., *Heavy traffic limits for queues with many exponential servers*, Oper. Res. **29** (1981), 567–588. 5.2.2

[37] Whitt W., *Weak convergence theorems for priority queues: Preemptive-resume discipline*, Journal of Appliad Probability **8** (1971), 74–94. 1.1.2

APPENDIX

A Simulation Tool

The objective of the simulation design is to implement a generic flexible infrastructure for the simulation, which can be used for wide variety of systems' models. This objective can be elaborated to the following demands:

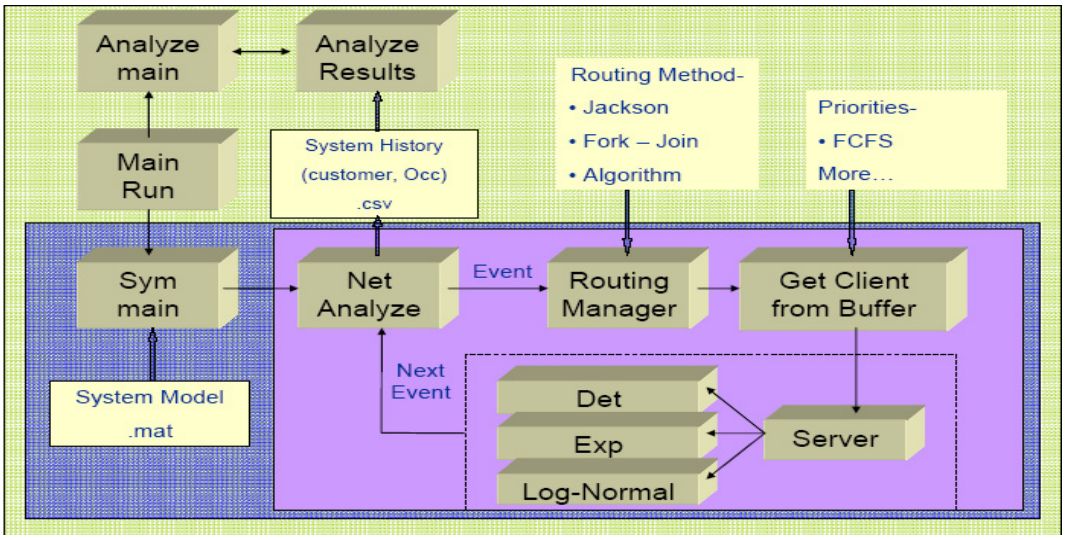
- The simulation structure and time advancement mechanism should be insensitive to the implemented network structure and parameters.
- The simulation should support any network structure imposed by the system's model.
- The simulation should support wide variety of routing and scheduling mechanisms.
- The simulation should support wide variety of sample mechanisms (distributions).
- The simulation should allow long periods of operations for complex models and still provide full system's and patients' history throughout the operation period.

Another important aspect in simulation design is automation. In order to calculate and compare system performance for a wide variety of routing policies in finite time, we ought to implement an automated ability to run, collect data and analyze several policies in one simulation batch.

The proposed simulation Design is composed of three layers, as can be seen in Figure A.1. We shall describe them from top (Automation and management layer) to bottom (Network simulation layer).

Automation and management layer: This is the external green layer in Figure A.1. The purpose of this layer is to automatize the process of building different scenarios, running them through the network simulator (see below), collecting the appropriate information and analyzing it. Typical management layer operation includes reading the user scenarios requirements from the *Main Run* process. Scenario parameters include-system definition (in M.file format, see *Scenario simulation layer* below), requested priority discipline and routing algorithm. The *Main Run* feed each scenario to the *Sym Main* process in the *Scenario simulation layer* which run the simulation. After all the scenarios

Figure A.1: Simulation Infrastructure



are processed *Main Run* activate the *Analyze Main* process which analyze the simulation data. Due to the computer memory constraints all of the system's and customers' information during network simulation is saved in excel format files (csv format), and then can be filtered and analyzed by the *Analyze Main*. The *Analyze Main* process produce and save all the system's requested statistical data and charts for each scenario. In this research the scenarios data was collected during a system's operational time of 10,000 days, and was analyzed after reduction of the first 1,000 days as a warm-up period (in order to reach steady state). The main processes (each one corresponds to the Matlab file with the same name) in this layer are:

Main Run - This is the main M. (Matlab) file from which the simulation is executed. A user inputs the requested batch of runs including systems, routing methods and priority methods for each run. This batch of runs then passes on to be processed by the network simulator sequently. There is no feedback information since the simulator information is saved in the csv files in the computer Hard-Drive. The information for a typical batch of runs may exceed 1-2 Gbyt.

