Service Engineering: Data-Based Course Development and Teaching

Full Version *

Avishai Mandelbaum
Faculty of Industrial Engineering & Management,
Technion, Haifa 32000, Israel,
avim@tx.technion.ac.il

Sergey Zeltyn
IBM Research Lab, Haifa 31905, Israel,
sergeyz@il.ibm.com

July 20, 2010

*There exists an abbreviated version of the present document [68]. In that version, we reduce to a minimum the description of lectures and homework (Sections 4 and 5), and we omit Section 6, on The Fusion of Research and Teaching.

Abstract

In this exposition, we discuss empirically-based teaching in the newly emerging field of *Service Engineering*. Specifically, we survey a "Service Engineering" course, taught at the Technion - Israel Institute of Technology. The course "was born" about fifteen years ago as a graduate seminar in "Service Networks", continued as an elective course and ultimately took its present form [83], as a core course for the undergraduate program in Industrial Engineering and Management.

The role of measurements and data as teaching-enhancers and research-drivers is underscored. In addition, we emphasize that data granularity must reach the individual-transaction level. We describe customized databases and software tools that facilitate operational and statistical analysis of services; this includes the use of SEEStat, a data-user interface that was developed at the Technion's SEE Lab [84], for research and educational purposes. Some unique aspects of the course are the incorporation of state-of-the-art research and real-world data in lectures, recitations and homework assignments, as amply presented throughout this work.

The three building blocks for an operational model of a basic service system, as we perceive it, are *arrivals* of service requests, *durations* of service transactions and delay-generated *(im)patience* of customers. These are fused, via service-*protocols*, into models of processing/queueing networks which, in turn, form our theoretical framework for Service Engineering. Within this framework, one develops insights, design principles (rules-of-thumb) and tools, for example in support of staffing and controls.

The application focus of the surveyed course has been telephone *call centers*, which constitute an explosively-growing branch of the service industry. Indeed, due to their prevalence and the diversity of their operational problems, call centers give rise to numerous challenges for Service Sciences, Engineering and Management. The course is now expanding to also cover healthcare, especially hospitals; some examples from other service areas (e.g. the justice system) are described as well.

Contents

1 Introduction to Service Engineering

1.1 Motivation and Contents of the Paper

The service sector is central in the life of post-industrial societies - more than 70% of the Gross National Product in most developed countries is due to this sector. In concert with this state of affairs, there exists a growing demand for high-quality multi-disciplinary research in the field of services, as well as for a significant number of *Service Engineers*, namely scientifically-educated specialists that are capable of designing service systems, as well as solving multi-faceted problems that arise in their practice. (Readers are referred to Fitzsimmons and Fitzsimmons [23], especially Part 1, for background on Services, including some distinguishing characteristics of Services relative to Manufacturing.)

In the U.S., the academic home for the area of Services has traditionally been the Business School, where Services have been taught most often as Service Marketing. Another business school option is Service Management, either within Operations Management courses or, rather rarely, as a stand-along Service Operations course. As an engineering discipline, the natural home for Services are Industrial Engineering units. Indeed, our original use of the term "Service Engineering" was conceived by combining the relevant words in "Service Management" and "Industrial Engineering". We had roughly in mind a "*New Age Industrial Engineer* that must combine technological (and scientific) knowledge with process design in order to create (and operate) the (service) delivery systems of the future [28]."

It is our belief that there exists a broad gap between academia's supply and the demand for Service Science and Engineering. Focusing on the education, universities either do not offer Service Engineering courses or, when they do so, the education quality is typically not at par with that of the traditional engineering disciplines, in particular that provided by leading Industrial Engineering units.

The goal of the *"Service Engineering"* (ServEng) course, "born" at the Technion - Israel Institute of Technology about fifteen years ago [83], is to narrow down the educational gap between demand and supply, while also stimulating Service research. As we perceive it, the ultimate goal of Service Engineering is to *develop scientifically-based design principles and tools (often culminating in software), that support and balance service quality, efficiency and profitability, from the likely conflicting perspectives of customers, servers, managers, and often also society.* The goal of our ServEng course is more restricted: we take an Operations point of view, hence we focus on *operational service quality*, on *service efficiency* and the tradeoffs between the two.

Our ServEng course takes a *data-based approach to teaching.* Thus, throughout the paper, course contents will be illustrated with the help of practical examples, drawing from

data-based case studies. Students encounter such case studies continuously during lectures, recitations and homework assignments. For example, a concept can be illustrated in a recitation via data from bank-tellers' service, homework practices the concept on call center data, and students are tested on it using data from an emergency department.

The remaining part of Section 1 is dedicated to our paradigm of the Service Engineering discipline. Specifically, Section 1.2 presents the course homepage, where the ServEng materials can be downloaded. Section 1.3 introduces Service Networks as our modeling framework for a service operation, with queueing theory and science constituting the main theoretical foundation. Section 1.4 elaborates on relevant milestones in the evolution of Queueing Theory. Then, Section 1.5 focuses on the main challenges, goals and methods in the ServEng discipline, in the context of the course. In this part, we also explain the need for approximations (compromises) in modeling and analysis of service systems.

Section 2 elaborates on the goals of the ServEng course and provides a brief survey of its history. Section 3 explains our data-based teaching approach and introduces the Data-MOCCA (Data Model for Call Center Analysis) system: through its graphical user interface, SEEStat, this system enables a friendly effective access to several large data repositories from service systems (currently call centers and emergency departments), which have been used for research and teaching (specifically in lectures, recitations and homework of the ServEng course).

Section 4 is the main one of our paper. Here we go over all lectures of the course, touching on various theoretical techniques and practical applications, spiced with practical examples that support the course material. Section 5 provides example of a data-based homework assignment, where students must use DataMOCCA/SEEStat tools for the analysis of call center data. Then, Section 6 explains our views on the relation between research and teaching, in terms of research that has been influenced by our course (in particular, carried out by course graduates) and/or has been based on data from the course repositories. We conclude in Section 7 with acknowledgements to those who helped bring the course to where it is now, interwinding it with some additional course history.

1.2 Course Homepage(s)

Before continuing, readers are advised to browse through the ServEng website, and get an idea of its structure and contents. References to relevant links will be provided as we go along. For example, the "Table of Contents" of the recently taught course is in `http://ie.technion.ac.il/serveng/Lectures/Schedule1.doc`.

The formal course website is `http://ie.technion.ac.il/serveng`, which includes the material of a *complete* course - the last one to have been taught. Then, as the course is

being offered, a second website is constructed gradually and updated as the course progresses; the address of this dynamic website is appended by the course's year and semester. For example, the site of the course taught during the Spring semester of 2010 has the address http://ie.technion.ac.il/serveng2010S. (The earlier Winter semester had 2010W appending its address.)

1.3 Service Networks: Models of Congestion-Prone Service Operations

The title of this section reflects our angle on service operations - we often view them as stochastic (random) or deterministic (fluid) systems, within the Operations Research paradigm of Queueing Networks. To support this view, let us first present our conception of the roles of Queues in services, from the perspectives of customers, servers and managers. We shall then describe Service Networks, continuing with some relevant queueing-theory history.

1.3.1 On Queues in Service

Queues in services are often the arena where customers, service-providers (servers) and managers interact (establish contact), in order to jointly create the service experience. Process-wise, queues play in services much the same role as inventories in manufacturing (see JIT = Just-in-Time, TBC = Time-based-Competition, etc.). But, in addition, "human queues" express preferences, complain, abandon and even spread around negative impressions. Thus:

- *Customers* treat the queueing-experience as a window to the service-providing party, through which their judgement of it is shaped for better or worse.
- *Servers* can use the queue as a clearly visible proxy for the state of the system, based on which, among other things, service protocols can be exercised (eg. customers priorities).
- *Managers* can use queues as indicators (queues are the means, not the goals) for control and improvement opportunities. Indeed, queues provide unbiased quantifiable measures (these are not abundant in services), in terms of which performance is relatively easy to monitor and goals (mainly tactical and operational, but sometimes also strategic) are naturally formulated.

Our point of view is thus clear: the design, analysis and management of queues in service operations could and should constitute a central driver and enabler in the continuous pursuit of service quality, efficiency and profitability.

1.3.2 On Service Networks and their Analysis

Service Networks here refer to *dynamic (process)* models (mostly analytical, sometimes empirical, and rarely simulation) of a service operation as a *queueing network*. The dynamics is that of serving *human customers*, either directly face-to-face or through phone-calls, email, internet etc. Informally, a *queueing network* can be thought of as consisting of interconnected service stations. Each station is occupied by servers who are dedicated to serve customers queued at the station.

In its simplest version, the evolution of a service station over time is stationary, as statistically-identical customers arrive to the station either exogenously or from other stations. Upon arrival, a customer is matched with an idle server, if there is any; otherwise, the customer joins a queue and gets served first-come-first-served. Upon service completion, customers either leave the network or move on to another station in anticipation of additional service. Extensions to this simplest version cover, for example, models with non-stationary arrivals (peak-loads), multi-type customers that adhere to alternative service and routing protocols, customers abandonment while waiting (after losing their patience), finite waiting capacities that give rise to blocking, splitting (fork) and matching (join) of customers, and much more.

In analyzing a Service Network, we find it useful to be guided by the following four steps (though, unfortunately, most often only the first three are applied or even applicable):

- *Can we do it?* Deterministic capacity analysis, via process-flow diagrams (spreadsheets, linear programming), which identifies resource-bottlenecks (or at least candidates for such) and yields utilization profiles.

- *How long will it take?* Typically stochastic response-time analysis, via analytical queueing network models (exact or approximate) or simulations, which yields congestion curves. Note that when predictable variability prevails and dominates, then the "fluid view" and the corresponding deterministic methods are appropriate; see Section 4.4.

- *Can we do better?* Sensitivity and Parametric ("what-if") analysis, of Measures of Performance (MOP's) or Scenarios, which yields directions and magnitudes for improvements.

- *How much better can we do?* or put simply: "What is optimal to do?" This type of questions has been typically addressed via Optimal Control (exact or asymptotic) which, as a rule, is difficult but becoming more and more feasible. (Another developing research direction is optimization via simulation - see [21] for a call-centers example, which is now in the process of being incorporated into ServEng.)

Several examples of service networks will be covered in the sequel; see, for example, the Dynamic-Stochastic (DS) PERT/CMP networks (sometimes referred to as fork-join or split-match networks) in Section 4.3.

1.4 Some Relevant History of Queueing-Theory

Queueing theory provides a central mathematical foundation for ServEng. Accordingly, a variety of queueing models, including Markovian, fluid and heavy-traffic, are studied in our course. It is thus appropriate to cover some relevant milestones in this theory's evolution.

1.4.1 The Early Days

The father of Queueing Theory was the Danish telecommunication engineer Agner Krarup *Erlang* who, around 1910-20, introduced and analyzed the first mathematical queueing models. Erlang's models [18] are standardly taught in elementary/introductory academic courses (for example $M/M/n, M/M/n/n$), as they are still corner-stones of today's telecommunication models (where $M/M/n/n$ is known as Erlang-B, "B" apparently for Blocking - the central feature of this model, and $M/M/n$ is referred to as Erlang-C, "C" conceivably because it is subsequent to "B"). Moreover, and more relevant to our present discussion, $M/M/n$ is still the work-horse that supports workforce decisions in many telephone call centers; see Section 4.6.1.

Another seminal contributor to Queueing Theory, Scandinavian (Swedish) as well, is Conny Palm, who in 1940-50 added to Erlang's $M/M/n$ queue the option of customers abandonment [76]. We shall refer to Palm's model as Palm/Erlang-A, or just Erlang-A for short (unfortunately and rather unjustly to Palm, but Erlang "was there first".) The "A" stands for Abandonment, and for the fact that Erlang-A is a mathematical interpolation between Erlang-B and Erlang-C. Palm, however, has been mostly known for his analysis of time-varying systems, also of great relevance to service operations and hence covered, in some way, in ServEng.

A next seminal step (one might say a "discontinuity" in the evolution of Queueing Research) is due to James R. Jackson, who was responsible for the mathematical extension of Erlang's model of a *single* queueing-station to a system of *networked* queueing stations, or Queueing Networks, around 1955-1965. Jackson was motivated by manufacturing systems and actually analyzed open and semi-open networks. Closed networks, relevant to healthcare and call centers with an answering machine, as it turns out, were analyzed in the mid 60s by William J. Gordon and Gordon F. Newell. Interestingly, Newell was a Transportation Engineer at Berkeley and also the earliest influential advocator of incorporating Fluid Models as a standard part of Queueing Theory - see his text book [73]. A student of Newell,

Randolph W. Hall, who is currently a Professor at University of Southern California, wrote an excellent Queueing book [35] that has influenced our teaching of Service Engineering; Hall is currently working on healthcare systems, adopting the fluid-view (described below) to model the flows of patients in hospitals; see [36].

Jackson networks are the simplest theoretically tractable models of queueing networks. (Their simplicity stems from the fact that, in steady state and at a fixed time, each station in the network behaves like a naturally-corresponding birth-death model, *independently* of the other stations.) The next step beyond Jackson networks are BCMP/Whittle/Kelly networks, where the heterogeneity of customers is acknowledged by segregating them into classes. But service operations often exhibit features that are not captured by Jackson and BCMP/Whittle/Kelly networks. Further generalizations are therefore needed, which include precedence constraints (fork-join, or split-match networks), models with one-to-many correspondence between customer types and resources (skills-based routing, agile workforce), and models that exhibit transient behavior (as opposed to steady-state analysis).

1.4.2 QED Queues

The key tradeoff in running a service operations is that between service efficiency and quality, which queueing models are ideal to capture. This tradeoff is most delicate in large systems (many servers), but here exact analysis of queueing models turns out to be limited in its insight. This was already recognized by Erlang around 1910, and later by Pollaczek around 1930, who both resorted to approximations. However, the first to put operational insights of many-server queues (as relevant to us) on a sound mathematical footing were Shlomo Halfin and Ward Whitt, at the early 80s, in the context of Erlang-C (and its extension GI/M/n). They introduced what we shall call QED Queues [34], which stands for queues that are *both* Quality- and Efficiency-Driven, hence their name.

QED Queues emerge within an asymptotic framework that theoretically and insightfully supports the analysis of the efficiency-quality tradeoff of *many-server* queueing systems. The theory culminates in a simple rule-of-thumb for staffing, the square-root staffing rule: if the offered load is R Erlangs (R=arrival rate times average service time), then

$$n \approx R + \beta\sqrt{R}$$

is a staffing level that would appropriately balance quality and efficiency. Here β is a constant, which corresponds to grade-of-service; its specific value (which is relatively small - less than 1.5 in absolute value) can be determined by economic considerations (for example the ratio between delay costs and staffing costs).

Prime examples of QED queues are well-run telephone call centers; but to properly

model these, one must generalize the Halfin-Whitt framework to allow for customers' impatience. This was done in a Technion M.Sc. thesis by Ofer Garnett, in the late 90s, and later published in [31]. At that same time, a generalization of the Halfin-Whitt framework to time-varying Jackson-like networks was carried out with Bill Massey and Marty Reiman [57] (under the name Markovian Service Networks). In analogy to QED queues, there are also ED (Efficiency-Driven) and QD (Quality-Driven) queues, all arising from asymptotic analysis as well: Erlang-C, in these three operational regimes, was treated in [9]; generalizations to Erlang-A and relatives is the subject of the Ph.D. thesis of the second author (S.Z), published in [96]. The research-part of the website of Ward Whitt is recommended for further references on QED/ED/QD queues [92].

QED approximations turn out to be amazingly accurate and robust. This arose explicitly in [9], where *square-root staffing* was shown to be an asymptotically optimal staffing level (n in an M/M/n queue). It turns out that the square-root recipe rarely deviates from the actual optimal value by more than 1 server, over a wide range of staffing levels - from the very few (10 or so) to the very many (1000s). This unexpected accuracy has been recurring for many other models since, and it can now be explained theoretically for the Erlang C/B/A models, as well as M/D/n; see, for example, [43]. The accuracy of QED approximations has practical significance since it expands the scope of these approximations to relatively small systems - in particular healthcare systems (e.g. emergency departments and internal wards). This was first observed by Otis Jennings and Francis de Vericourt [45]. It is further expanded on in the ongoing PhD thesis of Galit Yom-Tov [95], which supports decisions on static capacity (hospital beds) and dynamic capacity (e.g. nurses). Interestingly, Galit's basic queueing model for beds+nurses is the one that also appeared in the M.Sc. thesis of Polyna Khudyakov [46] on call centers with an answering machine - indeed, beds correspond to trunk-lines and nurses to telephone agents.

1.4.3 ED Queues

A balance between service quality and efficiency is often desirable, yet sometimes it is infeasible, or perhaps even not optimal. Then one settles for an ED (Efficiency-Driven) service system, in which the focus is on high utilization of resources (the servers), at the cost of service quality (relatively long delays). In call centers, ED performance arises when resources are restricted (e.g. government services); however, with large enough of a system, service level could still be acceptable or even better. (See Ward Whitt's homepage [92] for theoretical papers on many-server ED Queues.) But ED performance is in fact prevalent in healthcare, which renders small-server heavy-traffic asymptotics more relevant - see, for example, the ongoing M.Sc. thesis of Asaf Zviran [98], on fork-join queues in heavy-traffic.

1.4.4 Summary

To summarize, queueing networks have been successfully used to model systems of manufacturing, transportation, computers and telecommunication. For us they serve as indispensable models of service systems, in which customers are human and queues, broadly interpreted, capture prevalent delays in the service process. The service interface could be phone-to-phone (naturally measured in units of seconds), or face-to-face (in minutes), fax-to-fax (hours) letter-to-letter (days), face-to-machine (e.g., ATM, perhaps also Internet), etc. The finer the time-scale, the greater is the challenge of design and management. Accordingly, the greater is the need for supporting rigorous models, a need that further increases with scale, scope and complexity.

1.5 Service Engineering: Challenges, Goals and Methods

As already mentioned, we have been advocating the terminology *"Service Engineering"* to describe our research, teaching and consulting on service systems. Based on our experience, we now elaborate on the role of the Service *Engineer* as a natural mediator between the Service *Scientist* and the Service *Manager*.

1.5.1 Our Service Science and Service Engineering Paradigm

Figure 1: **Service Science Paradigm**

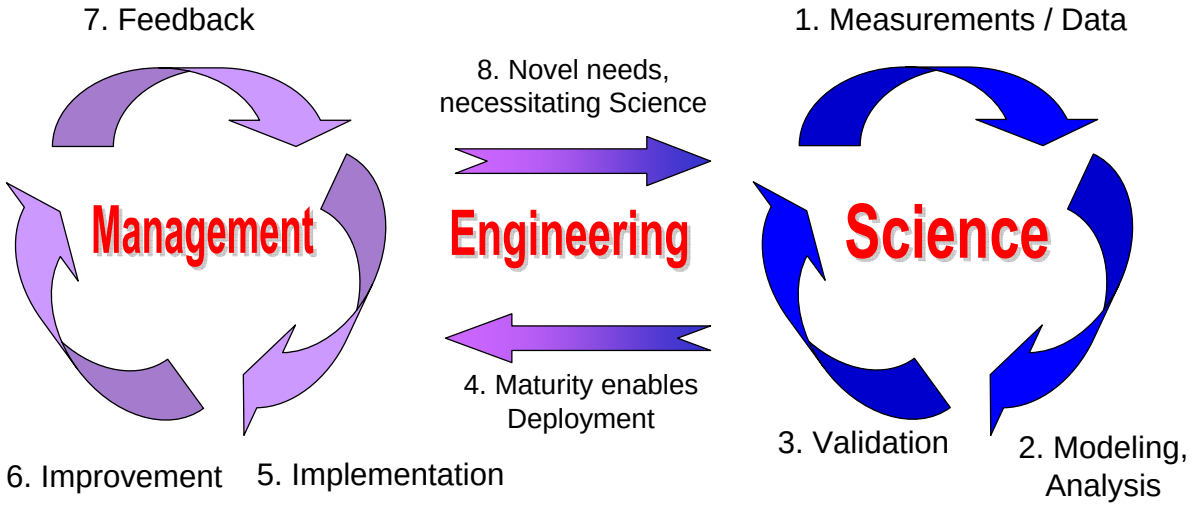


Figure 1 describes our view on the role that Service Engineering plays relative to the classical scientific cycle. This cycle starts with measurements that gather data, which is used to create models: mathematical, simulation, or sometimes merely conceptual ones. The models are then validated, which might lead to further measurements and model refinements,

continuing within this cycle until a maturity level is reached that enables deployment. Now Service Engineering transforms this Science to applications, via design. Then management takes over, implementing the new designs which, hopefully, leads to improvements. Feedback from the application often generates novel needs, which might necessitate further measurements, further models, and so on. This scientific paradigm, of measurement, modeling and validation (which is extended here to Engineering and Management), is axiomatic in natural sciences (Biology, Physics), but it is definitely a newcomer to the "Science of the Artificial", as Herbert Simon [86] called it – that is the Science of man-made systems, such as Service Systems. Being mathematicians in first-training, we were certainly not brought up with this thinking, but Service Science naturally calls for this approach, and we are thus learning.

1.5.2 Challenges and Goals

Research, teaching and practice of Service Engineering, as we perceive it, should take a *designer's view*. Design challenges pertain, for example, to the following issues.