Analyze Main - From this M.file the simulation analysis is executed. The user inputs the batch of information to be analyzed, including systems, routing methods and priority methods. This batch passes on to be processed by the Analyze Results sequently. The analyzed information is then gathered and compared.

Analyze Results - This M.file reads all the csv files of each scenario from the specified location, filters and analyzes the information. The relevant information includes customers' entrance time to each object (server/buffer) in the network, and the system occupancy rate at each time point. From this we can calculate and compare means and variances of LOS, waiting times for the transfer to the wards, wards occupancy rate, and so on.

Scenario simulation layer: This is the blue layer in Figure A.1. The purpose of this layer is to prepare the requested scenario for the network simulator. The main process in this layer is the *Sym Main*. This M.file gets the requested system, routing and priority from the *Main Run*, loads the appropriate system data which is saved in Matlab workspace file format (.mat), sets up the arrivals rate vector (which can vary with time), and the simulation run time and warm-up time.
System definition (.mat file) includes:

- Defining the number of objects or nodes in the system network.
- Defining the role of each node - service station, buffer, entrance node, departure node.
- Defining connections between the nodes with the help of two matrices. The first matrix contains precedence constraints, and is denoted as "Fork-Join connections". The second matrix contains probabilities according to which routing is done, and is denoted as "Jackson connections".
- Defining for each service station the requested amount of servers, service time distribution, mean and variance.

Network simulation layer: This is the purple layer in Figure A.1. This layer is the stochastic network simulation for specific scenario and system. It receives the requested scenario and system, simulates system operations throughout long periods of time and saves system information in the csv format on the computer HD.

The simulation operates according to the *event-driven* methodology. When we define *event* as the departure time of a customer from one of the system service stations or from the entrance node (which means entrance to the system). The simulation layer receives system parameters, including number of nodes in the system network - node may be a service station or a buffer, and the connection between the nodes by routing matrices (Fork-Join and/or Jackson). At each time step at the simulation there occurs a transition of customers between nodes. There may be two kinds of transitions - customer leaves

service station (after service completion) and enters the following buffer (or buffers for fork nodes), or customer enters service in some station and leaves the preceding buffer. After finishing all the current customer transitions, the event vector is updated and the next time step is set to be the minimum time event (more detailed definition can be found below).

Let us define $e = [e_1, \ldots, e_{N+1}]$ to be the event vector, where N is the number of service stations in the system. $e_i \quad \forall i \in \{1, \ldots, N\}$ denote the next customer departure time from station i, and e_{N+1} denote the next departure time from the entrance node - which mean the next customer entrance time to the system. Now let's describe the simulation operation's algorithm at each time step:

- Step 1 - Find the customer whose departure[4] time is the current step time. There will be only one departure - the probability of two simultaneous departures is negligible. Send the departing customer to *Routing Manager* in order to route him to the appropriate buffer or buffers[5].
- Step 2 - Check all service stations for idle servers - if there is at least one idle server, try to get another customer to be served. The non-idle condition and priority disciplines are carried out by the *Get Client From Buffer* module. Every customer accepted for a service in one of the service stations, is assigned to the *Server* module for calculation of his service duration.
- Step 3 - Update the system and customer's data and save it to the HD when needed (if the data exceed certain size on disc). Calculate the new event vector[6] e and define next step time as $min(e)$.

The advantage of this methodology is that the time steps are set adaptively to the system, in contrast to fixed-step time, which may miss some of the system events if they are too dense in time. In addition, it can be seen that this simulation implementation is insensitive to the requested network structure and parameters. Let us describe the main processes in this layer:

Net Analyze - This is the main simulation engine. Its operation is described above (by the above three steps). This module's input is the scenario and system received from *Sym*

[4] Remember that customer arrival to the system is defined as a departure from the entrance node.

[5] A customer may be routed to more than one buffer at once in the case of fork nodes.

[6] Collect from all of the service stations the next customer departure time- $e_i \quad \forall i \in \{1, \ldots, N\}$, and from the entrance node the next customer arrival- e_{N+1}.