- Service strategy: determinants of service-quality levels, full- versus self-service, customization versus standardization, warranty (after-sales support depth), etc.
- Choosing the Service Interface (channel): assume that service can be potentially performed by phone and/or by email, fax, letter, or perhaps face-to-face.
- Designing Service Process: should one use front- or back-office (or possibly both), should the tasks be performed sequentially or in parallel, etc.
- Control issues: which customers to admit to service, priority scheduling, skills-based-routing, exploiting idleness, etc.
- Operational resource management: how many servers to staff in order to reach an acceptable service level (with staffing being performed off-line or on-line, the latter as a corrective action to inaccurate forecasting), shift scheduling, etc.
- Designing the service environment: waiting experience, using busy-signal versus music in call centers, providing information to customers (eg. predicting delay durations), etc.
- Marketing: customer segmentation, cross- or up-selling, marketing-operations interfaces, etc.
- Information Systems: data-base design of call-by-call operational and business data, off-line and on-line queries, etc.

- Taking into account human factors: career paths, incentives, hiring policies, number of call center agents who are physically present at the workplace versus actual workforce level (number of agents who are actually available to take calls).

The above designer's view supports our vision of the ultimate goal of Service Engineering (both in research and teaching), as stated in Section 1.1: to develop scientifically-based design principles and tools that support and balance service quality, efficiency and profitability. We find that queueing-network models constitute a natural convenient nurturing ground for the development of such principles and tools. However, the existing supporting (Queueing) theory has been somewhat lacking, as will now be explained.

1.5.3 Scientific Perspective

The bulk of what is called Queueing Theory (recall Section 1.4) consists of research papers that formulate and analyze queueing models with realistic flavor. Most papers are knowledge-driven, where "solutions in search of a problem" are developed. Other papers are problem-driven, but most do not go far enough to a practical solution. Only relatively few articles develop theory that is either rooted in or actually settles a real-world problem, and very few carry the work as far as validating the model or the solution. In concert with this state of affairs, not much is available of what could be called *"Queueing Science"*, or perhaps the Science of Congestion, which should supplement traditional Queueing Theory with data-based models, observations and experiments. In service networks, such "Science" is lagging behind that in telecommunications, transportation, computers and manufacturing. Key reasons seem to be the difficulty to measure services (any scientific endeavor ought to start with measurements), combined with the need to incorporate human factors (which are notoriously difficult to quantify). Since reliable measurements ought to constitute a prerequisite for proper management (see TQM = Total-Quality-Management, for example), the subject of measurements and proper statistical inference is important in our context.

1.5.4 Engineering Perspective

Service networks provide a platform for advancing, what could be described as, Queueing Science and Management Engineering of Sociotechnical Systems. Management Engineering links Management Science with Management Practice, by "solving problems with existing tools in novel ways" [13]. Quoting the late Robert Herman [41], acknowledged as the "father of Transportation Science", Sociotechnical systems are to be distinguished from, say, "physical and engineering systems, as they can exhibit incredible model complexity due to human beings expressing their microgoals". (Significantly, Herman's models of complexity were nev-

ertheless "tractable through remarkable collective effects"; in other words "laws of large numbers" which, for services as well, turn out to play a central explanatory role.) The approach and terminology that we have been using, namely *Service Engineering*, is highly consistent with the once influential BPR (=Business-Process-Reengineering) evolution, as well as with ERP (=Enterprise-Resource-Planning) and CRM (= Customer-Relations-Management, or the more rational acronym Customer-Revenue-Management), placing heavy emphasis on the process-view and relying heavily on the accessibility of information technology.

1.5.5 Phenomenology, or Why Approximate

Service systems often operate over finite-time horizons (the notion of steady-state then requires re-interpretation). They employ heterogeneous servers, whose service capacities are time and state-dependent. Their customers are "intelligent", who typically (but not always) prefer short queues; they jockey, renege and, in general, react to state-changes and learn with experience. Finally, service systems suffer from high variability – both predictable and unpredictable, and diseconomies of scale – when being decentralized and inefficient (e.g., often FCFS/FIFO is the only option). Such features render the modeling of service networks a challenge and their exact analysis a rarity. This leads to research on *approximations*, typically short but also long-run fluid and diffusion approximations. Approximations also enhance exact analysis by simplifying calculations and exposing operational regimes that arise asymptotically.

The "ultimate products" of approximations are scientifically-based practically-useful rules-of-thumb. Examples of such rules-of-thumb will be demonstrated below. For example, in Section 4.7 on staffing, we cover staffing rules, customized to the QD/ED/QED operational regimes; in particular, square-root staffing corresponds to the QED regime. (Recall the discussion on these regimes in Section 1.4.)

Our approximations also reveal the "right" scale (time and space) at which phenomena evolve. For example, in ED many-server systems, waiting time is expected to be of the order of a service time. And in QED many-server systems, waiting time is expected to be one order of magnitude less than service time. To be concrete, in QED call centers, waiting is measured in seconds vs. service times in minutes; in transportation, time to seek parking in a busy city down-town is measured in minutes while parking time in hours; and in emergency departments, waiting to be hospitalized is measured in hours vs. hospitalization time in days. As another scaling example, QED call centers enjoy abandonment rates that are inversely proportional to the square-root of the number of agents; this enables one to draw performance curves, as in Figure 28, where a *single* graph predicts abandonment rates for call centers of *all* sizes.

2 Course Goals, History and Prerequisites

The Service Engineering (ServEng) course has been taught for nearly fifteen years at the Faculty of Industrial Engineering and Management (IE&M) in the Technion - Israel Institute of Technology. It started as a seminar for graduate students, entitled "Service Networks". It had then evolved through the stages of a graduate course that can be attended by advanced undergraduate students, to an elective undergraduate course, attended also by graduate students who seek an introduction to the field of Service Engineering. Currently, Service Engineering is an undergraduate *core* course, taught each semester and attended by approximately 100 students yearly. In addition, many of the IE&M graduate students in Stochastic Operations Research and Statistics take it as well. (In general, gradual transition from an elective to a core course seems to us appropriate when a new course in the field is designed.) The ServEng website [83] is designed to contain all course materials (lecture notes/slides, recitations, homework). This website is supplemented by related research papers, slides of seminars, software, databases and more.

Prerequisites for our course include core undergraduate courses in calculus, probability theory, statistics, stochastic processes and optimization (linear programming). Students of our department usually take all these courses during their first two years of studies.

As discussed in Section 1.1, the service sector is dominating the economics of developed countries. Still, prior to the launch of the ServEng course, Technion IE&M students had been exposed mainly to methods and techniques inspired by manufacturing applications. (This situation seems to prevail also among IE departments at other universities.) The ServEng course aims at filling this gap by providing students with appropriate models and tools for design, operation and analysis of service systems. Our teaching approach is data-oriented: examples from various service sectors are presented at lectures, recitations and homework assignments, with the call center industry being the central (but in no way the only) application area.

In one and a half decades of course development, our service system data repositories have been developed in parallel with the theoretical material of the course. In the next section, we describe the service system data that supports the course.

3 Data - a Prerequisite for Research and Teaching

We strongly believe that systematic measurements and data collection are prerequisites for the analysis and management of any service system. Therefore, our students encounter numerous service data sets and examples during lectures and recitations. Most of the course homework is also based on actual service data.

Early generations of the course used one-month tellers' data from a bank in Israel, in support of recitations and homework. As described in [64], the data was collected through bar-codes that were given to customers upon arrivals, which were then scanned at milestones of the service experience. Then the focus of the course shifted in the direction of call centers, which seemed to be a natural candidate for becoming our main application area. First, one could not but be impressed by the sheer magnitude of the call center industry, with its explosive growth over the last three decades. Millions of agents are employed in call centers over the world and, according to some estimates, worldwide expenditure on call centers exceeds $300 billion per year. Second, the call center environment gives rise to numerous managerial and engineering challenges that vary in their nature and time-scale, from real-time skill-based routing calls to long-term design and workforce planning. And thirdly, large call centers generate vast amounts of data that is relatively easily accessible. (For example, hospital data, typically dispersed in a number of different information systems, is more complicated to access.) A detailed history of each call that enters the system can, in theory, be reconstructed via the Automatic Call Distributor (ACD) and Interactive Voice Response Units (IVR). However, call centers have not typically stored or analyzed this data, using instead the ACD reports that summarize performance over certain time intervals (say, 30 minutes). In our research [10] and teaching, we advocate the change of this approach and emphasize the practical and research advantages of call-by-call data analysis.

In the ServEng course, our first call-center data-base covered a one-year operation of a call center at a small Israeli bank [10, 61]: about 350,000 calls, catered by about 15 agents. Once this database was incorporated into the course, the bank tellers' data (face-to-face services) has been since used exclusively in recitations, to teach or demonstrate techniques, while the telephone data has been used in homework, in order to practice these techniques.

Our small Israeli database clearly revealed the potential of incorporating real-world data into ServEng teaching and research. It hence paved the way to access vast amounts of call-by-call data, from large call centers who were willing, sometimes even eager, to participate in our data venture. But then several challenges arose. First, call center data is processed by vendor-specific programs, in formats (data-structures) that are typically not amenable to operational analysis. Second, a database of a large call center consists of up to hundreds of millions of records. Hence, brute force statistical processing of the records would take a prohibitively long time even for the most powerful statistical software. And thirdly, our experience is that a significant part of real-world databases is contaminated or inconsistent with the rest, which calls for an extensive cleaning and data-improvement effort prior to using the data.

As one anecdotal example, the first large US Bank that provided us with call center data

had four interconnected call centers. When switching to daylight saving time, one of these call centers failed to update its computer clock and thus, calls that were dialed in City X arrived to City Y one hour earlier! The database from this U.S. Bank, which is described in [89], was in fact cleaned and has recently been incorporated into the ServEng course, as will be described in the next subsection. (Just for comparison sake, U.S. Bank employed around 1000 call center agents, who catered to about 350,000 calls per *week* - which was the yearly volume of our small Israeli bank.)

3.1 DataMOCCA and SEEStat - An Environment for Online EDA (Exploratory Data Analysis)

Data-based research and teaching give rise to numerous data-related challenges. These range from partner-identification and data acquisition through cleaning to maintenance and analysis. To address the challenges presented by service data, a research laboratory was established in 2007 at the Technion: SEELab, where SEE stands for "Service Enterprise Engineering" [84]. The SEELab has been funded by a private donation to the Technion - indeed, research agencies are unlikely to support its activities as these are not considered basic research (but rather only supporting the latter). Under the dual leadership of Dr. Valery Trofimov (technical) and AM (academic), the SEELab *receives, processes, cleans, validates and maintains* a repository of databases from service enterprises. To this end, the SEELab developed DataMOCCA (Data Models for Call Centers Analysis) [87], which is a software suite that renders the data ready for support of research and teaching.

DataMOCCA offers a *universal* model for operational call center data, or, more generally, transaction-based services. (Universality is in the sense of being source-independent, be it a call center or an emergency department or an internet site.) The main user interface is within a (flexible and friendly) graphical environment, SEEStat, which enables real-time Exploratory Data Analysis (EDA), at second-to-month resolutions. As part of the EDA, SEEStat includes statistical algorithms such as parametric distribution fitting and selection, fitting of distribution mixtures, survival analysis (mainly for estimating customers' (im)patience), and more - all these algorithms interact smoothly with all the databases.

The SEElab databases are designed and maintained at two levels: the basic level is close to the raw data, after cleaning, validating and processing; the upper level consists of precompiled summary tables, which are created only once (and then only occasionally modified), and their design is efficient enough to support real-time processing, with few-seconds response time, covering gigabytes-large databases. This provides the infrastructure for a convenient EDA environment, which accommodates real-time statistical analysis and simulations.

Currently, SEEStat interacts with call-by-call data from four large call centers: a U.S. bank, two Israeli banks and an Israeli cellular-phone company. Three of the four databases cover periods of 2-3 years; the fourth one, which is presently the most active, has about 1.5 years data-worth where, during the last 6 months, call center data has been deposited on a *daily basis* at a data-safe within the SEELab.

The U.S. bank database is universally accessible (via the Internet) at the SEELab's server (more on that momentarily). This database has close to 220 million calls, out of which about 40 million were served by agents and the rest by a VRU (Voice Response Unit, or simply put an answering machine). DataMOCCA and SEEStat have well-served our ServEng course. Indeed, a multitude of examples, using SEEStat, are presented in classes and recitations. Moreover, one of the homework assignments is SEEStat-dedicated: it allows the students hands-on experience with the U.S. Bank data, mentioned above, having them analyze it in interesting insightful ways. Concrete SEEStat applications from the ServEng course are described in Section 4 below. Further details, in particular of the SEEStat-based homework, are provided in Sections 4 and 5.

While originally designed for operational databases from call centers (hence its name), DataMOCCA has been expanded to accommodate additional sources and types of data. Specifically, the SEELab now "owns" data also from several hospitals (mostly their emergency departments), from an internet website (click-stream data), and a couple of samples from face-to-face services. We are further preparing to augment our operational databases with financial/marketing and contents data. (In an emergency department, contents refers to clinical data.) In addition, two simulators have been written, one for a call center and the other for an emergency department. The "vision" is to connect these simulators to the real data of the SEELab, and then have these SEELab resources (data and simulators) readily accessible worldwide, for use by students and researchers of Service Engineering and its supporting Service Science.

SEEStat Online: As already mentioned, data from two banks are universally accessible through SEEStat at the SEELab's server. The connection procedure, for any research or teaching purpose, is simply as follows: go to the SEELab webpage
http://ie.technion.ac.il/Labs/Serveng;
then, either via the link **SEEStat Online**, or directly through
http://seeserver.iem.technion.ac.il/see-terminal,
complete the registration procedure. Within a day or so, you will receive a confirmation of your registration, plus a password that allows you access to the EDA environment of SEEStat, and thus to the above mentioned databases. Note that your confirmation email includes two attachments: a trouble-shooting document and a self-taught tutorial. We

propose that you print out the tutorial, connect to SEEStat and then let the tutorial guide you, hands-on, through SEEStat basics - this should take no more than 1 hour. (For teaching purposes, one of the recitations at the course site contains a SEEStat quick guide, which has been used to introduce it to students.)

4 Course Syllabus: Theory, Examples, Case Studies

The ServEng course consists of roughly four parts:

1. *Prerequisites:* measurements and models.
2. *Building Blocks:* demand, services, customers (im)patience;
3. *Models:* deterministic (Fluid) and stochastic - mainly queueing models, both conventional (Markovian) and approximations;
4. *Applications:* design, workforce-management (e.g. staffing) and skills-based routing.

We now provide a brief description of these four parts. Then, throughout the section, we elaborate on them, lecture by lecture.

Measurements, at the granularity level of individual service transactions, are prerequisites for the design, analysis and management of service systems. Thus, after opening the course with an introduction to Service Engineering, we survey transactional measurement systems in face-to-face, telephone, internet and transportation systems. We then proceed with an introduction to *Modeling*, using Dynamic Stochastic PERT/CPM models (also called fork-join or split-match networks) as our modeling framework. These models capture operational congestion that is due to resource constraints and synchronization gaps.

Measurements and modeling prerequisites directly give rise to deterministic (fluid/flow) models of a service station, which capture average behavior and enable relatively simply yet far-reaching analysis - for example capacity (bottleneck) analysis.

The next course segment is dedicated to the three building blocks of a basic service-model: demand, service and (im)patience. First we study *service demand*, emphasizing the importance of reliable forecasting techniques. (In particular, our model for exogenous customers' arrivals is a Poisson process, or a relative). Then we analyze the *service process*, describing its operational characteristics: service-duration, which is a static characteristics, and process structure, capturing dynamic evolution. Service durations in call centers often turn out log-normally distributed [10] (at the resolution of seconds, but sometimes exponential at minutes-resolution). The structure of the service process is naturally captured by phase-type distributions. We end with *customers' patience*, or perhaps *impatience*, and its

manifestation - the abandonment phenomena, which is important in call centers and other services (e.g. Internet and even Emergency Rooms).

The three building blocks, of arrivals, services and (im)patience, are fused into basic queueing models where customers are iid and servers are also iid. A central role is played by Markovian Queues, underscoring the applicability of the Erlang-A queue in the call center industry [66]. Then we discuss design principles (pooling to exploit economies of scale) and present operational workforce management techniques (staffing and scheduling), including staffing in the QED, ED and QD operational regimes. We conclude the course with models that acknowledge customers differentiation (priorities) and servers heterogeneity/skills (SBR = Skills-Based-Routing). An optional last lecture surveys queueing networks, specifically Jackson and Generalized Jackson, as models of multi-stage service systems.

4.1 Course Material and Supporting Texts

ServEng is a one-semester course. Our standard Israeli semester consists of 14 weeks, three hour of lectures per week, accompanied by a weekly one hour recitation. At every lecture and recitation, students get lecture notes (copies of the slides) which are used in class. Typically, these notes are supplemented with board lectures, providing an introduction to a sub-subject or clarifying subtle issues. (The notes most often constitute a superset of the class material, and the actual material covered varies.) All lecture notes appear in the ServEng website [83], under the Lectures and Recitations menus.

Israeli students, unlike U.S. students, are not accustomed to using text-books for self-study - they rely on class material, hence the lecture notes play a central role in the course. Yet, there are three books that are mentioned as relevant and useful, and our library has ample copies of these books for students' use. These books are mentioned at the outset of the course, and students are asked to read few chapters from these books. The books are by Randolph Hall [35], J. Fitzsimmons and M. Fitzsimmons [23] and C. Lovelock [54] (in that order of relevance to the course), which we now elaborate on.

Our teaching philosophy was influenced by Hall's book [35] on Queueing Methods - this book is rather unique among Queueing books as it discusses measurements, gives fluid models the respect they deserve and emphasizes science-based applications. Hall's book serves as an optional textbook, which students are encouraged to consult, especially during the first half of the course. Placing the course textbooks in perspective, Hall's book lies at the intersection of Operations Research and Industrial Engineering; Fitzsimmons and Fitzsimmons [23] borders on Industrial Engineering and Operations Management; and Lovelock [54] touches on the four extreme points that span Service Engineering: Operations, Marketing, Human Resource Management and Information Systems.

Occasionally in our course, we have used a number of business cases, either within course material or as final course projects. An example of course material is the National Cranberry Cooperative case [79, 80, 81], used within Fluid Models. Two final-project examples are Harrison's Manzana Insurance case [37], and Federgruen's Vonage case [20]. (We adapted the latter to our course needs by changing its base model from Erlang-C to Erlang-A.)

We now proceed with a detailed survey of our lectures, providing ample real-life examples, in line with our paradigm of data-based teaching.

4.2 Measurements and Models; Little's Law

In the first lecture of the course, we introduce students to the Service Economy and to the discipline of Service Engineering. We also discuss course logistics and "rules of the game", for example the relative weights of homework (40-50%) and final exam (60-50%) in the final grade.

The subsequent two lectures are dedicated to *measurements* and *models* - these, in our opinion, are the prerequisites for any advances and practice of Engineering and Science, in particular to the discipline that we are teaching.

4.2.1 Data and Measurements

As already mentioned, we believe that research and teaching of Service Engineering must be based on *measurements*: real-world data, collected in various service systems at various levels of granularity. All stages of modeling and design of service systems must be empirically supported, starting with systematic measurements and data collection at the initial stage of modeling and ending with the validation of both models and managerial decisions against empirical data.

Call Centers provides an excellent illustration for our approach. As it operates, a large call center generates vast amounts of data. Its Interactive Voice Response (IVR) units and Automatic Call Distributor (ACD) are special-purpose computers that use data to mediate the flow of calls. Each time one of these computers takes an action, it records the call's identification number, the action taken, the elapsed time since the previous action, as well as additional pieces of information. As a call winds its way through a call center, a large number of these records may be generated. From these records, a detailed *operational history* of each call that enters the system can, in theory, be reconstructed: its arrival time; who was the caller; the actions the caller took at the IVR and the duration of each action; whether and how long the caller waited in queue; whether and for how long did an agent serve the call; who was the agent; was there some agents's additional work associated with the call after service completions, and more.

Large call centers are typically also equipped with Computer Telephony Integration (CTI) - the capability of integrating, upon call arrival, the caller identity (from the Telephone) with the callers's company history (from the company's Computers). Then additional data - *contents* and *financial* - become available from the company's information systems: call subject/reason; types of actions taken by the agent; related account history; the economic worth of the call, and more. There are clear analogues of these types of transactional data - operational, contents and financial - in other service systems. We demonstrate such data from face-to-face, internet, transportation and healthcare services. Taking the latter, as an example: in a hospital, operational data records the flow history of a patient, from being admitted say at the emergency department, through transfer to a hospital ward, until the ultimate release; contents data is replaces by clinical data; and financial data retains its meaning.

In practice, however, service systems in general, and call centers in particular, have not conscientiously stored, let alone analyze, records of individual calls (transactional data). This is attributed, in part, to the historically high costs of maintaining large databases (a large call center generates gigabytes of call-by-call data per day), though such quantities of data are no longer prohibitively expensive to maintain. It is also due to the fact that the software supporting management of services (itself developed at the time when data storage was expensive) often uses only simple data-models which require limited summary statistics. But the most important reason for this lack of or inaccessibility to transactional data is due to managers' lack of understanding and appreciation for the value of its analysis.

It comes at no surprise, therefore, that call centers most often describe their performance in terms of *averages*, that are calculated over short time intervals (15, 30 or 60 minutes in length, with 30 being the most common). These ACD aggregated data are then used both for planning purposes and performance evaluation.

In our teaching, research and consulting, we are attempting to help change this state of affairs. Indeed, we encourage the theme that "average do not tell the whole story" and, instead, we advocate the use of individual-transaction level (call-by-call) data through demonstrating its benefits. To this end, DataMOCCA databases, as described in Section 3.1, provide an effective infrastructure for our approach.