Main, and his output is the csv files which are created throughout the entire simulation's operation (this way clearing computer memory during the run) and contain the system and customers information.

Routing Manager - This module is in charge of the routing discipline. The input to this module is the departure client and the station he departures from. Then the routing is performed according to Fork-Join discipline and/or Jackson discipline or by some hard-coded algorithm - by this variety of options we can perform any routing discipline needed including Feedback and close-loop algorithms. The change in customers location in the network is then updated in the network data structure.

Get Client From Buffer - This module is in charge of the non-idle condition and the priority discipline, and its job is to move customers from the preceding buffers to the service stations, when it is possible. As in the *Routing Manager*, the priority discipline can be any given algorithm, including close-loop and time-varying algorithms. In addition every given station in the system can have its own different priority discipline.

Server - This module generates an appropriate distribution sample for customers' service duration. The input to this module is the server's properties, including distribution type, mean and variance parameters. In our implementation the module can generate samples for exponential, deterministic and log-normal distributions.

רשתות Fork-Join תחת עומס גבוה:

קירובי דיפוזיה ובקרה

אסף צבירן

רשתות Fork-Join תחת עומס גבוה:

קירובי דיפוזיה ובקרה

חיבור על מחקר

לשם מילוי חלקי של הדרישות לקבלת תואר

מגיסטר למדעים בחקר ביצועים וניתוח מערכות

אסף צבירן

הוגש לסנט הטכניון – מכון טכנולוגי לישראל

אדר ב' התשע"א חיפה מרץ 2011

המחקר נעשה בפקולטה להנדסת תעשיה וניהול בהנחיית
פרופ' רמי אתר מהפקולטה להנדסת חשמל
ופרופ' אבישי מנדלבאום מהפקולטה להנדסת תעשיה וניהול.

אני מודה למנחים שלי רמי ואבישי על האמונה שלהם בי,
התמיכה וההשקעה הרבה לאורך כל עבודתנו המשותפת.
אני שמח שהייתה לי הזכות ללמוד כה רבות מכם.

בנוגע לפקולטה להנדסת תעשיה וניהול, למרות שלא התאפשר לי להגיע
בתדירות גבוהה, בכל פעם שהגעתי לפקולטה הרגשתי בבית.

אני מקדיש את התזה לאשתי אביטל ולביתי אלה, אתן ההשראה שלי.

אני מודה לטכניון על התמיכה הכספית הנדיבה בהשתלמותי

תקציר המחקר

1. מבוא

רשתות Fork-Join הינן רשתות בהן לקוח הנכנס למערכת השירות מתפצל (Fork) למספר מטלות המבוצעות במקביל ובטור עד לסיום כל המטלות, מיזוג (Join) המטלות המוגמרות ויציאת הלקוח מהמערכת. רשתות אלו הינן ייצוג טבעי למספר רב של שימושים ומערכות כגון מערכות מחשוב מקבילי, תקשורת של חבילות מידע דרך שרתי אינטרנט, מערכות ייצור, ניהול ארגונים גדולים ובפרט ניהול מחקר ופיתוח של מספר רב של פרויקטים, מערכות שירות ומערכות בריאות, אכן, במערכות בריאות בכלל ובבתי חולים בפרט ניתן לראות דוגמאות רבות של תהליכים בהם מעורבים פיצול ומיזוג של מטלות ואינפורמציה. למשל, החלטה על אשפוז מטופל בחדר מיון מערבת מספר רב של גורמים והחלטות כגון פענוח בדיקות דם, צילומי רנטגן, בדיקת אחות, בדיקת רופא והיסטוריה רפואית.

במודלים הנחקרים בעבודה זו המטלות משויכות חד-חד-ערכית ללקוח הנכנס לשירות ולכן הן ייחודיות (Non-Exchangeable), במובן שמיזוג העבודה המוגמרת ויציאת הלקוח מתאפשרת רק אם כל המטלות המשויכות אליו סיימו שירות ומוזגו יחדיו. תכונה זו ניתנת להמחשה כאילו כל לקוח הנכנס לשירות מקבל מספר זיהוי ייחודי המוצמד לכל אחת ממטלותיו ומבחין ביניהן לבין מטלות של לקוחות אחרים. תכונה זו מאפיינת שימושים ומערכות רבות כגון מערכות מחשוב, תקשורת, ניהול ארגונים, מערכות שירות ומערכות בריאות. הדוגמה הנגדית למערכות עם לקוחות ייחודיים היא מערכות ייצור בהם כל החלקים היוצאים מפס ייצור מסוים הינם זהים וניתנים למיזוג עם כל חלק אחר מפס ייצור אחר, אנו נקרא ללקוחות אלו לקוחות לא ייחודיים (Exchangeable).