4.2.2 Models

The second prerequisite for Service Engineering is *modeling*. Indeed, a major part of the course is dedicated to different models of service systems, their analysis and areas of application. We distinguish between *empirical* and *analytical* models. (Presently, the course utilizes no *simulation* models, though this is going to change in the near future.) Empirical models,

which are directly data-based, are further classified into conceptual, descriptive and explanatory models. Analytical (or mathematical) models are either deterministic (fluid-based) or stochastic (e.g. Markov chains). All models could be either static (long-run, steady-state) or dynamic (transient).

The first models that students encounter are *empirical models*: for our purposes, these are nothing but insightful depictions of data, from the operational view-point. (See the description of Homework 3 in Section 5.) In parallel, students learn definitions and calculations (empirical, theoretical) of operational characteristics and performance: capacity (static and dynamic), bottlenecks, waiting times, queue-lengths and throughput rates.

No attempt is made at the course to well-define the above mentioned model classes - these are used merely to structure one's thinking about modeling, and we "explain" them mainly through case studies. For example, some fluid models are in fact empirical models, as will be later clarified through Figure 12. And analytical models could overlap with empirical models, as will be illustrated momentarily via the simple yet fundamental Little's Law [51].

Little's Law is a conservation law for a black-box model of a service system, relating the following three performance measures:

- λ - *throughput rate*, which is the rate at which customers (service units) flow through (arrive to, depart from) the system;
- L - *customer count*, which is the number of customers in the system (Work-In-Process);
- W - *sojourn time*, which is the time that customers spend within the system.

In its most widely used version, Little's Law relates the above measures as being steady-state or long-run averages, postulating that

$$L = \lambda \times W. \tag{1}$$

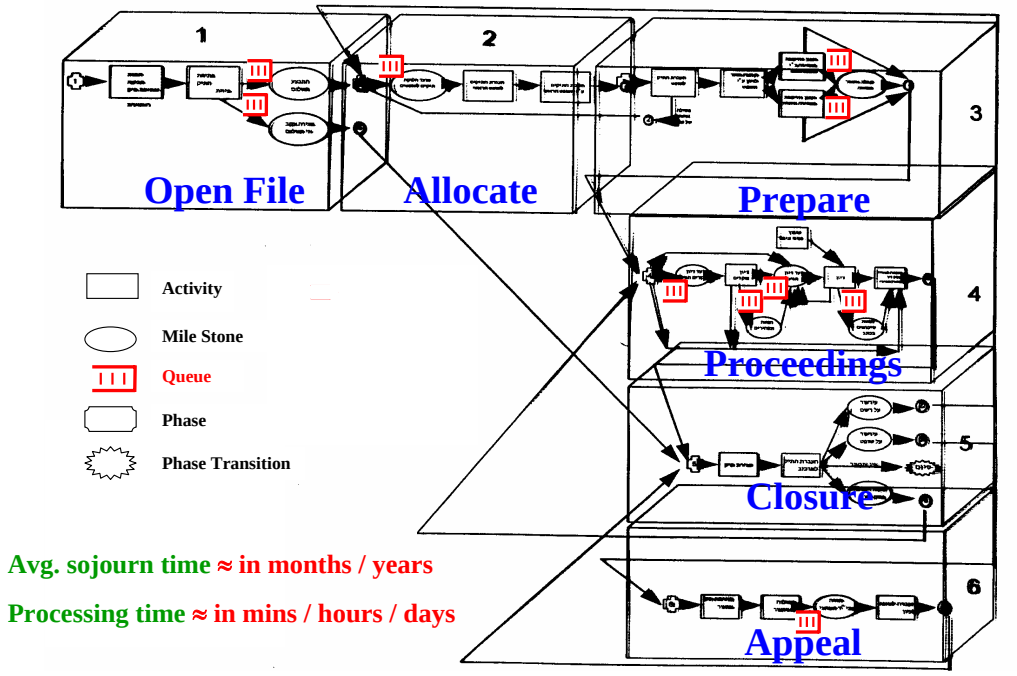
The law actually binds together the three main "players" that constitute a service system: the *customer*, associated with W, which is a measure for operational service quality; the *service-provider*, concerned with λ, which reflects the server's service effort; and the *manager*, controlling L, which is a visible proxy for congestion. (The three players are bound via two degrees of freedom.)

The reality of service systems calls for versions and generalizations of Little's Law that go beyond steady-state/long-run. Specifically, service-systems often operate in *cycles* (periods), for which there is also a per-period version (same law, but calculations of its three constituents are performed over a finite-horizon). And most service-systems "enjoy" *time-varying* dynamics (e.g. arrival rate at a peak hour that is twice the daily hourly average),

which requires a transient, less known but no less important, version. In view of its importance and relevance, we shall describe this transient version later, in Subsection 4.2.5.

4.2.3 Addressing the Skeptic: Is Service Engineering Relevant?

Figure 2: **"Production of Justice" in the Labor Court of Law**



It is not unreasonable to question whether the methods of "exact sciences" (eg. Mathematics), and their related models (Statistical, Operations Research), are relevant for supporting the operations of the legal system. Indeed, Flanders [24], who is what we have been calling a *smart influential skeptic*, gives a negative answer to this question. Common reasons that support Flanders' view are that legal processes are very complex, in fact beyond quantification and modeling. (Figure 2 displays an operational flowchart of "cases" in the Haifa labor-court of law, taken from an undergraduate project.)

A second problem is that of incentives: in a typical service system, one expects that customers and servers share a common interest of reducing waits; in the legal system, on the other hand, the accused side is often stalling the process, in order to postpone, as much as possible, a fine or imprisonment. And a third reason is that justice-professionals (e.g. judges) tend to view their work as "art", contrasting it with the other extreme view of a system 'that 'produces justice".

Justice professionals are no different than most other professionals, be it doctors or professors or .... For example, as was true for Flanders [24], there is also much resistance to the

Service Engineering approach within the very significant health-care world. To wit, citing a recent paper of Kopach-Konrad et al. [48] "Very few health care providers are trained to think analytically about how health care delivery should function. Thus, it is often difficult for these professionals to appreciate the contributions that systems engineering approaches can bring. Conversely, engineering professionals often have little, if any, education in health care delivery." We are trying to close this gap from both sides. First, together with our students, who work on healthcare projects, we are gradually accumulating practical knowledge about healthcare processes. And conversely, physicians and administrators from the largest hospital in northern Israel have been cooperating and learning from these projects, some to the level of eventually fully attending our Service Engineering course.

Flanders [24] has been made a reading assignment, which is discussed in one of the first classes. It provides excellent reading for the Service Engineer, who is challenged there with issues, articulated in the context of the Justice System but as prevalent and central elsewhere, such as: the tradeoff between model simplicity and complexity; who are the customers? the servers? what are the costs of delays? what are some alternatives to model-based analysis of the "Production Process of Justice"?, and more.

In their day-to-day practice, Service engineers must identify the "Flanders's" of service systems, viewing them as an opportunity - a well-articulated influential skeptic (opposition) who, if convinced of the virtues and relevance of Service Engineering, would turn into exactly the type of ally that the discipline seeks.

Interestingly, Flanders demonstrates unfamiliarity with Little's Law. Indeed, he proposes λ and L as operational performance measures, being convenient to measure (bottom of page 316 and top of 317); yet he notes that W would have been more appropriate (Note 5. at paper's end), without realizing that that $W = L/\lambda$. We thus believe that the following case study of Little's Law, applied to the "Production of Justice", would have certainly helped a Service Engineer alter the views of any "Flanders", in favor of the profession.

We now take a data-based approach to demonstrate the steady-state version of Little's Law (in an environment where Service Engineering is not commonly applied, though it should be - the Justice System). This is followed by a brief exposition of the time-varying version of Little's Law, and its role as measuring offered load.

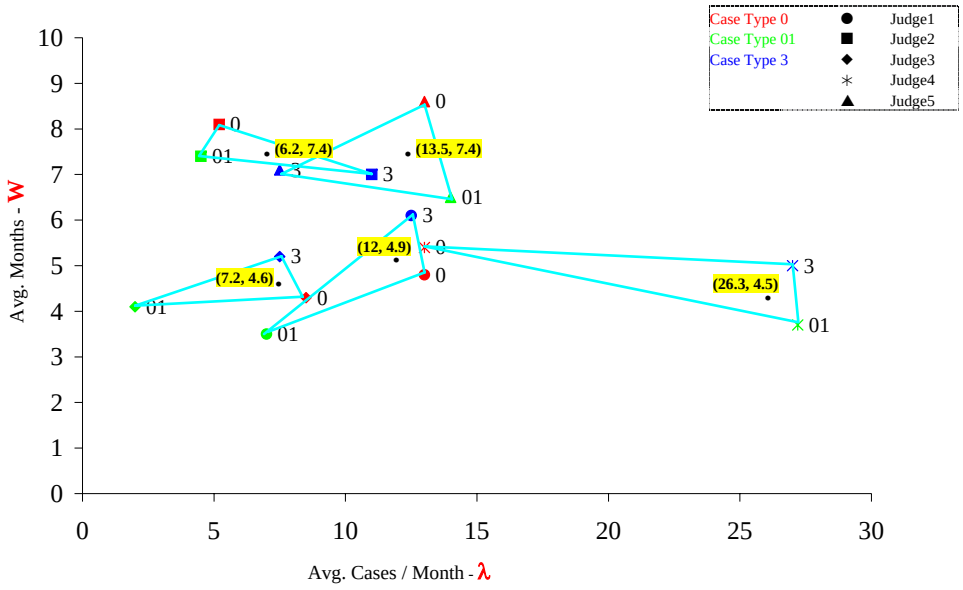
4.2.4 Case Study - Little's Law and The "Production" of Justice: Simple Models at the Service of Complex Realities

Figure 3 exhibits the operational performance of five judges in Haifa's labor court. Two performance measures are displayed: the average number of cases completed per month, λ (x-axis), and the average handling time of a case, W (y-axis). Each of the five judges

attends to the same three types of cases. Hence each judges's performance is represented by a triangle, with a corner corresponding to a pair (λ, W) of a particular case. Average operational performance of a judge (the barycenter of the corresponding triangle) is marked in yellow.

"Good performance" (operationally, as opposed, for example, to being deep, accurate, polite) manifests itself at the south-east area of Figure 3. Thus, Judge 4 (denoted $*$) is clearly the *best* performer, providing the shortest handling time *and* completing the most cases per month.

Figure 3: **Performance of Judges: Throughput Rate and Processing Time**



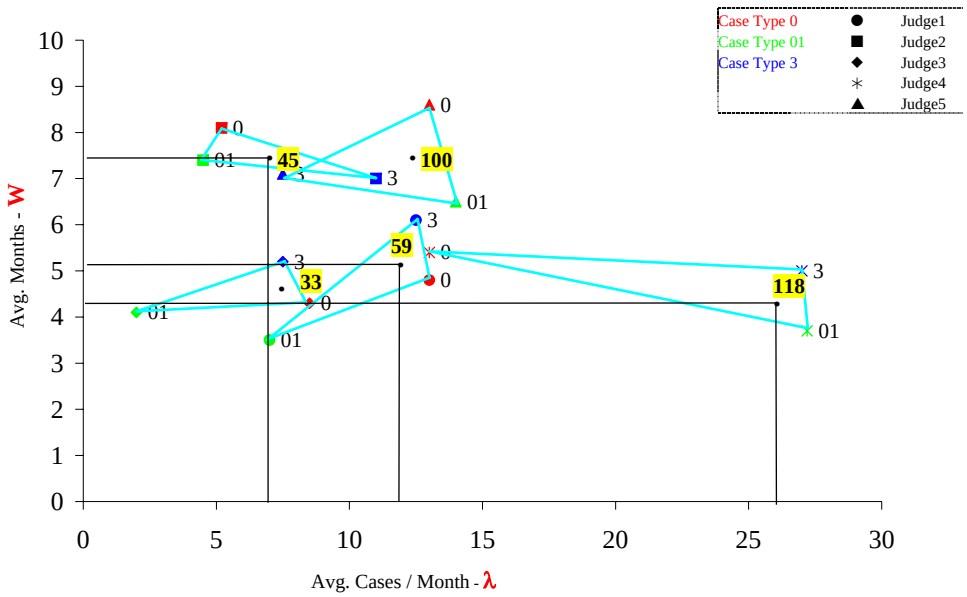
However, the prevalent operational performance for a judge is the number of pending cases, namely the *backlog* L, which is based on the premise [24] that a larger backlog corresponds to worse performance. Now Figure 3 does not directly exhibit statistics for average backlog. However, one can easily calculate the backlog via Little's Law $L = \lambda \cdot W$ which, for each judge, is the areas of the corresponding rectangles in Figure 4. One observes that now Judge 4/$*$ has turned into the *worst* performer, having the highest number of pending cases. But we already know that the operational performance of this judge is the best, leading one to deduce that the prevalent measure of average backlog is inadequate for measuring the operational performance of judges.

4.2.5 Little's Law for Time-Varying Systems

Towards the end of the course (see Section 4.7.3), we teach staffing algorithms for service systems that face time-varying arrival rates (which excludes little else). These algorithms

are based on the central notion of *offered load* which, in turn, amounts to a version of Little's Law in a *time-varying* environment. We now introduce this time-varying version, followed by a brief discussion of the offered load and the relation of the two.

Figure 4: **Performance of Judges: Application of Little's Law**



Assume that customers arrive to a service system according to the time-varying rate $\lambda(t)$. (For simplicity of notation, assume that time $t \in (-\infty, \infty)$.) Denote the (random) sojourn time of *all* customers within the system by W (implicitly assuming here that all customers follow this common distribution - which can easily be relaxed to allow for sojourn times that depend on the time of arrival). Let $L(t)$ stand for the average number of customers in the system at time t. Then $L(t)$ is given by

$$L(t) = \mathrm{E} \int_{-\infty}^{t} \lambda(u)\ \mathrm{P}\{W > t-u\}\ du\ , \quad -\infty < t < \infty, \tag{2}$$

which, in turn, equals any of the following three equivalent representations:

$$\begin{aligned} L(t) &= \mathrm{E}[A(t) - A(t-W)] = \mathrm{E}\left[\int_{t-W}^{t} \lambda(u)\ du\right] \\ &= \mathrm{E}[\lambda(t-W_e)] \cdot \mathrm{E}[W]\,, \end{aligned}$$

at all times $t \in (-\infty, \infty)$; here $A(t)$ is the cumulative number of arrivals up to time t, and W_e has the so-called excess life-time distribution associated with W:

$$P(W_e \le t) = \frac{1}{E(W)} \int_0^t \mathrm{P}\{W > u\}\, du, \quad t \ge 0. \tag{3}$$

Note that when the arrival rate does not vary with time, $\lambda(t) \equiv \lambda$, then the last two representations of $L(t)$ immediately reduce to Little's Law in (1).

Each of the four representations of $L(t)$ has an insightful interpretation. The first one, in (2), follows from a "splitting" argument: $\lambda(u) \cdot du$ is (approximately) the average number of arrivals during $(u, u + du]$; out of these, a fraction of $\mathrm{P}\{W > t - u\}$ are still within the system at time t; and now "sum" over all $u \in (-\infty, t]$ to get the average number of customers present at time t. The first representation is well-known in the context of the $M_t/G/\infty$ queue [17]. It was extended to general time-varying systems in [8] and [27].

The second representation says that those present at time t are exactly those arriving during $(t-W, t]$, which immediately gives rise to the third representation. We have no direct intuitive explanation for the fourth representation. In some sense, taken together with the third representation, it amounts to a mean-value result: $\mathrm{E}[\lambda(t - W_e)] = \mathrm{E}\left[\int_{t-W}^t \lambda(u)du\right] \;/\; E[W]$. It is also the closest to the original Little's Law in (1), in which λ is replaced by the *time-lagged* $\mathrm{E}[\lambda(t - S_e)]$. An insightful calculation of this last representation, which we do not reproduce here, is by verifying it first for a constant W (exactly the mean-value theorem, since W_e is then uniformly distributed), and next extending the constant case to a random W taking a finite number of values.

None of the four representations is amenable to easy calculations in terms of model primitives. To this end, one resorts to approximations, based on the last representation and Taylor's expansion of $\lambda(\cdot)$. These will be described momentarily, in the following subsection.

4.2.6 The Offered-Load

Consider a service system in steady-state. Assume that customers arrive at a constant rate λ, and all seek service of a (random) duration S. The *arrival-rate of work* to the system is then

$$R = \lambda \cdot \mathrm{E}[S]$$

units of work per unit of time, where work is measured in units of time. (For example, 100 phone calls per minute, with an average of 3 minutes per call, generate 300 minutes of work per minute; or 300 Erlangs, as expressed in telecommunication.) We call R the *offered-load* to the system. The relation of R to system's capacity determines the system's operational performance. Thus, calculating R is the first step in dimensioning the capacity of a service system.

What if the arrival rate λ varies in time? A hint at an answer is the observation that, with arrival at a constant rate, R above is in fact the average number of *busy* servers in a corresponding system with an *infinite* number of servers, say $M/G/\infty$ for concreteness. Analogously, if customer arrivals is time-varying Poisson at rate $\lambda(t)$, and arrivals seek service that is distributed as S, the time-varying offered-load $R(t)$ is then the average number of busy-server at the corresponding $M_t/G/\infty$ system. The latter is given precisely by the time-varying Little's Law (2), in which W is replaced by S.

Conceptualizing the workload offered to a service station, in terms of a corresponding infinite-server system, is natural and far-reaching. This is an important step in the education of a Service Engineer, especially since this view is scarcely appreciated in practice. (For example, instead of forecasting the *number* of arrivals to a service system, be it a call center or an emergency department, we advocate the forecast the offered-load for the system's resources, which combines (in a non-trivial manner) arrival counts with service times.) To this end, it is important to have easily-calculated expressions for $R(t)$, in terms of systems primitives (the arrival rate and the distribution of S).

If $\lambda(t)$ is a polynomial, then $R(t)$ can be expressed directly in terms of moments of S_e (See Theorem 10 in [17]). Otherwise, one could use a polynomial approximations, namely a Taylor expansion. Specifically, let $\lambda^{(k)}(t)$ denotes the k^{th} derivative of $\lambda(\cdot)$, evaluated at time t. Then, the first- and second-order Taylor expansions, with the latter being

$$\lambda(t-u) \approx \lambda(t) - \lambda^{(1)}(t) \cdot u + \lambda^{(2)}(t) \cdot \frac{u^2}{2},$$

yield a first- and second-order approximations for the offered-load at time t:

1. $R(t) \approx \lambda(t - E[S_e]) \cdot E[S]$;
2. $R(t) \approx \lambda(t - E[S_e]) \cdot E[S] + \dfrac{\lambda^{(2)}(t)}{2} \cdot Var(S_e) \cdot E(S)$.

The first-order approximation is often referred to as the (time) *lagged*-PSA (PSA = Piecewise Stationary Approximation), with $E[S_e]$ being a proxy for the time-lag. It is especially important in environments such as Emergency Departments, where sojourn time are long; there, the offered-load on beds in an Emergency Department at time t is heavily affected by arrivals that took place long before t. (To demonstrate this point, we use the data in [52], analyzing the phenomenon of ambulance diversion .) From the second order approximation, we notice that there is also a space-shift by $\dfrac{\lambda^{(2)}(t)}{2} \cdot Var(S_e) \cdot E(S)$, depending on the curvature of the arrival rate. In particular, since $\lambda^{(2)}(t)$ is negative at t that is a local peak of $\lambda(\cdot)$, the offered-load at times of peak demand are shifted down (relative to the first-order approximations).

Being the backbone of time-varying staffing (which is taught at the end of the course; see Section 4.7.3), the above behavioral insights on $R(t)$ are practically significant. ([32] is recommended for further discussion and references.) In fact, the offered-load concept extends far beyond a single basic service station, which is presently the subject of active research at the Technion.

4.3 Dynamic Stochastic Service Networks

In Lecture 4, we formally introduce Service Networks (or processing networks), which were described previously in Section 1.3. These provide us with a framework for teaching the main drivers of operational delays: *scarce resources* and *synchronization gaps* (to be distinguished from other causes, beyond the control of the Service Engineer, for example bad weather in an airport).

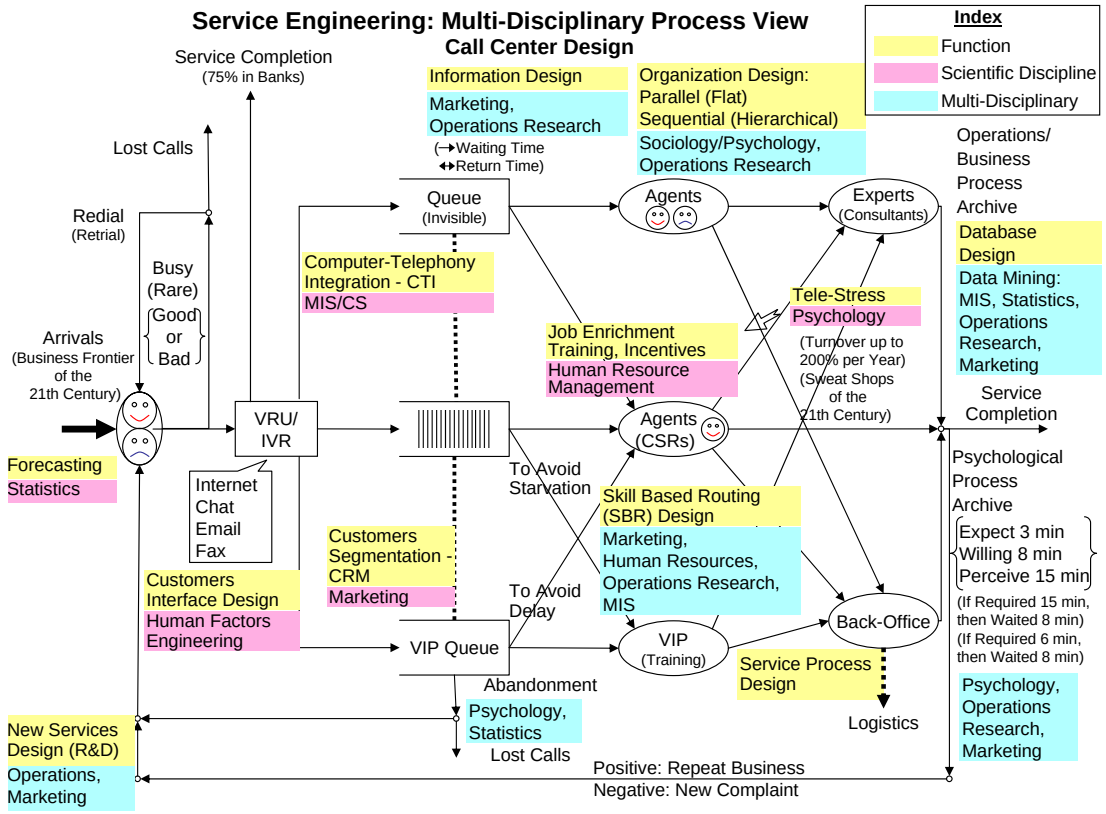
We actually start with loosely-defined conceptual models of service networks, made concrete through Call Centers and Emergency Departments - see Figures 5 and 6.

These conceptual models demonstrate (at least) two important points. The first point is the power of the "language of modeling", in that two such distinct systems, call centers and emergency departments, can be commonly described. The second point is the role of the Service Engineer as an *integrator* of several disciplines (recall [28]). Indeed, in the above two figures, activities of the Service Engineer, to be carried out at the Call Center (Emergency Department), are marked `yellow`; below the activity appears the scientific discipline(s) that support it - in `pink` if there is a single such discipline, and in `blue` if the support is to be multi-disciplinary. (For example, Skills-Based Routing (SBR) design and management requires support from Operations Research - providing the routing algorithms (see Section 4.8); Marketing - for segmenting customers; Management of Human Resources - for designing skills of agents and training them; and Management Information System (MIS) - for the infrastructure of SBR.)

At the Technion, the disciplines that appear in Figures 5 and 6 are *all* uniquely taught under the "single roof" of our Faculty of Industrial Engineering and Management. Hence, in recent versions of the course, we have actually been using these two figures already at our first class, which helps in making concrete our introduction to the Service Engineering profession, and in stimulating our students' (the future Service Engineers) esprit de corps. (It takes about 40 minutes to cover these figures.)

Figures 5 - 6 are too crude for capturing operational subtleties in design and control, for example the two triggers of operational delay: first, waiting in *resource queues*, due to scarce resources (e.g. an telephone-agent of a physician), which is prevalent in both worlds; and second, *synchronization queues*, for example, waiting for join of a physician, a blood

Figure 5: **Call-Center Network - Conceptual Model**



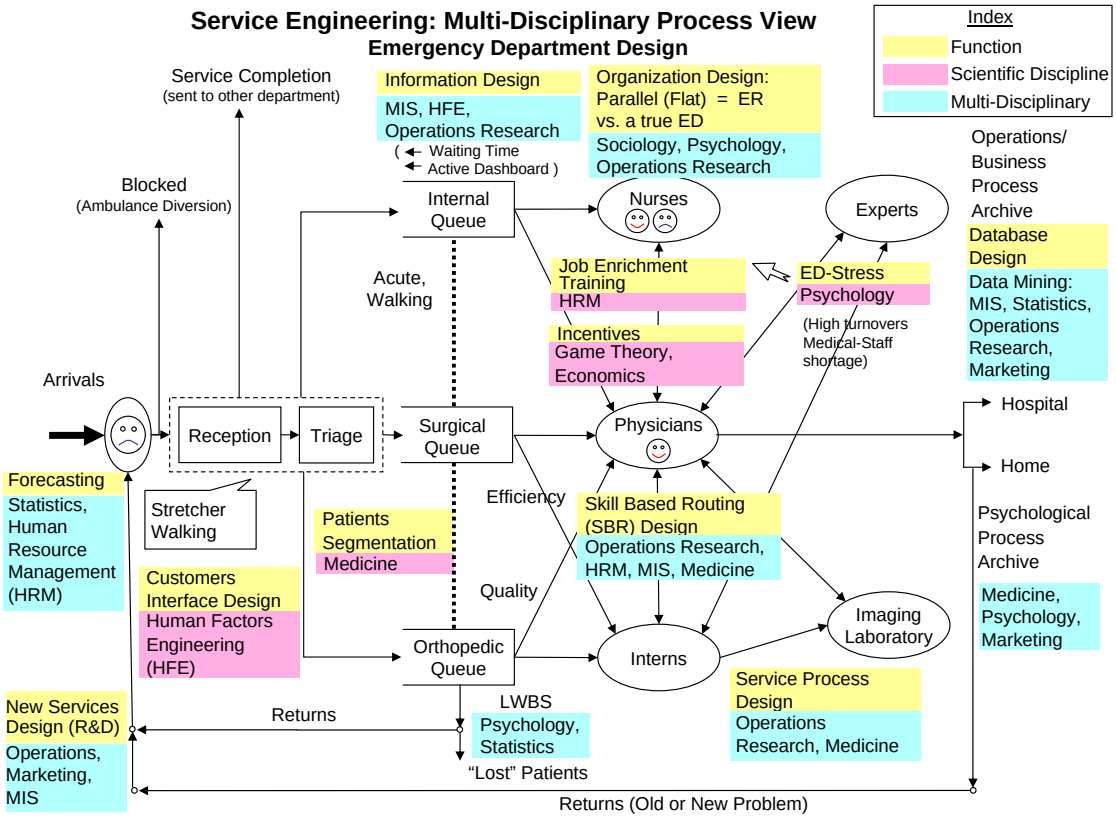
test and an MRI. We thus simplify (specialize) these conceptual models to, what we have been calling, Dynamic-Stochastic PERT/CMP Networks: DS-PERTs for short, or Fork-Join (Split-Match) networks as they are technically referred to.

4.3.1 DS-PERTs (Fork-Join or Split-Match Networks)

Class discussion about DS-PERT networks (DS-PERTs) is structured through a teaching note that gradually introduces Static-Deterministic PERT (the classical OR models), Static-Stochastic PERT (adding randomness to activity durations), and finally Dynamic Stochastic PERT networks (acknowledging scarce resources). In this note, which is based on the Technion M.Sc. thesis [7], we analyze DS-PERTs in four stages, as articulated in 1.3.2.

More specifically, first we review the classical OR/IE technique of project management: PERT (Project Evaluation and Review Technique), or CPM (Critical Path Method). Here project *activities* (*tasks*) constitute, for example, the service-process of a customer (e.g. within an emergency department). The activities are assumed to have constant (deterministic) durations, they must adhere to precedence constraints (formally encapsulated by a

Figure 6: **Emergency-Department Network - Conceptual Model**



directed acyclic graph), and there are no resource constraints (only synchronization gaps). Second, one proceeds to a world with stochastic variability via Stochastic PERT networks (activity durations are random variables), showing that, in this case, the mean service time must increase.

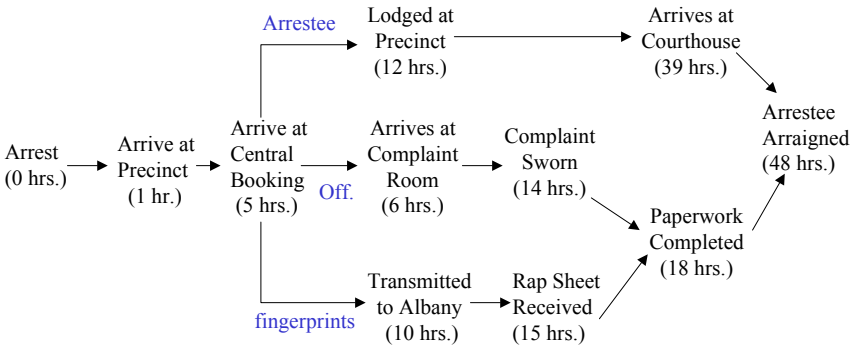
The last third step is moving into a multi-customers environment, where customers arrive to the network, each constituting a project, and they compete for scarce resources along the way. We are thus adding resource constraints, by assuming that a pool of dedicated (or flexible) resources serves/performs each activity; hence a resource queue arises if all these resources (servers) turn out busy. Given the same network structure, the magnitude of the deterioration of sojourn times, when moving from a static-stochastic system to dynamic-stochastic, is arbitrary bad (increasing with the utilization levels of the resources or, as previously described, the offered-load relative to resource capacity). This well demonstrates the need for a *Dynamic Stochastic* point of view, typical of service systems, which is captured by DS-PERTs.

In general, DS-PERTs have not been amenable to mathematical analysis. Hence, for

research, practice and teaching, one resorts to simulation. Simulation-based performance analysis appeared, for example, in Larson [49] (Figure 7), who modeled the New-York Arrest-to-Arraignment system as a (simple yet very insightful) DS-PERT.

Our involvements with DS-PERTs started is Adler et al. [1], who adopted the framework of DS-PERTs to the analysis of new product development, emphasizing the benefits of expanding the (single) project-view to the process-view. This involvement continued in [12], which is a natural application of DS-PERTs to multi-project environments, adding also some simulation-based control. Significant mathematical progress on the stochastic control of (some simple) DS-PERTs has been achieved in an ongoing Technion M.Sc. thesis. Progress has been made possible through approximations in conventional heavy-traffic, which is a natural operational regime for the hospital Emergency Department; the proposal for this thesis appears in [98], where readers can find further background and sources on Fork-Join networks.

Figure 7: **DS-PERT: New-York Arrest-to-Arraignment System [49]**



4.3.2 Applicable Conceptual Framework

As already mentioned, DS-PERT networks serve as a conceptual framework that teaches students at least the following important points: usefulness of models and the modeling process; clear *process-view* of a service system as a multi-project resource-constraint system (project = customer to be served); and the sources of operational delays, namely resource- and synchronization-queues.

In the first generations of the course, there was no homework accompanying the subject of DS-PERTs. But we added one when we rediscovered the obvious that no self-practice of a concept implies no understanding of it. Specifically, the DS-PERT view was often used in students' projects and theses, carried out after taking the ServEng course: in addition to the above-mentioned benefits, this view served as a means for understanding and charting the

service system's territory. When, during such projects, we discovered students' lack of grasp of the concepts, a homework assignment was added (see Section 5), in which students choose their favorite service system and, based on class discussions and some examples of high-quality homework from previous semesters, they depict it as a DS-PERT network model. (In their homework, students are asked to speculate on possible uses of their model, but no actual analysis is required.)

The homework requires depicting a service system through four points of view - activities, resources, information and a combined view - with a flow chart corresponding to each of these views. The four views are demonstrated in the four Figures 8-11. These were taken from a project that analyzed the flow of patients through an emergency department at a Haifa hospital:

- *Activities* Flow Chart: Figure 8 is a PERT diagram (precedence relations, allowing for loops) of the activities that an Emergency Department patient undergoes, from reception to release.

- *Resource* Flow Chart: Figure 9 is a queueing-network view, through the "eyes" of the resources, which reveals the two types of operational queues: resource queues prior to a resource, in red (rectangles); and synchronization queues after the resource, in green (incomplete triangles).

- *Information* Flow Chart: Figure 10, which was added only recently to the course, having realized in our applications that information management/engineering in an important aspect of service engineering.

- *Combined* Activities-Resources Flow Chart: Figure 11, in which each column corresponds to a resource, has been found useful for a joint single-chart summary of activities and resources.

Until recently, synchronization queues had played a minor role in our conceptualization, teaching and applications of Service Networks. This was due to two facts: first, such queues do not commonly arise in call centers which, originally, served as our main application source for Service Engineering; and second, resource queues dominate Queueing Theory and, moreover, the existing scarce research on synchronization queues is hardly transferrable to the ServEng classroom. However, synchronization queues are prevalent in healthcare, in particular in hospital EDs (e.g. a patient waiting for lab tests and for an x-ray, and then for a physician to view the results and re-examine the patient). Thus, with the increased emphasis that healthcare services have been gaining in our course, we are looking for adaptable existing theory, and we have also been attempting to develop new theory, which can then be taught in support of DS-PERT service networks.

Figure 8: **Activities Flow Chart in the Emergency Department**

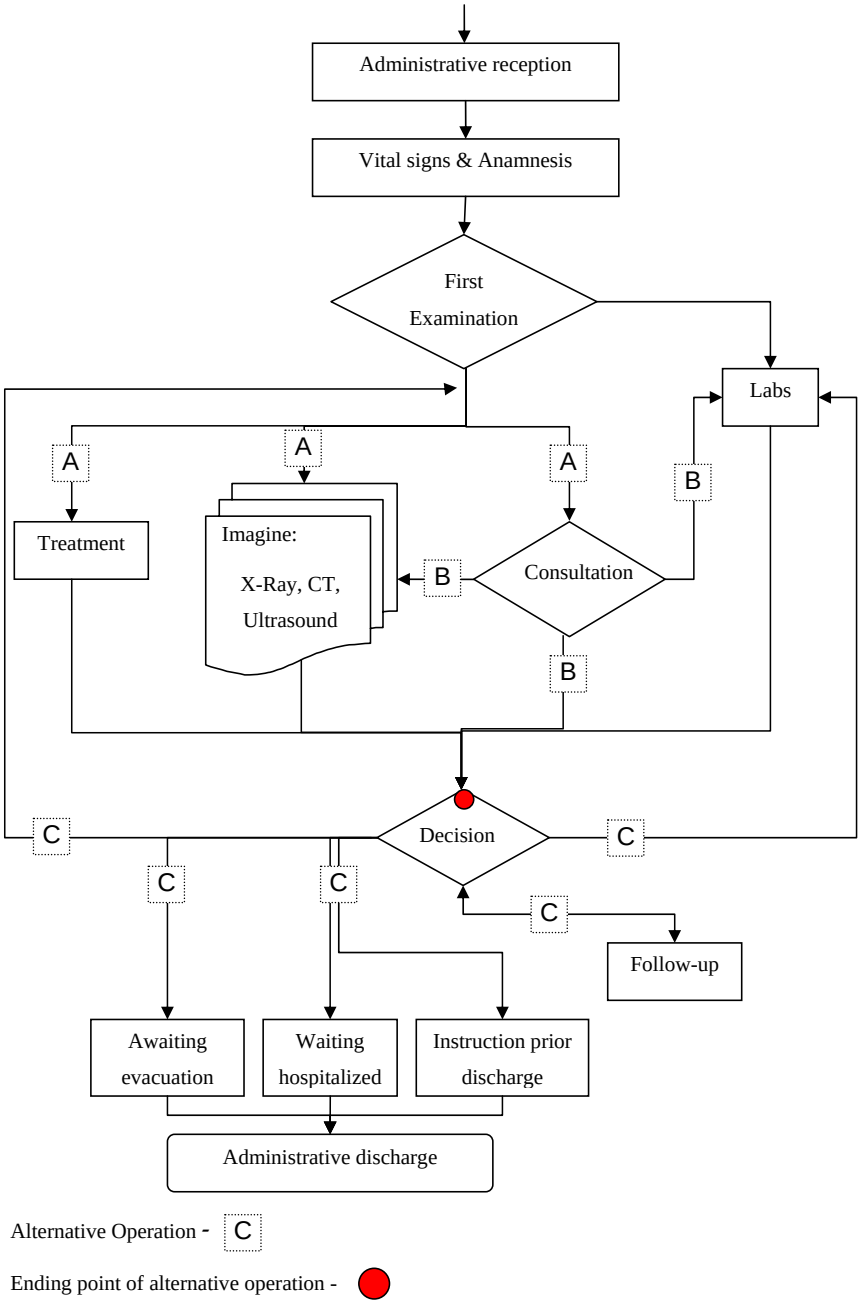


Figure 9: **Resources Flow Chart in the Emergency Department**

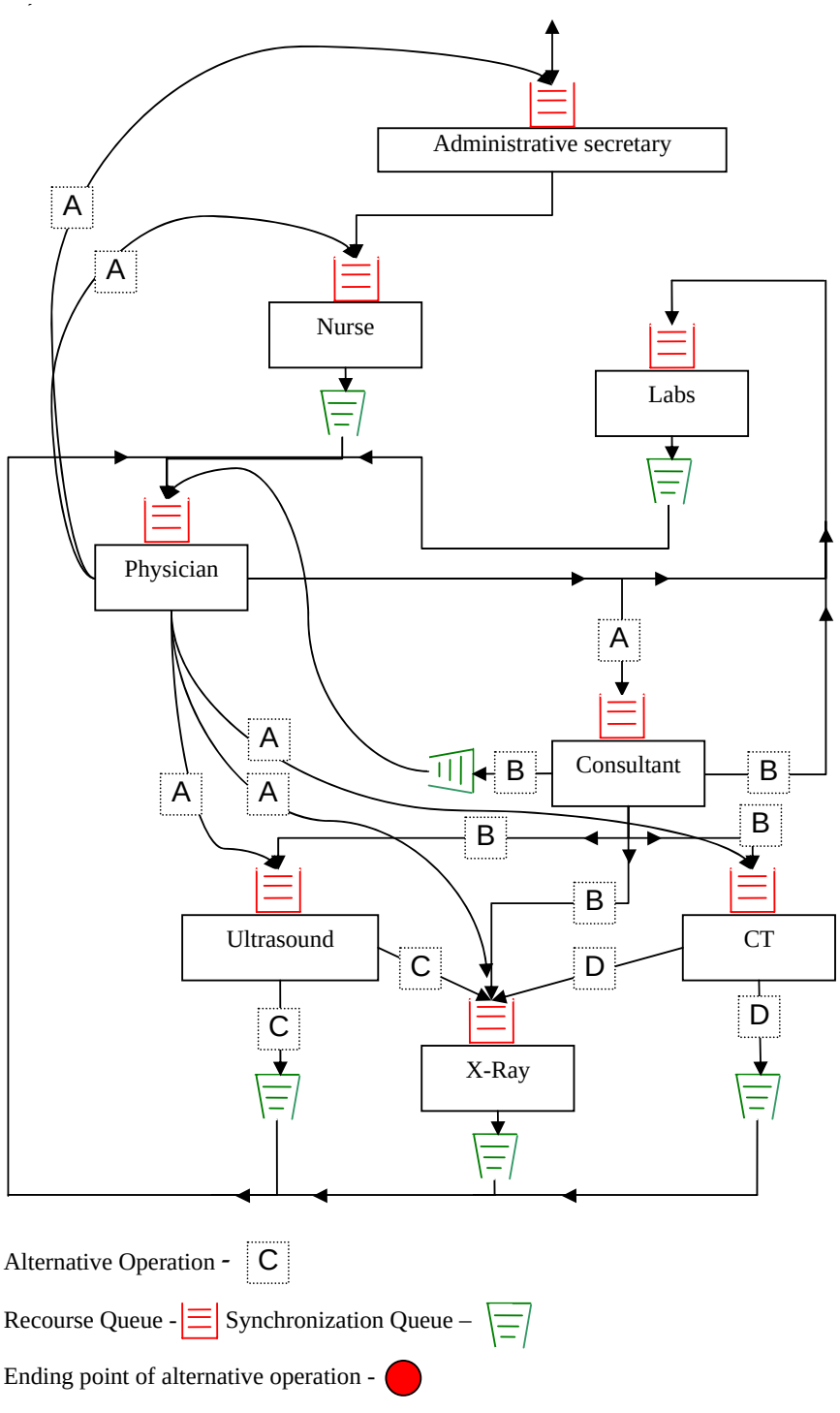
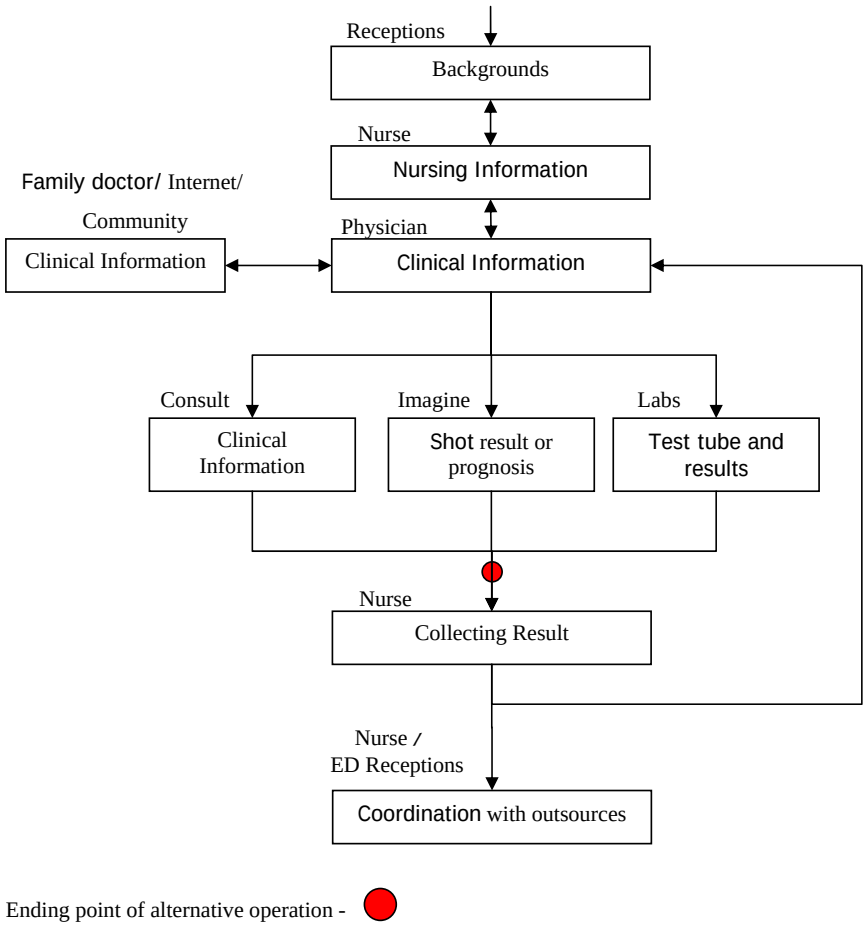