במודל הכולל לקוחות ייחודיים באה לידי ביטוי בעיה של חוסר סנכרון בסדר יציאת מטלות (לקוחות) בערוצי עיבוד מקבילי שונים, תופעה זו גוררת הצטברות מטלות הממתינות למיזוג בתורי הסנכרון (תורים עבור מטלות שסיימו עיבוד ומחכות למיזוג עם מטלות בערוצי עיבוד מקבילים אחרים), ירידה בתפוקת המערכת ואי ניצול נכון של פוטנציאל התפוקה של השרתים בנקודות המיזוג של המערכת. כמובן שתופעה זו אינה קיימת במודלים הכוללים לקוחות לא ייחודיים (מכיוון שאין משמעות לסדר הלקוחות) ותכונה המאפיינת מערכות אלו היא שאחד מתורי הסנכרון חייב להיות ריק בכל זמן (זהות התור הריק הינה משתנה מקרי).

הגדרנו בעיית בקרה סטוכסטית עבור מודל Fork-Join כללי עם לקוחות ייחודיים במובן של קבלת תפוקת מערכת מכסימלית לכל אינטרוול זמן סופי. הוכחנו כי מדיניות בקרה אופטימאלית עבור

בעיה זו קיימת, ומקיימת תכונה האנלוגית לחלוטין לתכונה שתוארה עבוד מודל עם לקוחות לא ייחודיים, כלומר מדיניות בקרה הינה אופטימאלית אם ורק אם אחד מתורי הסנכרון ריק בכל זמן. כמו כן הוכחנו כי מדיניות בקרה הינה אופטימאלית אסימפטוטית בעומס גבוה עבור תנאי אנלוגי לתנאי המתקיים במודל עם לקוחות לא ייחודיים.

בהמשך הוכחנו קיום של מדיניות אופטימאלית אסימפטוטית בעומס גבוה עבור שני מודלים המייצגים משפחות רחבות של מערכות שירות.

2. בקרה אסימפטוטית אופטימלית לרשת הכוללת תחנות מרובות שרתים

בחלק זה של המחקר ניתחנו בעיית בקרה של רשת הכוללת מספר תחנות שירות בשני ערוצי שירות מקביליים, כך שכל תחנה מורכבת ממספר שרתים. ניתן לראות כי במערכות מסוג זה, לקוחות עוקפים את חבריהם בתוך תחנות השירות כתוצאה מתהליך השירות הסטוכסטי. עקב כך, תהליך יציאת הלקוחות בשני הערוצים המקביליים תחת מדיניות FCFS (הראשון שמגיע לתחנה נכנס ראשון לשירות) איננו מסונכרן ותנאי האופטימאליות איננו מתקיים עבור מדיניות זו. עבור מודל זה הוכחנו כי, תחת עומס גבוה, מתקיים תנאי האופטימאליות האסימפטוטית עבור FCFS, כלומר תחת מדיניות זו בעומס גבוה תור הסנכרון המינימלי שואף ל-0 כשהנצילות שואפת ל-1 (קצב הכניסה לתחנה וקצב השירות בתחנה בגבול הינם זהים) . תוצאה זו מצביעה על שקילות של ביצועי המערכת עם לקוחות ייחודיים לביצועי מערכת ייצור עם לקוחות לא ייחודיים.