Figure 10: **Information Flow Chart in the Emergency Department**



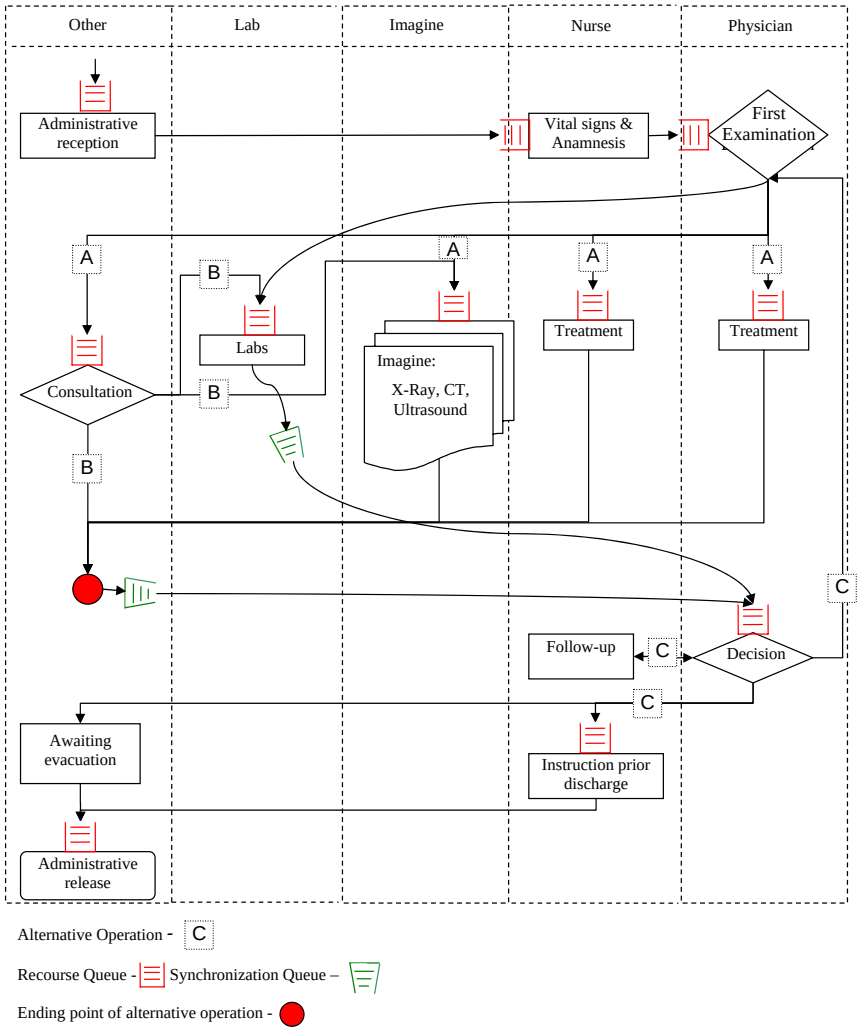
4.4 Fluid Models of Service Networks

Lecture 5 of the ServEng course introduces the fluid approach, which yields the first mathematical models of a service station and a service network. Being intimately related to Empirical models (we shall see that below), fluid models provide a bridge from the empirical to the theoretical part of our course.

As our experience suggests, the main roles that fluid models play in Service Engineering are the following:

- Fluid models are interesting and useful in their own right, being legitimate deterministic models for many real service systems.
- Fluid models provide useful first-order (deterministic) approximations of stochastic service systems, supporting both performance analysis and control of the latter.

Figure 11: **Activities-Resources Flow Chart in the Emergency Department**



- These models constitute powerful technical tools in the analysis of stochastic systems.

We now elaborate on the Fluid View, followed by several useful families of fluid models that we teach: individual-scenario analysis, queue-buildup diagrams, cumulative flows, and spreadsheet models, the latter based on finite-difference approximations of ordinary-differential-equations (ODEs).

4.4.1 The Fluid View: Motivation and Applications

The fluid view of a service system is adequate when predictable variability (e.g. manifested through time-varying demand and/or capacity) is prevalent and dominant relative to

stochastic variability. Such dominance is practically verified through relatively small coefficients of variation. For example, consider call-arrivals to a call-center, which can be usefully described by a time-inhomogeneous Poisson process, where "rush-hour" (peak-hour) arrival rate is 10,000 calls per hour, being say thrice the daily-average arrival rate. Than, during peak-hour, $CV = \sqrt{10,000}/10,000 = 0.01$, suggesting that time-variability (of the mean) dominates stochastic-variability around the mean. Hence, for some useful purposes, such a call center can be viewed as a system through which fluid (customers) flow, with the help of a given pumping capacity (agents).

Nice animations of the fluid-view are the two short videos in [25], which convincingly convey the reasons why transportation engineers view their systems through fluid-lenses. Indeed, Gordon Newell, who was a transportation engineer [72], was the earliest to champion the fluid-view of queueing systems, arguing (freely quoting from [73]) that "while stochastic-queues constitute the majority of queueing theory, predictable-queues are those most prevalent in practice". (And predictable queues are adequately analyzed by fluid models.)

The value of the fluid view increases with the complexity of the system from which it originates. We have discussed above why it is natural to view a service network as a queueing network. Prevalent models of the latter are stochastic (random), in that they acknowledge uncertainty as being a central characteristic. But it turns out that viewing a queueing network through a deterministic eye, animating it as a fluid network, is often also appropriate and useful. For example, the "fluid view" often suffices for bottleneck (capacity) analysis (recall the "Can we do it?" step from Section 1.3, which is the first step in analyzing a dynamic stochastic network); for motivating congestion laws (eg. Littles Law, or "Why peak congestion lags behind peak load"); and for devising (first-cut) staffing levels (which are sometime last-cut as well).

4.4.2 Some Useful Families of Fluid Models

Examples of systems that are prone to fluid analysis include transportation systems (as already mentioned [72, 73]), service factories and backoffice flows (e.g. mail sorting - see the classical paper of Oliver and Samuel [74]), passengers traffic in airports, and more. In fact, fluid models have been prevalent in Industrial Engineering and Operations Research, though not referred to as such. For example, the classical inventory EOQ (Economic Order Quantity) model is nothing but a fluid-model for the analysis of the tradeoff between inventor ordering-costs and holding-costs.

Sample-Paths Models: An individual sample-path of a stochastic system, in which randomness plays a minor role, can be thought of as a fluid-model. An example that we teach, and for which we have the data, is a typical government unemployment office in Israel, which

summons the unemployed for periodic visits, with invitations at either 8:00am (first shift) or 12:00pm (second), towards job assignments. Two large queues immediately build up at these times (two seesaws), which then gradually diminish till shift's end. It turns out that an individual sample-path is highly representative of system evolution, hence its individual analysis sheds light on the system as a whole - for example, analyzing both the unilateral and joint effect on queue-size, of service-capacity increase and of inviting the unemployed more than twice per day.

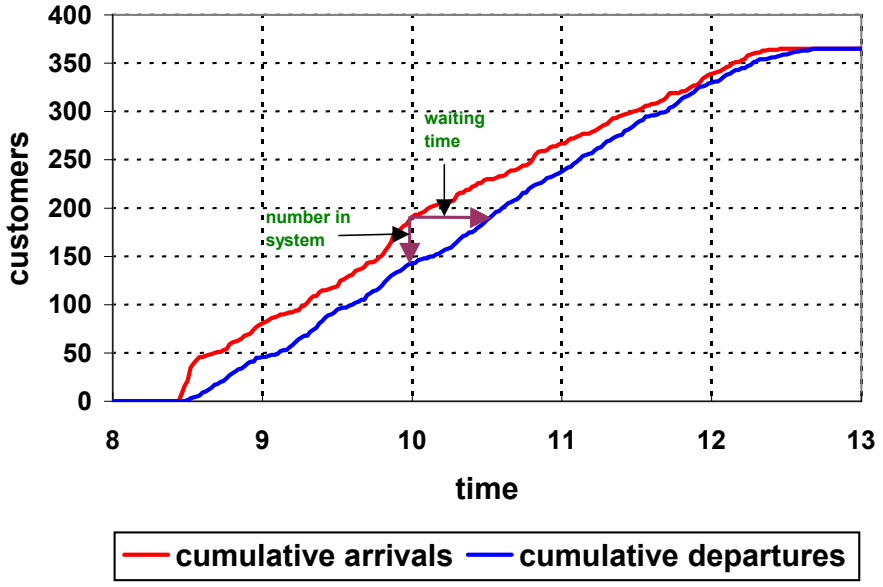
Flow-Rate Models: Our analysis of the unemployment queue captures the same tradeoffs as in the EOQ model, though now for customers that are queueing for service. Note that the famous EOQ picture of seesaw-inventory-levels (or analogously, queue-levels in a service system), should be either a highly-representative single sample-path or an average of similar (small CV) sample-paths of a stochastic (inventory or queueing) system. In either cases, this seesaw "picture" summarizes a fluid-model that is based on *flow rates* (in- and out-flows), which is commonly refereed to as an *inventory/queueing buildup-diagrams*.

Flow-rate fluid models are also useful for identifying periods of rush-hour and the magnitudes of their queues - in a single-station settings (Hall [35]), and for exposing bottlenecks and identifying dominant customer's routes - in a network setting - see [79, 80, 81]. The latter reference is devoted to a classical Harvard National Cranberry case, which we only briefly teach. We have found it instructive to leave the cranberries as the "heroes", thus representing customers that seek service in systems that are amenable to fluid analysis.

Cumulative Models: A third fluid model that we teach utilizes *cumulative* processes. It is best explained through a picture, as the one appearing in Figure 12. This figure also well demonstrates our earlier assertion that fluid models are intimately related to empirical models (created directly from measurements).

To wit, Figure 12 displays empirical *cumulative numbers* of arrivals and departures, as a function of time, in a queue of a face-to-face teller service at an Israeli bank, during some heavy-loaded day. The vertical difference between the two curves corresponds to queue-length (say, about 45 customers at 10 am). Assuming, in addition, a First Come First Served discipline (which is a reasonable assumption for this example), horizontal distances then correspond to waiting-times. For example, a customer who arrived at 10am had to wait about half-hour. Finally, and again in general (no need for FCFS), the area between the curves is the total waiting time within the system (say, in units of customers×hours) over the measured horizon: dividing this area by the total number of customer, flowing through the system during the measured horizon, yields average waiting time; and dividing it by the horizon-length gives the average queue-length; Little's Law is of course the connection between these two points of view. Again, note that Figure 12 can represent a single typical day, or the average of a number of similar days (say, Mondays).

Figure 12: **Cumulative Arrivals and Departures in an Israel Bank**



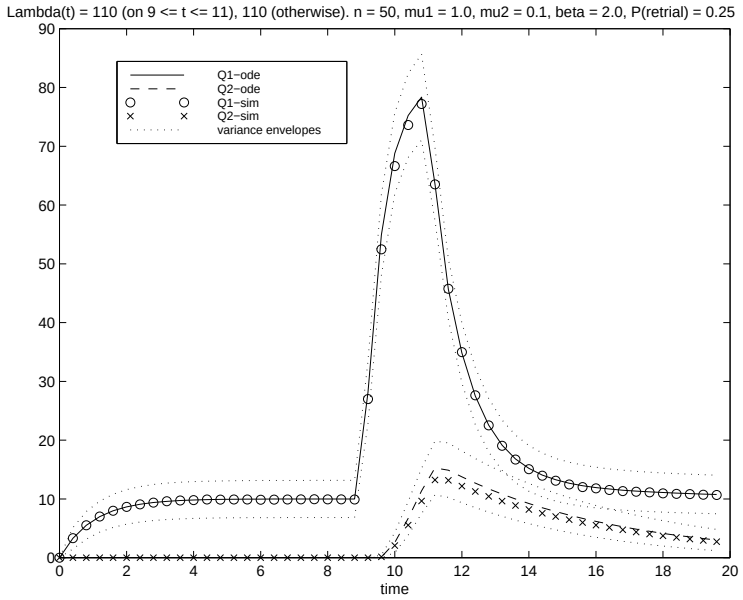
ODE/Spreadsheet Models: The last fluid model that we teach is based on ODE's. It is easily implemented by students on a spreadsheet, via ODE's finite-difference approximations. Students apply these implementations for both performance analysis and optimization (the latter with the EXCEL Solver). An example of the outcome of such an analysis is displayed in Figure 13. It arose in the time-varying model of call-centers with abandonment and redials, as in Figure 14. (The figure was taken from [58], devoted to queue-length analysis. Waiting-times, which are typically more important in the service-context, were later analyzed in [59].)

We shall not expand on this family of ODE-fluid models except for saying that it is impressively accurate, highly insightful and easily taught and implemented by engineering students. (See the description of Homework 6 in Section 5.) A convincing example is Figure 13, in which the circles correspond to sample-path averages of the number of customers in the main (upper) queue of Figure 14, at the corresponding times; the curve through these circles is in fact the Fluid-ODE path, accurate enough to be thought of as an interpolation through the circles; and the two broken-lines around the Fluid-ODE path specify confidence intervals around it (created from second-order diffusion approximations).

Additional Fluid-Views: We sometimes manage to also touch on theoretical material concerning first-order deterministic fluid approximations, via Functional (Strong) Laws of Large Numbers (FLLN). (We have even expanded to Functional Central Limits (Donsker's Theorem), as time and audience permitted). Mandelbaum and Massey [56] is our source for these functional approximations of time-varying systems in conventional heavy-traffic (single- or

few-server systems); these correspond, for example, to physicians-staffing models in hospitals. Many-server time-varying systems are treated generally in [57], and tailored to call centers in [58, 59].

Figure 13: **Analysis of a Sudden Rush-Hour, during 9:00am-11:00am.**



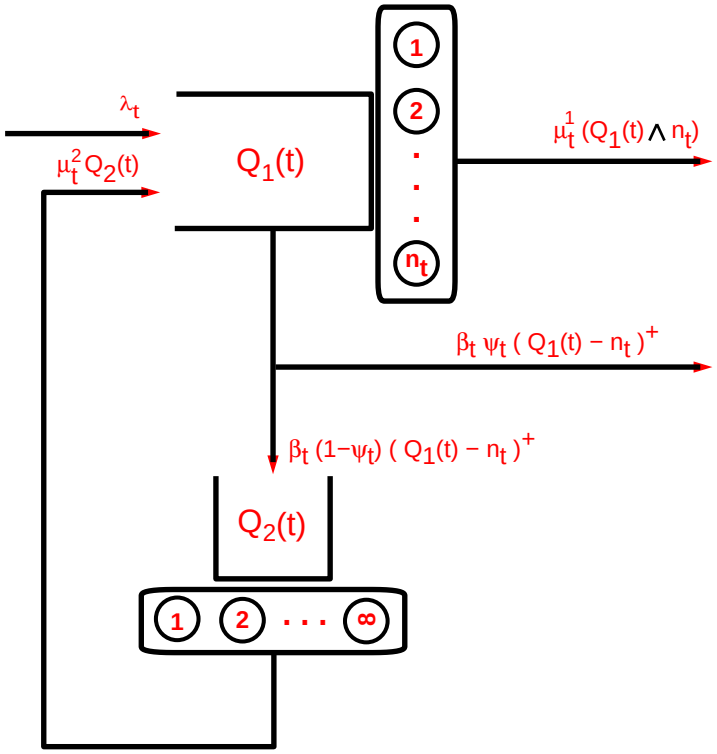
Finally, one should not fail to mention that fluid models provide powerful technical tools for characterizing stability and instability of stochastic networks. (Dai [16] started this very active research direction.) While we do not teach this topic, we do mention its existence, as a warning (and an appetizer for curious students) that sometimes naive-looking queueing networks do in fact hide unexpected subtle behavior. (Some related material, including simulations of this unexpected behavior, is provided through the reading packets at the course website).

4.5 The Building Blocks of a Basic Service Station

Starting from Lecture 6, we proceed with a series of lectures on the main building blocks of a stochastic model for a service station. These are:

- Arrival process, representing customer's demand;
- Service process, capturing work per customer that is required from servers;
- Customers (im)patience, which is what distinguishes service to a human customer from that to, say, a part in a production line.

Figure 14: **A Call Center with Abandonment and Redials: A Time-Varying Model**



Each building block is introduced, both empirically, mainly via SEEStat, and theoretically, via a corresponding mathematical model: Poisson processes and forecasting for arrivals; Exponential, LogNormal and Phase-Type distributions for services; and hazard-rates and statistical uncensoring for impatience.

After teaching Arrivals and Services, they are combined to form the workload process - the amount of work, measured in time-units of service, that is present within the service system, assuming that there are no resource constraints. The average workload, as a function of time, is then the offered load (function), introduced in Section 4.2.6, which serves as the backbone for staffing and scheduling, as discussed in Section 4.7.

Having covered the three building blocks, they are then fused into a stochastic model of a basic service station, in Section 4.6. We focus on Markovian models but touch on others as well.

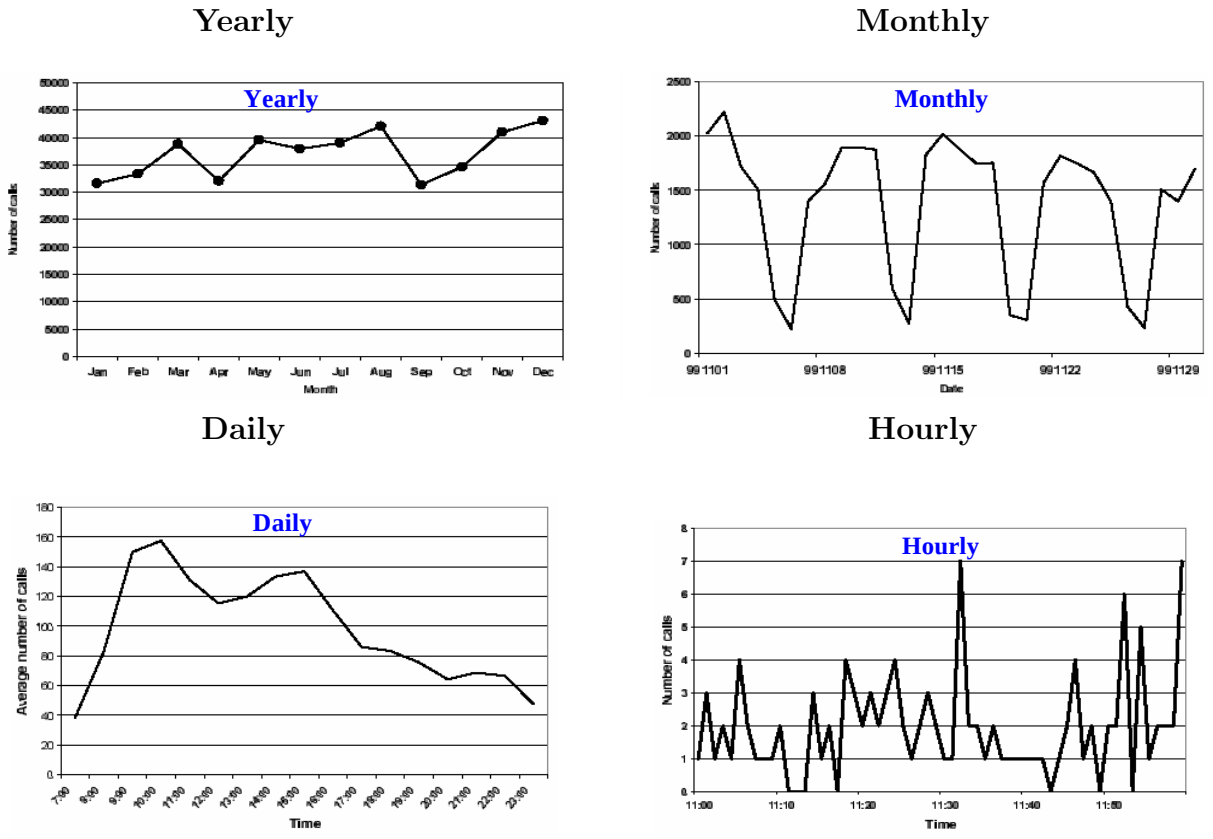
Remark: We chose to elaborate relatively more on the second building block, the service process, in order to preempt readers' wondering what there is to teach, over more than 3 hours lecture, about the operational characteristics of this very simple concept. Indeed, in Queueing theory, "service process" is synonymous for service time/duration, which looks naive

enough. (The reasons for *not* elaborating on the other two building blocks are as follows. For customers' arrivals, there is an accepted theory and modeling practice (forecasting), which is well-documented elsewhere - though we do believe to have some original insights here. And Customers' (im)patience, while hardly touched on in "operational circumstances", is amply described in the teaching note [65], of which there is yet no "service-process" analogue.)

4.5.1 Arrivals of Customers

In Lecture 6, the students learn about customers' arrivals or, in other words, customers' demand for service. Models are taught for *exogenous* arrivals, yet acknowledging the importance of arrivals from within (e.g. arrival of patients to hospital wards after being released from the Emergency Department).