3. בקרה אסימפטוטית אופטימלית לרשת הכוללת מנגנון החזרה לאחור הסתברותי

בחלקו השני של המחקר ניתחנו בעיית בקרה של רשת שבה מאפשרים החזרה לאחור של לקוח לתחילת ערוץ שירות מקבילי בהסתברות מסוימת (הגרלת משתנה ברנולי). תכונה זו מאפיינת שמושים ומערכות רבות בעולם האמיתי, לדוגמה- ביצוע בדיקת איכות בסוף תהליך שירות / ייצור / פיתוח, לאחר בדיקה זו מתקבלת החלטה האם המטלה בוצעה כשורה או שנדרש לבצעה שוב מההתחלה. תכונה זו מאפיינת פעמים רבות מערכות רפואיות ותהליכים בבתי חולים בהם מבוצעות בדיקות קפדניות של כל תהליך כגון פענוח צילום רנטגן, בדיקת דם, אבחון רפואי. ניתן לראות שבמערכות מסוג זה סדר הלקוחות

בערוצי שירות מקביליים שונים הוא שונה, עקב תופעת הערבוב המתבצעת ע"י מנגנון ההחזרה לאחור ההסתברותי.

עבור מודל זה ניתן לראות כי FCFS איננו אופטימאלי אסימפטוטית, על כן בעיה זו קשה יותר לפתרון. בעבודה זו הגדרנו את מדיניות הבקרה הבאה:

בכל ערוץ ניתן עדיפות עליונה לשירות לכל לקוח ששירותו הסתיים בכל ערוצי השירות האחרים.

עבור המודל הנידון ומדיניות זו הוכחנו כי, תחת עומס גבוה, מתקיים תנאי האופטימאליות האסימפטוטית, כלומר תחת מדיניות זו בעומס גבוה תור הסנכרון המינימלי שואף ל-0 כשהנצילות שואפת ל-1 (קצב הכניסה לתחנה וקצב השירות בתחנה בגבול הינם זהים). תוצאה זו מצביעה על שקילות של ביצועי המערכת עם לקוחות ייחודיים לביצועי מערכת ייצור עם לקוחות לא ייחודיים. כמו כן תוצאות סימולציה הראו כי מדיניות זו משפרת את זמן השהיה במערכת ב-33% וזמן השהיה בתורי הסנכרון בכמעט 66%.

4. כיוונים להמשך מחקר

בעבודה זו הצגנו מודלים פרטניים המייצגים משפחות כלליות של רשתות, אנו הגבלנו את עצמנו לאילוצים מסוימים שייפשטו את הצגת ופתרון בעיית האופטימיזציה האסימפטוטית בעומס גבוה. ניתן להרחיב את המודלים המוצגים באופנים הבאים:

- ניתן להרחיב את מודל הרשת למודל הכולל מספר סופי כלשהוא של רשתות שירות מקביליות.
- ניתן להרחיב את המודל להתפלגות שירות כללית ולמדיניות מתן עדיפויות ללקוחות שאינה מאפשרת הפסקת שירות ללקוח שכבר נמצא בעמדת שירות (Non-Preemptive).
- ניתן להרחיב את מודלי הרשתות לתצורת Fork-Join כללית יותר הכוללת מספר רב של נקודות פיצול ונקודות מיזוג מטלות.
- ניתן לאחד את שני המודלים שהוצגו להלן (סעיף 2 ו-3) ולהרחיבם כך שכל ערוץ שירות מקבילי ייצג רשת Jackson.

אנו מאמינים כי המדיניות המוצעת בסעיף 3 תהיה אופטימאלית אסימפטוטית בעומס גבוה לכל הרחבות המודל המוצעות להלן.

- ניתן להרחיב את המודל כך שיכלול מספר רב של סוגי לקוחות עם מסלול והתפלגות שירות ייחודית לכל סוג לקוחות.

כמו כן אנו מציעים לבדוק את בעיית הבקרה האופטימאלית שהוגדרה בעבודה זו תחת עומס גבוה בתחום תפעולי שונה הנקרא Halfin-Whitt. במודל זה הנצילות שואפת ל-1 כך שמספר השרתים שואף לאינסוף ובגבול קצב הכניסה לתחנה וקצב השירות בתחנה הינם זהים. תחת מודל עומס גבוה, בעיית הבקרה המוגדרת כאן אינה פתירה ועל כן לא קיימת מדיניות המביאה את ביצועי המערכת עם לקוחות ייחודיים לביצועי מערכת ייצור עם לקוחות לא ייחודיים. מעניין לבדוק האם בעיית הסנכרון הנגרמת במודל זה מצביעה על יתרון עבור רשתות עיבוד מקבילי בהן מספר השרתים בכל תחנה הוא מוגבל, כלומר קיים חיסרון לגודל.