Figure 15: **Arrivals to a Call Center: Time Scale**



Arrivals to a service system can be studied in several time-scales, each giving rise to a different type of models. See Figure 15 that presents arrival rates to an Israeli call center in yearly, monthly, daily and hourly scales, respectively.

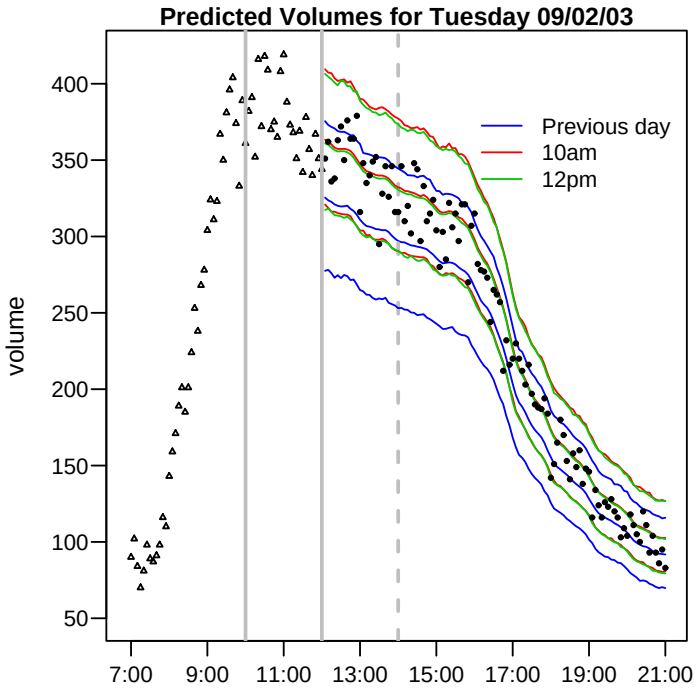
Analysis on the yearly scale could be useful for strategic decision making, the monthly pattern demonstrates strong differences in arrival volume between weekdays and weekends, daily scale illustrates the typical call center intra-day pattern (two peaks in the morning and in the afternoon) and, finally, hourly scale shows random arrival volume variations.

In our course, we focus on operational short and middle-term decision making based on arrival-rate patterns. Poisson process provides an appropriate framework for this goal. Specifically, the homogeneous Poisson model is a natural model for arrivals during short time intervals (say, half-hour or hour) and a non-homogeneous Poisson process models daily arrival rate in call centers, hospital and other service systems.

Statistical validation of time-varying Poisson model for one of call centers that we use in the ServEng course was performed in Brown et al. [10].

In the ServEng course we present several methods of Poisson process definition and provide intuition on its prevalence in real-life service systems via Bernoulli Law of rare events. We also introduce PASTA [93] (Poisson Arrivals See Time Averages) principle, which plays an important role in the analysis of stochastic models. Then a non-homogeneous Poisson process is studied.

Figure 16: Intraday Forecasting in a U.S. Bank



Although the study of up-to-date arrival rate forecasting methods is out of scope of the ServEng course, we explain to students the importance of forecasting at our lectures and recitations. For example, we demonstrate Figure 16, taken from Weinberg et al. [90] that illustrates importance of intra-day forecasting. In Figure 16 the goodness-of-fit of three intraday arrival volume forecasts is compared: the forecasts are based on the data of the previous day, of the morning data (till 10pm) and of the data till noon. It is observed that information on arrival rate in the morning significantly improves forecast quality.

Finally, we note that the Poisson model is not universal, there exist areas (e.g. some Internet applications), where other arrival models should be used, see, for example, Paxson and Floyd [78].

4.5.2 The Service Process

The importance of properly managing the customer-server interaction, or what we call the *service process*, can hardly be underestimated. Citing a manager of a banking call center that we cooperated with: "decreasing the mean call service time in our call center by one second would save us hundreds of thousands of dollars per year"; equivalently, and perhaps more tangibly, an increase of one second will cost that much.

Lectures 7 and 8 of the ServEng course explore the service process, focusing on its operational attributes: *duration* and *structure* (as opposed to contents or economic worth, for example). We now continue with a data-driven discussion of these two attributes.

Service Duration: The minimal, and most prevalent, description of service-duration is in terms of its *mean* and standard-deviation (std); better yet, the latter should be replace by the *coefficient-of-variation*, or CV for short: $CV = \sigma/E$, which is a natural measure of stochastic variability (around and relative to the mean).

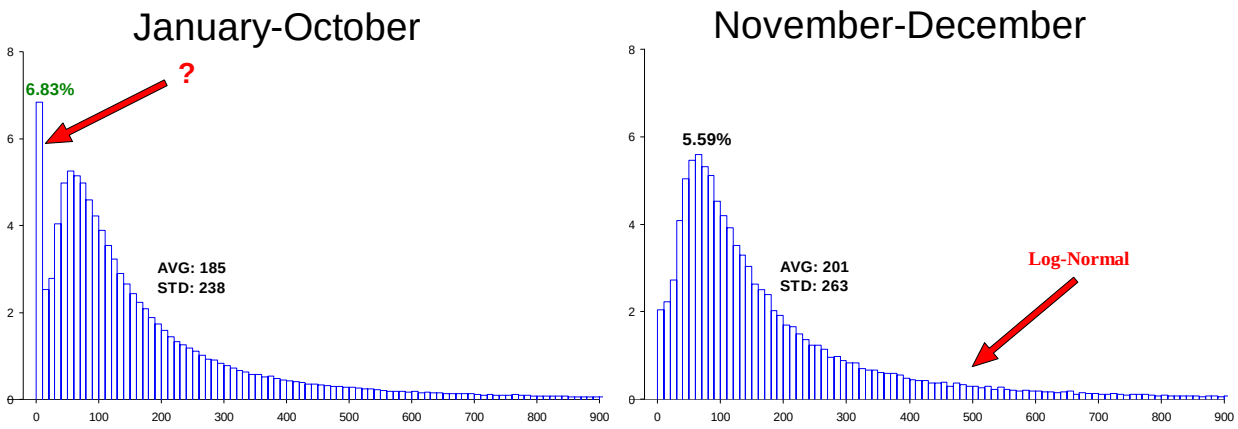
It is sometimes natural to replace the mean with the median, or another distribution quantile. Interestingly, communication engineers, at the days of Erlang and Palm, quantified stochastic variability of durations via the concept of *form factor*, FF in short, which stands for: $FF = 1 + CV^2$. (We shall return to this momentarily, in the context of LogNormal service durations.)

A more refined encapsulation of duration is its empirical distribution. It is captured by the *histogram*, which can be thought of as either an empirical aparametric-characterization or a discrete-approximation of duration. Finally. the most informative and compact, hence most useful, characterization of service duration is a parametrized distribution, for example Exponential, Gamma, LogNormal, and sometimes deterministic (e.g. the duration of a recorded announcement).,

In Figure 17, we are displaying two histograms of service durations in an Israeli call center, one for the period of January-October and the other for November-December. As

verified in [10], the November-December histogram wonderfully fits a LogNormal distribution (excluding the remote tail). This LogNormal duration is, in fact, rather typical of call centers, and other service durations beyond call centers - for example Length of Stay in hospital wards, when measured in days, as seen in Figure 18. A simple visual test of LogNormality is plotting the histogram of log(duration) which, if positive, results in the Normal bell-shape.

Figure 17: **Distribution of Service Durations in a Call Center (10 seconds resolution): LogNormality, and Agents "Abandoning" Customers**



The Human Aspect: The LogNormal distribution would also fit January-October (the left histogram in Figure 17) except for a point-mass near the origin: about 7% of the calls last less than 10 seconds. Yet, not much can be achieved in a 10-seconds call; and indeed, such calls occur when *agents hang-up on customers*, almost immediately after formally accepting the call. It turns out that this phenomenon of agents "abandoning" customers is not rare: it is prevalent in call centers that adopt incentive schemes that over-emphasize average service durations (or total number of calls performed), without sufficient attention given to the management of the call and its contents.

Figure 17 is an important teaching-device, as it effectively drives home the following important points:

- *Parametric Characterization*: It is possible for service durations to have a concise parametric characterization, LogNormal in our case. This enables one to capture the statistical aspects of duration in terms of few parameters, the mean and variance of log(duration) in our case.

  Remark: Let T be a LogNormal random variable, and denote $\log(T)$ by $N(\mu, \sigma^2)$. Then, the *median* of T is $\exp(\mu)$, and its *form-factor* is $\exp(\sigma^2)$, which makes these two parameters more natural characterizations of location and scale (when compared against the mean $= \exp(\mu + \frac{1}{2}\sigma^2)$, and $CV = \sqrt{\exp(\sigma^2) - 1}$).

- *"Natural" duration*: Service processes have their natural durations - neither too long nor too short would do, both practically and theoretically. Elaborating some, common practice focuses only on shortening durations, which is driven by flawed economics that ignores some central factors (e.g. too-short of a service could lead to high likelihood of return, resulting in both operational losses and customers' dissatisfaction; and adding a tele-marketing phase to service would increase duration but possibly also profits.). Theoretically, the scale of service duration often determines the natural time-scale, and possibly the nature (e.g. discrete queueing vs. continuous fluid) of the model to be used.

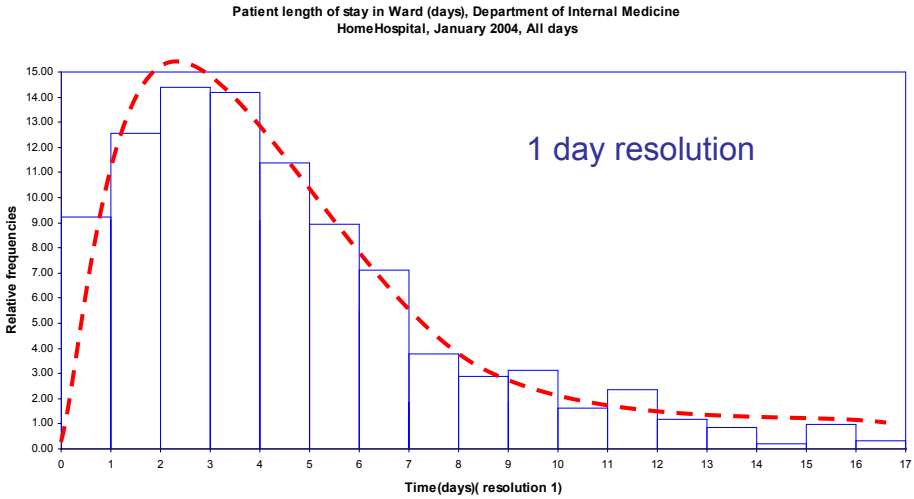
- *Resolution*: The time-scale of the distribution must fit the time-scale of the phenomena of interest - *seconds* in the case of telephone call durations. However, when measuring Length-of-Stay (LOS) in hospital wards, natural scales are hours or days, and each depicts a different reality. As an example, consider Figures 18 and 19, both depicting LOS at the internal wards of a Haifa hospital. In days, duration is close to being LogNormal; in hours, however, the policy of patients' release, which is centered around 3pm each day, clearly shows up. Another example is Figure 20, where minutes resolution gives rise to an exponential distribution (while there could possibly be a LogNormal "hiding" underneath, at the seconds-scale).

- *"Averages do not tell the whole story"*: For exposing operational phenomena of significance, service management must often rely on the full (empirical) distribution function, and at the right granularity. This is important to emphasize since, often, practitioners archive only summary averages, not even standard-deviations, let along empirical distributions. (This relates to issues addressed in our Measurement lecture; see Section 4.2.)

We conclude our discussion of service durations with some comments on exponentially distributed durations - this is a prevalent model due to the memoryless property of the Exponential distribution (the only such continuous distribution), which renders it central in Markovian models of service systems.

An immediate (only necessary) test for exponentiality is $CV = 1$ (mean = std). Having passed this test, one could visually confirm exponentially, based on the fact that the operation of rounding retains the memoryless property. It follows that any rounding of the exponential (e.g. the lower integer part) yields a Geometric distribution (the only discrete memoryless distribution); thus, an equal-bin histogram of exponential corresponds to a probability function of geometric, which is easy to recognize visually: specifically, there must be a constant height-ratio between any two adjacent bars of the histogram. See Figure 20, which

Figure 18: **Length-of-Stay (LOS) at the Internal Wards of a Hospital: LogNormality, in *Days***



shows suspects of two exponential examples. The left, with $CV = 1$, describes durations of phone-calls; the right, with $CV \approx 7.89/7.69$, arises from durations of a face-to-face service in a local municipality. Both are plotted in resolutions of *minutes* (as opposed to seconds in Figure 17).

Figure 19: **Length-of-Stay (LOS) at the Internal Wards of a Hospital: Mixture (of Skewed-Normals), in *Hours***

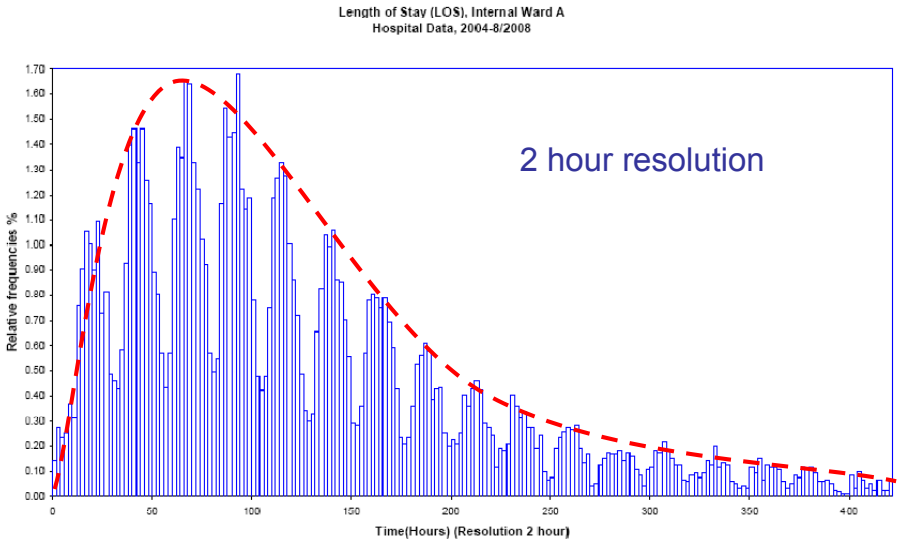
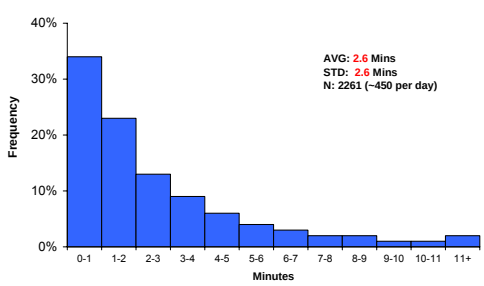
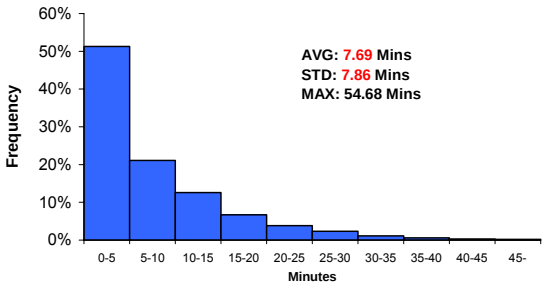


Figure 20: **Recognizing the Exponential Distribution when Seeing One: A Call Center and a Local Municipality, in minutes**



Range	Frequency
0-5	51.3
5-10	21.1
10-15	12.6
15-20	6.7
20-25	3.8
25-30	2.3
30-35	1.1
35-40	0.6
40-45	0.3
45-	0.2



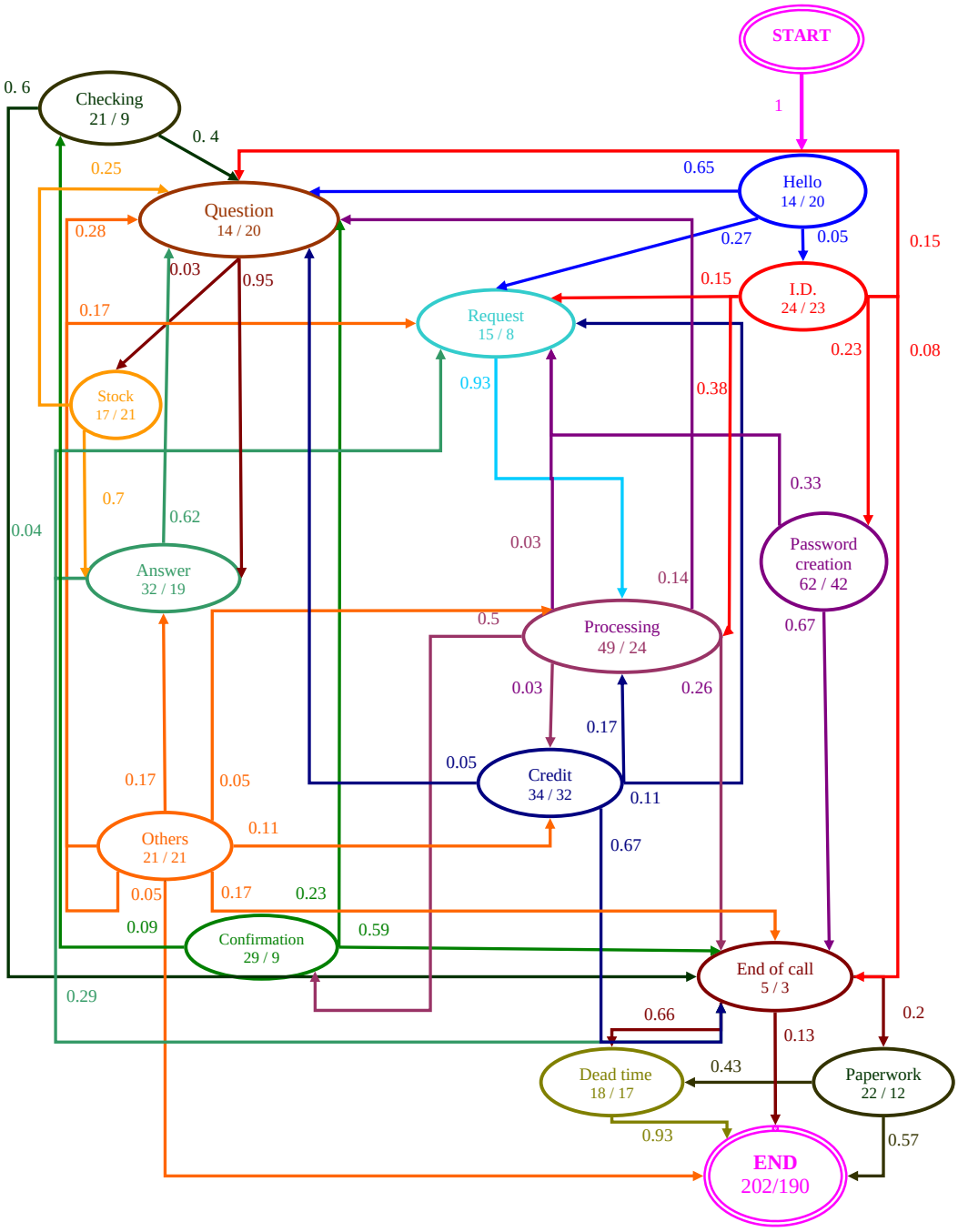
Service Structure: Duration, or its (empirical) distribution function, is a *static* attribute of the service process. (Though we shall later introduce the hazard-rate as a means for capturing dynamic attributes of a distribution). *Structure* is the attribute that captures the *operational dynamics* of the service process, which we model by a *Phase-Type (PH) distribution*: the phases correspond to evolving phases of the service process. See Figure 21, created within an undergraduate project, which presents a (data-based) model of a telephone conversation between customers and telephone agents. Note that resolution is in seconds: for example, the duration of the "I.D." phase has an average of 24 seconds, with std=23; total duration is 202 seconds on average, with std=190.

At the first stages of the ServEng course, we taught PH models only as a conceptual framework for the structure of service processes. This role comes out convincingly in the case study that motivated [60]: a service network in a local municipality transformed into a single-stop service station; to this end, the servers were trained to perform all services (flexible, or universal, tellers), yet the service process remained unaltered. Modeling the service network as a Jackson network, training all its servers as universal servers, and retaining the service process unaltered, transforms a Jackson Network into an $M/PH/N$ queueing station, where N is the total number of servers in the originating network.

There are also theoretical justifications for fitting PH models to service processes: the family of PH distributions is dense among all non-negative distributions, it can be naturally augmented into Markovian models of service systems, it is amenable to computation and last, but not least, it gives rise to elegant theory (often covered in the prerequisite course on Stochastic Processes). One can thus use PH models to demonstrate phenomena such as small vs. large stochastic variability (Erlang with n phases has $CV = 1/\sqrt{n}$, and mixtures of exponentials, or hyper-exponentials, have $CV > 1$ [60]); increasing vs. decreasing, or non-monotone hazard-rates (we have a teaching note on this [38]), and more.

But only recently have PH models of service processes matured into practice. For exam-

Figure 21: **Phase-Type Model of a Telephone Call (# within Phases: Mean/STD)**



ple, recently, in several undergraduate projects at the Technion, students analyzed phone-calls in a banking call center, fitting to it PH models. The models turned out very helpful in supporting micro-analysis of phone calls, for example the economic analysis of purchasing additional IT support (towards eliminating some phases), or performing tele-marketing

phases (which adds phases but possibly also profit). PH models are presently also developed to fit IVR interactions (at the SEELab, using data from several call centers). The latter work is statistics-focused, in the spirit of Xie et al. [94], who developed PH models for sojourn times in nursing homes.

In all our process models, a central challenge, still being worked on, is to accommodate the requirement that phase-durations are all exponentially distributed. (This is clearly not the case in Figure 21, where CVs range from about 1/3 to 1.5, yet some phases do have CVs very close to 1.) We are also unable yet to consolidate the LogNormal and Phase-Type ingredients: in its simplest form, the question is: how to "best" approximate LogNormal by a Phase-Type distribution, where "best" is constrained by the process structure that arises empirically, while also accounting for the tradeoff between model simplicity and complexity. Answers to these questions are certain to also shed light on the intriguing prevalence of LogNormally distributed service-durations.

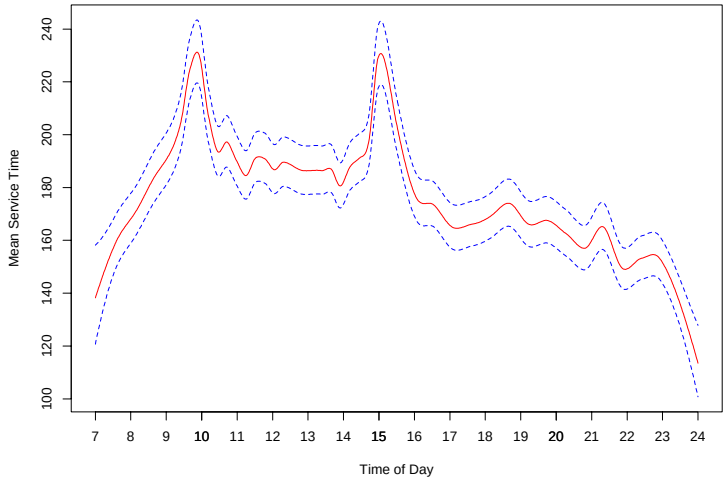
Teaching The Service Process: We interchangeably use Service-Process and Service-Time, mainly due to "old" habits from Queueing Theory, where mostly "time = duration" matters, and partly because the time-dimension is often the dominant operational dimension of the service process.

Our teaching of the service process starts with an empirical introduction, via SEEStat. We go over some of the examples above, and others where service-time distributions are exponential, lognormal or general phase-type. These examples, beyond their intrinsic interest and motivational value, also support data-based service-science and engineering; for example by comparing histograms of service durations over a 2-year period, finding out that the histograms almost overlap (for durations longer than 1 minutes) at 5 seconds resolutions! thus, not much is changing at that call center, operationally, for calls longer than 1 minute, which is clearly important to be aware and take advantage of.

We teach how to operationally calculate/estimate service durations, leaving in the relevant or sieving out the rest. We remind students of the theory of LogNormal and PH distributions, providing them (in the recitation) with statistical tools (e.g. QQ-Plots) for validation distributional hypotheses. We conclude with a discussion of the offered-load, as presented in Section 4.2.6, followed by an empirical puzzle. The offered load part uses Litvak's analysis of ambulance diversion [52] for empirical support, and it requires the fusion of the two building blocks: arrivals and service durations. The empirical puzzle is the following:

Empirical Puzzle: The *intraday* pattern of mean service duration is an important characteristic of the service process - it is often highly time-varying. See, for example, Figure 22, displaying the mean intraday service duration in a small Israeli bank [10]: red corresponds to the mean, and the blue strip around it are confidence intervals.

Figure 22: **Israeli Bank: Mean Service Time vs. Time-of-Day**



Observe that the peaks of mean service duration, in Figure 22, are twice the durations at some other times (very high variability), and the peaks occur precisely at the times of peak arrival rates (see Figure 15) and service-level deterioration, namely at about 10am and 15pm. Possible explanations for this phenomenon, which is in fact quite common, include:

- Services are longer during congested periods since customers start with complaints, which leads to apologies by agents, etc.
- Agents are over-tired hence they slow down at times of peak loads.
- Customers that call during peak hours require, for some reason, longer services.
- But the most interesting explanation, operationally, could be the following: due to high loads and low service levels, some customers opt to abandon. Those who remain are, conceivably, those who need service urgently; if, in addition, the mean service time of the latter happens to be longer than that of the general population, we deduce an explanation for the peaks in Figure 22.

The last explanation turns our attention to the phenomenon of customers abandonment - which is the third building block of a basic service station.

4.5.3 Customer (Im)Patience

The subject of customers' (im)patience, e.g. on the phone while waiting to be answered by an agent, or in the Emergency Department while waiting for a nurse or a physician, or on the Internet while waiting for a response, plays a central role in the analysis of service systems.

From a practical point of view, (im)patience could lead to customers' *abandonment* which, in our opinion, is one of the two most important operational performance measure of a call center - the other one being *redials*. Indeed, both abandonments and redials are *subjective* operational measures, through which customers inform the system, through their actions, on their perception of the service that they are being offered or received.

- Abandonment signals that the customer perceives the offered service to be unworthy of its wait.
- Redials are either "good", indicating the desire for an additional service (e.g. as a consequence of satisfaction from the previous service); or they are "bad", signaling dissatisfaction with the previous encounter (e.g. inaccurate information provided, or a flawed item sold).

From an operational/modelling point of view, redials can often (though not always) be absorbed into the original arrival process, with hardly altering, say, it Poisson nature. (See the review papers [29] and [2] for further references and discussions.) Abandonments, on the other hand, significantly affect operational performance, especially in the heavy-traffic regimes (QED and ED) that are typical of call centers. Hence the focus in the course, and its accompanying research, has been on customers' (im)patience and the resulting abandonments.

Some Personal History: On a personal note, one of the authors (AM) is greatly in dept to the abandonment phenomena - it was, in fact, the original trigger of his data-driven approach to Service Engineering, both research and teaching. To elaborate, in early studies of call centers, jointly with Professors I. Meilijson and O. Kella during the early 90s, it became clear that abandonments had to be accounted for in performance analysis of call centers. To this end, Erlang-A was used, where it was assumed that (im)patience is exponentially distributed and, hence, only average (im)patience was a required input. Average exponential (im)patience was relatively easy to estimate, based on prevalent ACD averages, which gave a satisfactory practical solution at the time.

However, it was rather clear that (im)patience is not memoryless/exponetial, hence scientific curiosity naturally triggered first the question of "what is human-(im)patience really like?" and, given an answer, "how should (im)patience be accounted for in models of call centers, and other service systems"? And this was the start of a long fascinating journey.

Part of this journey is laid out in Figure 61, which summarizes the modeling history of ServEng, from the view-point of queueing models with (im)patience customers. The first to analyze such a model was Palm [75]. Later references, history, and background on queueing

models with (im)patient customers are available in [31, 66, 67], some of which was adapted to the classroom in [65]. Having such ample sources justifies us being relatively terse from now on.

Teaching (Im)Patience: We start Lecture 9, on (im)patience, with a conceptual model of delays/waiting for service. Specifically, we distinguish among (five) waiting times that arise while seeking service, namely those that customers *(i) anticipate, (ii) are willing to exert, (iii) are being offered, (iv) experience, and (v) perceive*; a perceived waiting time affects the anticipated waiting time prior to the subsequent visit, and so on.

We reduce this complex model (worthy of psychological research, we believe) to an operational model by making the following assumptions: customers are not new to the service system, hence (roughly) "anticipated = offered"; and they are unemotional, hence "perceived = experienced". One is left with the time that a customer is *willing* to wait, denoted τ, and the time a customer is *required* to wait (offered wait), denoted V; a moment reflection now reveals that the actual (experienced) waiting time is just $\min\{\tau, V\}$: in other words, if the offered wait exceeds the customer's (im)patience then the customer abandons, otherwise the customer awaits service. In both cases, the actual waiting time equals $\min\{V, \tau\}$.

We refer to τ, the time that a customer is willing to wait, as *(im)patience*. Its distribution is the building block that we are teaching. Then, given arrivals, (im)patience and services, a queueing model outputs the distribution of V, the time that a customer is required to wait. The pair $\{\tau, V\}$ then yields all operational performance measures. For example, $\Pr\{\tau < V\}$ is an estimate for the fraction abandoning, $\mathrm{E}[\min\{\tau, V\}]$ is the expected wait, $\mathrm{E}[V|\tau > V]$ is the expected wait of the served customers, etc.

The lecture continues with some striking empirical examples of the abandonment phenomena (for example, presenting data of a call center with only 24% calls answered). Then, we discuss the practical and theoretical significance of customers' (im)patience, moving on to its operational model: the distribution function of τ.

Estimating (Im)Patience: Inference of the (im)patience distribution poses statistical challenges. Specifically, this is a classical example of *censored* data: the durations of (im)patience for abandoning customers are revealed when they abandon, while these durations for served customers are left-censored by the actual waiting time. (The actual waiting times of served customers provide lower bounds for their (im)patience.) One must thus "uncensor" the (im)patience distribution, which is performed aparametrically by the Kaplan-Meier estimator (see, for example, Cox and Oakes [14]). A much simpler (maximum likelihood) estimator is taught for the average of exponential (im)patience, which is the one students actually apply later on (and it is the one to use with the Erlang-A model, which is becoming the standard in applications.)

Hazard Rates, as Dynamic Models of (Im)Patience: It was already Palm [76], in the 40s and 50s, who proposed to use the *hazard rate* of τ as a means for *dynamically* describing (im)patience, or rather customers' irritation as a function of the time of their waiting. The hazard rate is defined via

$$h(t) = \frac{f(t)}{1 - F(t)}, \quad t \geq 0,$$

where f is the (im)patience density and F is the (im)patience cumulative distribution function. In words, given that a customer has already waited t seconds, $h(t)$ is proportional to the likelihood of abandoning within the second following t. (See [29] for detailed explanations.) As an example, an increasing/decreasing hazard rate indicates growing/diminishing impatience over time. Comparisons among hazard rates, over time-periods or across customer-types, can lead to unexpected insights.

For example, Figure 23 displays hazard rates of regular and high-priority customers in an Israeli bank (the one being used in the course.) This figure provides one with two important observations. First, priority customers turn out to be more patient than regular customers. This could be the reflection of a more urgent need on the part of priority customers; or it could be an evidence of their higher level of trust that they will be served soon after arrival. Second, both functions have peaks of abandonment around 10-15 and 60 seconds, which turns out to reflect two announcements to customers: upon joining the queue and for those who have waited one minute, respectively. The announcements inform customers on their relative position in the tele-queue. This phenomenon gives rise to important questions. Do, in fact, announcements encourage abandonment, which could be in contrast to their original goal? Do they, on the other hand, provide customers with an opportunity to take a rational decision concerning abandonment which could decrease frustration and, probably, overall abandonment? (In principle, announcements could imply larger immediate abandonment but smaller abandonment during the periods between announcements.)

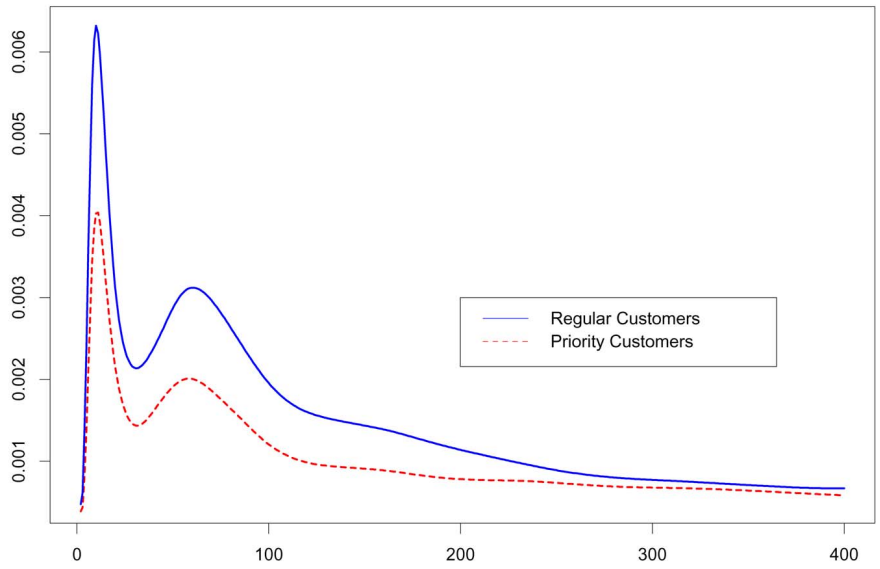
The integration of theoretical-, field- and laboratory-studies is needed in order to answer the above questions, as well as many related additional ones. See [71], arising from a PhD at the Technion (supervised by the second author of that paper), for an example of a psychological study that is based on laboratory experiments.

An Example of a Congestion-Law: The following remarkable conservation law applies to models with exponential (im)patience (in steady-state):

$$\Pr\{\text{Abandonment}\} = \theta \cdot \mathrm{E}[W_q]\,, \tag{4}$$

where θ is the parameter of the exponential (im)patience, and W_q is the waiting time. This simple relation easily follows from Little's Law: the abandonment rate is equal to the

Figure 23: **Comparing (Im)Patience of Regular (Blue) and VIP (Red) Customers (Israeli Bank)**



following two expressions:

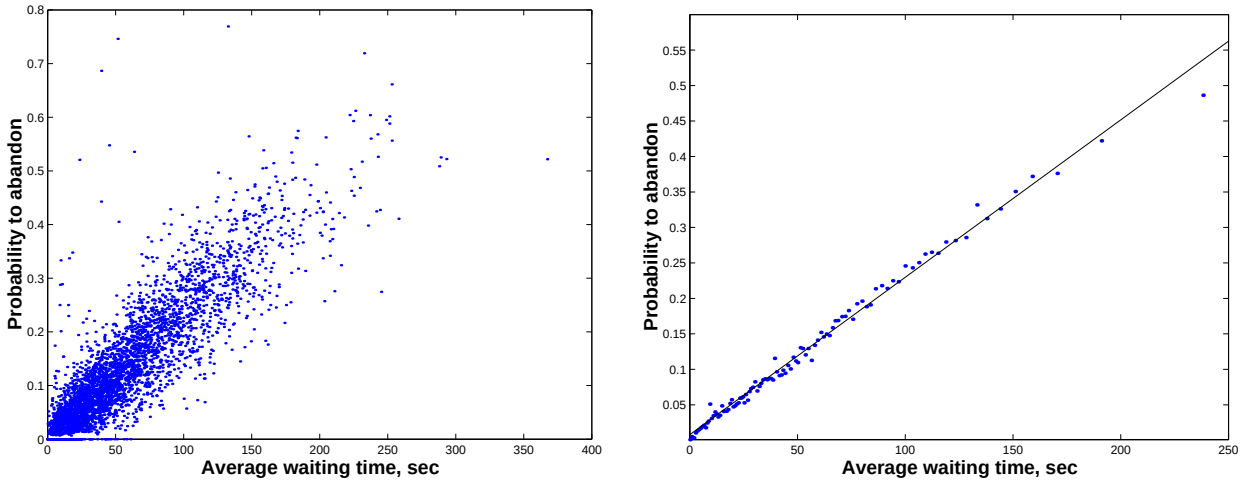
$$\lambda \cdot \Pr\{\text{Abandonment}\} = \theta \cdot \mathrm{E}[L_q]\,,$$

where $\mathrm{E}[L_q]$ is the expected queue-length, and the right-hand-side follows from the exponentiality of (im)patience. Substituting $\mathrm{E}[L_q] = \lambda \cdot \mathrm{E}[W_q]$, and canceling out the λ on both sides, yields the relation (4).

The role of θ in (4) is either that of the reciprocal of average (im)patience, or (in the QED regime) the value of the (im)patience density at the origin. The relation (4) has far-reaching consequences for modeling also call centers with non-exponential (im)patience, in particular for revealing whatever is required of the (im)patience distribution towards operational performance analysis (e.g. only its density at the origin, in the QED regime - we shall return to this at the end of Section 4.6.2).

Our (admittedly) somewhat obscure discussion is elaborated on in [66]. The conservation law (4), and its QED generalizations, are example of, what we call, *Congestion Laws*. In analogy to the natural sciences (and also manufacturing [42]), there are operational laws that apply to service systems, and which are essential to teach and acknowledge as such. An important research agenda for Service Science is the empirical validation and mining of such laws, and [66] is a step in that direction, with respect to (4).

Figure 24: **Probability to Abandon vs. Average Waiting Time**



As an illustration, consider Figure 24, which is an empirical depiction of the relation (4). It was plotted using yearly data of an Israeli bank call center [10]. First, P{Ab} and E[W_q] were computed for the 4158 hour intervals that constitute the year. The left plot of Figure 24 presents the resulting "cloud" of points, as they scatter on the plane. For the right plot, we are using an aggregation procedure that is designed to emphasize dominating patterns. Specifically, the 4158 intervals were ordered according to their average waiting times, and adjacent groups of 40 points were aggregated (further averaged): this forms the 104 points of the second plot in Figure 24. (The last point of the aggregated plot is an average of only 38 hour intervals.)

We observe a convincing linear relation between P{Ab} and E[W_q]. Based on (4) and Figure 24, the slope of this line is an estimate of average patience, which here equals 446 seconds (about 9 minutes average (im)patience, which most often seems to students surprisingly long at the first encounter). Given that patience times are indeed exponential, (4) thus provides a very convenient method for estimation of the average patience, namely the Erlang-A parameter θ. (It is, in fact, equivalent to the MLE estimator of the exponential parameter, given our censored data.) We shall explain later, in Section 4.6.2, that we achieve in fact more - we are covering the case of general (im)patience, assuming operation in the QED regime.

4.6 Stochastic Models of a Basic Service Station

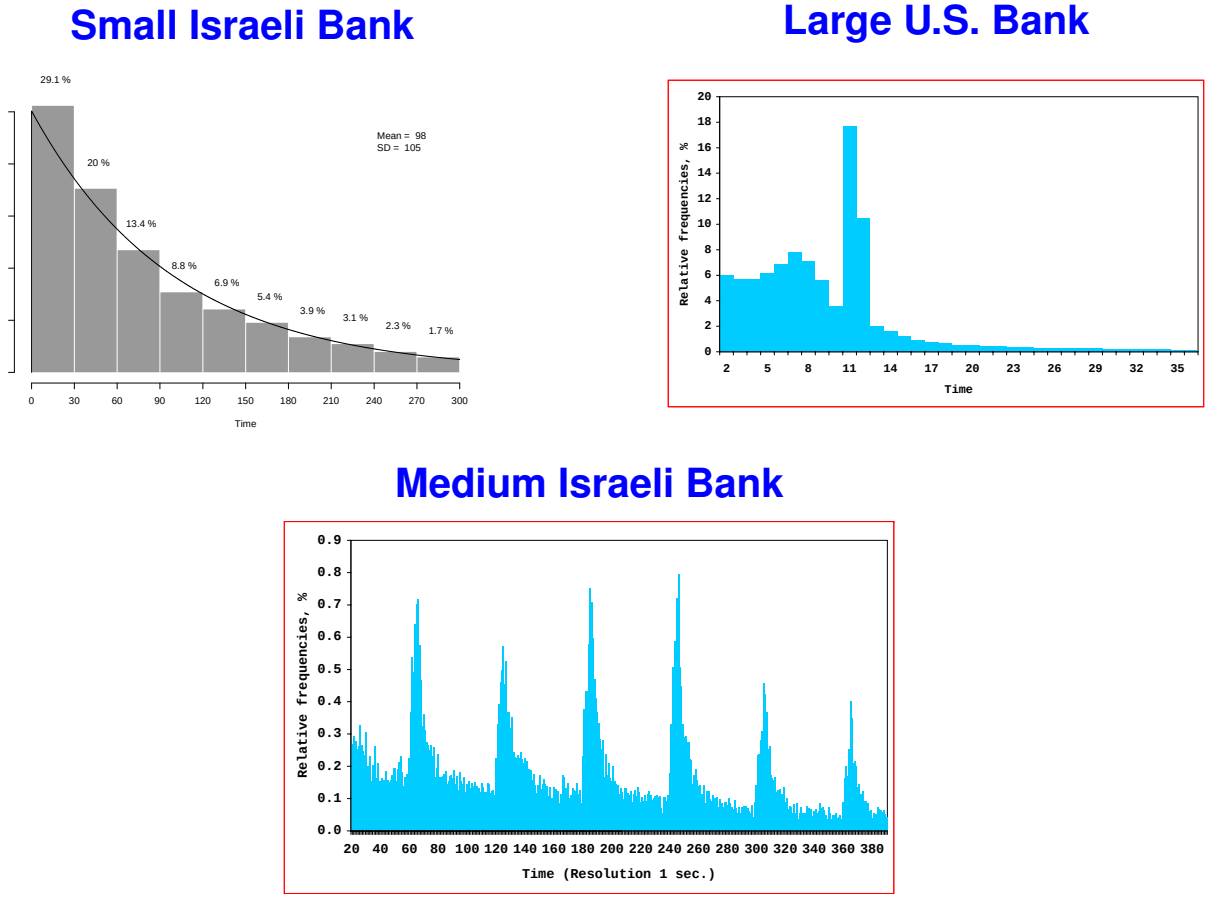
Lectures 10-12 of the course are dedicated to models of a basic service station: with a homogeneous customers' population (as opposed to multi-type) and iid servers (as opposed to varying-skills). In some sense, such models integrate the building blocks from Lectures

7-9.

Specifically, Lecture 10 provides a survey of the classical Erlang-C (M/M/n model). It also introduces the 4CallCenters software [26], or 4CC for short, which students learn to use in the course. This software is a friendly, effective practical tool for the analysis of basic Markovian queues (Erlang A/B/C and relatives). Some students continue to use it after graduation, and others are using it worldwide (especially in Germany, where it served a popular practical book on call centers [39]).

Lecture 11 reiterates the importance of the abandonment phenomena and presents the basics of Erlang-A (M/M/n + M) - the model which, in practice, is gradually replacing Erlang-C as the queueing engine of workforce-management software. Finally, Lecture 12 covers non-Markovian queues, with general service and inter-arrival distributions (G/G/n).

Figure 25: **Waiting Times in Call Centers**



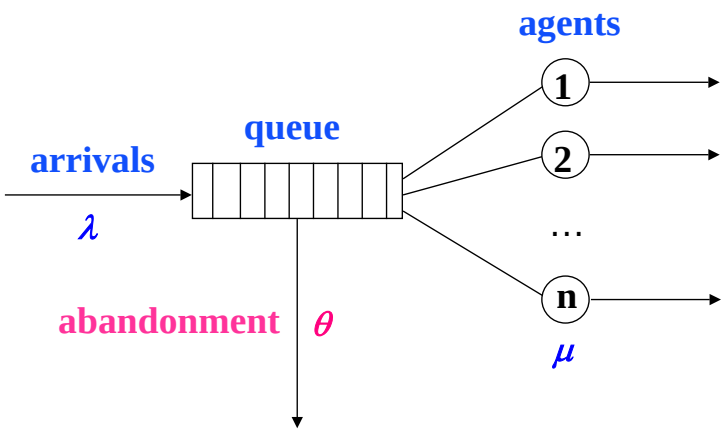
Integrating the building blocks into operational models is aimed at understanding, and possibly predicting, customers' delays. This is a challenging undertaking, as demonstrated

by Figure 25 that displays waiting-time histograms at three call centers. (Note that the resolution in Small Israeli Bank is 1/2 minute, vs. a second at the other two. The 60-second peaks in Medium Israeli Bank arose from dynamically upgrading the priority of waiting customers, every 60 seconds. The Large U.S. Bank histogram is explained later in Section 4.8.)

4.6.1 Markovian Queues: Erlang-C, Erlang-B and Erlang-A

We start with a general introduction to Markov Jump processes, which is a review of material from a pre-requisite course on Stochastic Processes. We then continue with an analysis of the classical Erlang-C and Erlang-B models (M/M/n and M/M/n/n, respectively). Although these models were introduced about 100 years ago (recall Section 1.4), they are still very useful in the modeling of various service systems. For example, both Erlang-C and Erlang-B have been successfully applied in healthcare: Erlang-C can be used to describe the queue to an X-Ray unit, or an internal ward, while Erlang-B can, under some circumstances, model (the beds in) an Emergency Department, given that ambulance diversion takes place if all beds are occupied.

Figure 26: **Schematic Representation of the Erlang-A Model**



However, Erlang-B/C do not take customers abandonment into account and, hence, are inappropriate for modeling call centers. Since the latter have been the application focus of our course, so is Erlang-A theory-wise. (Surprisingly, the necessity of taking customers abandonment into account is still not completely internalized by too many call center managers.) We thus introduce the Erlang-A model (see Figure 26), often denoted M/M/n+M, with the following assumptions:

- The arrival process is Poisson with rate λ.

- The service times are iid, exponentially distributed with rate μ.
- The number of servers is equal to n.
- The patience times are iid, exponentially distributed with rate θ.
- The building blocks of arrivals, services and (im)patience are statistically independent.

The assumptions about arrivals and services are the same for Erlang-A and Erlang-C. Recalling the concept of patience time τ, or perhaps impatience (hence (im)patience), this is the time that a customer is *willing* to wait for service - a wait that reaches τ results in an abandonment. We also introduce V, the *offered waiting time*, which is the time that the customer is required to wait - in other words, the time a customer, equipped with infinite patience, *must* wait in order to get service. The actual waiting/queueing time then equals $W_q = \min\{V, \tau\}$. In Erlang-A, the (im)patience time τ is distributed $\exp(\theta)$. Its independence of the state of the queue is reasonable for call-centers queues, under the additional assumption that waiting customers are not exposed to information about their status in the queue. As mentioned, a comprehensive treatment of Erlang-A is given by [66], and its teaching-note version is [65].

We already mentioned that, in practice, (im)patience is hardly even exponential. But we also indicated that Erlang-A and relation (4) are proved useful for performance analysis of models with *general* (im)patience, which we demonstrate further in Section 4.7.

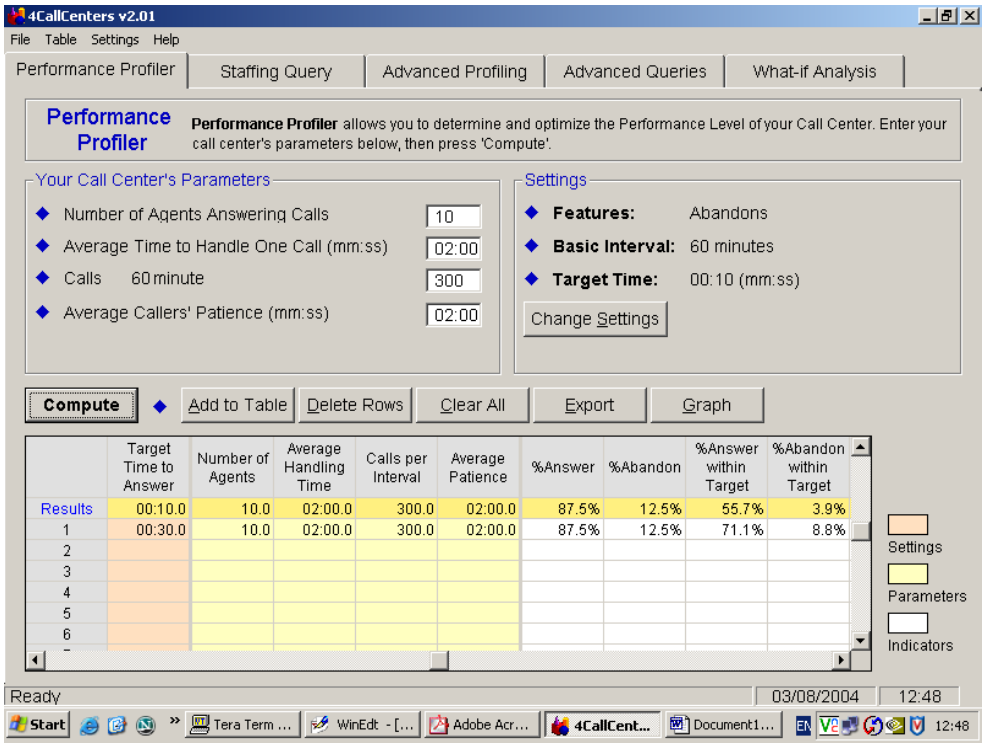
4.6.2 4CallCenters (4CC): A Personal Optimization Tool

4CC [26] grew out of Ofer Garnett's MSc thesis, and its mathematical engine is described in Appendix B of [31]. 4CC was originally developed by Garnett, jointly with a business partner who programmed the user's interface. Failing to achieve commercial success, it was then imported to the Technion, where it has gone through debugging several upgrades. These were all performed by graduate students, who have customized 4CC to its specific use by ServEng students (e.g. adding features that facilitate group work).

The 4CC software covers Markovian queues in steady-state: Erlang A/B/C and relatives (e.g. finite queueing capacity). 4CC is used in lectures, recitations and homework assignments of the course.

Figure 27 displays a 4CC output, which students generate through a friendly graphical user interface. The output screen demonstrates how to calculate the following four-dimensional operational-service measure, which we judge very useful in call centers. Specifically, given T and ϵ, which are some target time levels, customers of call centers can be divided into the following four classes:

Figure 27: **4Callcenters. Example of Output GUI.**



- $P\{W \leq T; \mathrm{Sr}\}$ - fraction of well-served customers;
- $P\{W > T; \mathrm{Sr}\}$ - fraction of served, with a potential for improvement (say, a higher priority on their next visit);
- $P\{W > \epsilon; \mathrm{Ab}\}$ - fraction of poorly-served customers;
- $P\{W \leq \epsilon; \mathrm{Ab}\}$ - fraction of those whose service-level is undetermined. (Managers often feel that customers who abandon within a short time, are not to be characterized as poorly served).

The values of the four Erlang-A parameters are displayed in the middle of the upper half of the screen: $n = 10$, $1/\mu = 2$ minutes, $\lambda = 300$ calls per hour, $1/\theta = 2$ minutes. Let $T = 30$ seconds and $\epsilon = 10$ seconds. Then one should perform two 4CC computations: with *Target Time* 30 and 10 seconds. (Both computations appear in Figure 27.) We get:

- $P\{W \leq T; \mathrm{Sr}\}$ - fraction of well-served is equal to 71.1%;

- $P\{W > T; \mathrm{Sr}\}$ - fraction of served, with a potential for improvement, is 16.4% (87.5% − 71.1%);
- $P\{W > \epsilon; \mathrm{Ab}\}$ - fraction of poorly-served is 8.6% (12.5% − 3.9%);
- $P\{W \leq \epsilon; \mathrm{Ab}\}$ - fraction of those whose service-level is undetermined is 3.9%.

4CC output actually generates many performance measures, additionally to those displayed in Figure 27: one could scroll the screen to values of agents' occupancy, average waiting time, average queue length, and more if so desired.

Robustness of Erlang-A and 4CC: Note that, in many call centers, the patience distribution is not exponential. Recall, for example, Figure 23, while the exponential hazard rate must a constant function. Yet, Erlang-A and relation (4) are proved useful for performance analysis of many-server queues with general (im)patience, specifically M/M/n+G, through the following observation [96]: in the QED regime, the relation (4) remains intact, but with θ being replaced by the value of the (im)patience density at the origin, denote it $g(0)$. Thus, Figure 24, as is, can be used to estimate $g(0)$. Moreover, all QED approximations of Erlang-A carry over to M/M/n+G, again as is, only with θ being replace by $g(0)$; it follows that 4CC can be applied as well. The very practical bottom line is that one treats the "world" as Erlang-A (when estimating θ and using 4CC), while in fact covering the more general M/M/n+G.

4.6.3 Non-Parametric Queueing Models

Markovian queueing models are approximate/compromised models of reality (e.g. service times is not exponential) but they *are* amenable to exact analysis (via the Ergodic Theorem). A complementary approach is to develop more exact models (compromise less) but, then, in view of their intractability, approximate their analysis (typically in heavy-traffic, which is indeed the practically more relevant regime to focus on). We call the latter non-parametric queueing models, in analogy to (Non-Parametric) Statistics, where no specific *distributional assumptions* are made. The GI/GI/1 queue is a prime example for a non-parametric queueing model, as nothing more than means and variances are typically assumed for its inter-arrival and service times.

A Markovian description of GI/GI/$\cdot$ models is not only beyond the scope of the present course, but it also does not yield implementable results or insights. One thus resorts to heavy-traffic approximations, which turn out tractable. In this lecture, we briefly expose students to (typically a subset of) the following topics:

- M/G/1: The Khintchine-Pollaczek formula for average waiting-time, the proof of which utilizes the "pearls" of applied probability: Little's Law, PASTA, Biased Sampling,

Wald's Theorem. We apply the formula to demonstrate, for example, the costs of stochastic variability in service durations (e.g. average waiting-time is halved when transforming and M/M/1 queue into a corresponding M/D/1, which can be animated by a robot replacing a human server.)

- Lindley's equations for the G/G/1 queue. We use it to demonstrate an easy way for recursively calculating waiting times (via a spreadsheet), and to bring out the relation of stability with the condition $\lambda < \mu$.

- The Allen-Cunneen approximation, which generalizes Khintchine-Pollaczek to GI/GI/n in heavy-traffic. It says that the *congestion index*, expressing average waiting time in units of average service duration, is given by

$$\frac{\mathrm{E}[W_q]}{\mathrm{E}[S]} \approx \frac{1}{n} \cdot \frac{E_{2,n}(\rho)}{1-\rho} \cdot \frac{C_a^2 + C_s^2}{2}\,, \tag{5}$$

  where the ratio between the left and right sides converges to 1 as the *traffic intensity* ρ approaches 1 (100% utilization) from below; here $\mathrm{E}[S]$ is the expected service duration, $\rho = \frac{\lambda}{n \cdot E[S]}$, $E_{2,n}(\rho)$ is the delay-probability in M/M/n (Erlang-C formula), and C_a^2 and C_s^2 are, respectively, the squared coefficient of variation of inter-arrival and service durations.

- Kingman's Exponential Law of Congestion for GI/GI/1 and GI/GI/n, which simply and wonderfully says that, in heavy-traffic, average waiting time is Exponentially distributed, with its mean given by (5).

- Approximations for the M/G/n+G queue (the basic call center), based on Whitt [91].

Formula (5) clearly exposes the sources of congestion, ρ and the C's, while decoupling their non-linear effect; ρ represents resources *efficiency* and C_a and C_s the *stochastic variability* in the arrivals and service, respectively. Having the n servers work parallel, helping each other, yields average delay that is $1/n$ of the delay had each of the servers catered to every n-th customer. The following issues are worth raising when teaching the above bullet points:

- These are approximations in, what has been called, *conventional* heavy-traffic - namely, up to a moderate number of servers (say 10) that are highly utilized (say 80% for a single server,and 90% for a 10-server system). This is to be contrasted against the many-server heavy-traffic regimes, specifically QED and more so the ED regime, which we turn to later.

- Conventional heavy-traffic gives rise to a remarkable *invariance principle* (Kingman's *Exponential* Law), which remarkably requires only two-moments data. (The situation is much more complicated in the QED regime, and it is simpler in the ED regime.)

- For a single server ($n = 1$, implying $E_{2,n}(\rho) = \rho$), the right-hand-side of (5) is an upper bound to its left. Furthermore, the formula is exact for M/GI/1 ($C_a = 1$ due to Poisson arrivals, or rather the exponential inter-arrival times), which is precisely the Khintchine-Pollaczek formula.

- The non-linear (convex) behavior of ρ is especially sensitive in high traffic, which is a powerful manifestation of the *tradeoff between resources efficiency and operational service quality*. Students internalize this best by plugging in (say in an M/M/1 setting) $\rho = 0.5, 0.9, 0.95, 0.99$, which results in a congestion index of $1, 9, 19, 99$, respectively; for example, in a single-server queue, server's utilization of 95% gives rise to average sojourn times that are 20 average service durations (e.g. 2 hours for 6 minutes service).

- The efficiency-quality tradeoff is ameliorated as n, the number of servers, increases. Indeed, with n large enough (very-few 10s of servers), the system reaches the many-server regimes, where the efficiency-quality tradeoff is circumvented through the economies of scale that start dominating. This is the subject of our next lecture.

- Khintchine-Pollaczek allows one to analyze the effect of stochastic variability in service durations (all summarized by its CV). Allen-Cunneen adds the effect of stochastic variability in the arrival process (all summarized by the CV of the inter-arrival times, assuming renewal arrivals). For example, transforming an M/M/1 to D/M/1 halves average waiting time; this corresponds to changing random arrivals into a reservation system, and it has the same effect as changing a human server into a robot (M/M/1 changing into M/D/1); ideally, a reservation system with a robot server has no queues (assuming $\rho < 1$).

Hall [35] is a appropriate reference for most of the results mentioned above. Going deeper requires deeper Queueing books, which do exist (e.g. Chen and Yao [11] or Asmussen [4]). There is, however, no text-book treatment yet of many-server queues, which we turn to now.

4.7 Operational Regimes and Staffing: QED Queues

A central challenge in the design and management of a service operation in general, and of a call center in particular, is to achieve a balance between *operational efficiency* and *service quality*. In Lecture 13, we consider the staffing aspects of this problem, namely having the

right number of agents when required. "The right number" means, first of all, not too many, thus avoiding overstaffing. This is a crucial consideration since personnel costs typically constitute about 70% of the costs of running a call center. "The right number", however, also means not too few, thus avoiding understaffing and consequent costs associated with poor service quality.

4.7.1 Data-Based Motivation of the main Operational Regimes

It turns out that, depending the desired balance between quality and efficiency, several appropriate operational regimes could arose through corresponding staffing rules.

Table 1: **Example of Half-Hour ACD Report**

Time	Calls	Answered	Abandoned%	ASA	AHT	Occ%	# of agents
Total	20,577	19,860	3.5%	30	307	95.1%	
8:00	332	308	7.2%	27	302	87.1%	59.3
8:30	653	615	5.8%	58	293	96.1%	104.1
9:00	866	796	8.1%	63	308	97.1%	140.4
9:30	1,152	1,138	1.2%	28	303	90.8%	211.1
10:00	1,330	1,286	3.3%	22	307	98.4%	223.1
10:30	1,364	1,338	1.9%	33	296	99.0%	222.5
11:00	1,380	1,280	7.2%	34	306	98.2%	222.0
11:30	1,272	1,247	2.0%	44	298	94.6%	218.0
12:00	1,179	1,177	0.2%	1	306	91.6%	218.3
12:30	1,174	1,160	1.2%	10	302	95.5%	203.8
13:00	1,018	999	1.9%	9	314	95.4%	182.9
13:30	**1,061**	**961**	**9.4%**	**67**	**306**	**100.0%**	**163.4**
14:00	1,173	1,082	7.8%	78	313	99.5%	188.9
14:30	**1,212**	**1,179**	**2.7%**	**23**	**304**	**96.6%**	**206.1**
15:00	1,137	1,122	1.3%	15	320	96.9%	205.8
15:30	1,169	1,137	2.7%	17	311	97.1%	202.2
16:00	1,107	1,059	4.3%	46	315	99.2%	187.1
16:30	914	892	2.4%	22	307	95.2%	160.0
17:00	**615**	**615**	**0.0%**	**2**	**328**	**83.0%**	**135.0**
17:30	420	420	0.0%	0	328	73.8%	103.5
18:00	49	49	0.0%	14	180	84.2%	5.8

Consider Table 1, which displays a typical daily ACD (Automatic Call Distributor) report of a moderate-to-large call center (in the U.S., from the Health Insurance industry). Each line summarizes the operational performance during half-hour of the day. Specifically, for

every half-hour interval, the report depicts the number of incoming calls, abandonment fraction, the Average Speed of Answer (ASA), the Average Handling Time (AHT), the agents' occupancy and the average number of agents over the interval in consideration.

We observe that performance levels, represented by ASA and Abn%, vary significantly over the day. We shall concentrate on three time intervals, highlighted in bold: 13:30, 14:30 and 17:00.

The first interval is characterized by 100% occupancy, relatively high abandonment rate (9.4%) and considerable ASA (more than 1 minute). During this half-hour, the call center is working in the *Efficiency-Driven (ED) regime*, in the sense that the emphasis is on agents' utilization, or efficiency. (Note that the number of agents is smaller than in the adjacent intervals. A probable cause could be lunch break.)

The interval that starts at 17:00 presents a contrasting service pattern. There is no abandonment and the average wait is negligible (2 sec). The agents' occupancy is far below 100% (83%). Such a service regime will be called *Quality-Driven (QD)*, in the sense that the emphasis is on customers' service quality.

Finally, the last interval (14:30) demonstrates an intermediate service regime: utilization is high (96.6%), and abandonment and waiting are neither negligible nor high. Since in this half-hour, high efficiency and service level are achieved simultaneously, this operational regime has been called *QED (Quality and Efficiency-Driven)*.

The examples above show that there exist clear differences in operational-performance which, as will now become clear, could be pre-designed (though we do not claim that was the case here). We shall now motivate the formal definitions of the three operational regimes.

First, calculate the *offered load*, $R = \dfrac{\lambda}{\mu}$, for the three intervals. We get

$$R_{ED} \;=\; 1061 : \frac{1800}{306} \;=\; 180.37$$

for the ED-interval (1800 is the number of seconds in an interval), $R_{QD} = 112.07$ for the QD-interval, and, finally, $R_{QED} = 204.69$ for the QED-interval.

In the ED regime, we observe that the offered load R_{ED} (180.37) is considerably larger than the number of agents n (163.4). This implies that agents could not have coped with the offered load unless abandonment took place. The formal characterization of the ED regime is in terms of the following relationship between n and R_{ED}:

$$n \;=\; R_{ED} \cdot (1 - \gamma)\,, \tag{6}$$

where the constant $\gamma > 0$ is interpreted as a *service grade*: larger γ will imply larger wait and abandonment. In our example, $\gamma = 1 - n/R_{ED} = 0.094$, which is equal to %Abn. The last fact is not a coincidence and can be verified theoretically [96].

For the QD-regime, we have $R_{QD} = 112.07$ and $n = 135$. The characterization of this regime is

$$n \;=\; R_{QD} \cdot (1+\gamma)\,, \qquad \gamma > 0\,. \tag{7}$$

Now we proceed to, what we believe is, the most important operational regime: QED at 14:30. In this regime, the difference between n (206.1) and R_{QED} (204.69) is relatively small and should not be quantified in units of R, as in (6) and (7). Furthermore, this difference can be either positive or negative. The appropriate characterization [34, 31] turns out to be

$$n \;=\; R_{QED} + \beta\sqrt{R_{QED}}\,, \qquad -\infty < \beta < \infty\,, \tag{8}$$

where, in our example, the service grade $\beta = (n - R_{QED})/\sqrt{R_{QED}} = 0.10$.

Note that the definitions (7)-(8) of staffing rules are appropriate for systems with abandonment: Erlang-A and the more general M/M/n+G model (general (im)patience). In the case of Erlang-C, (7) and (8) can be retained, but the service grade β in (8) should be positive. Otherwise, the system will explode. Following the same stability considerations, (6) can not be appropriate for Erlang-C. By [9], the "right" definition is then $n = R_{ED} + \gamma$, which we do not elaborate on further.

4.7.2 The QED Regime

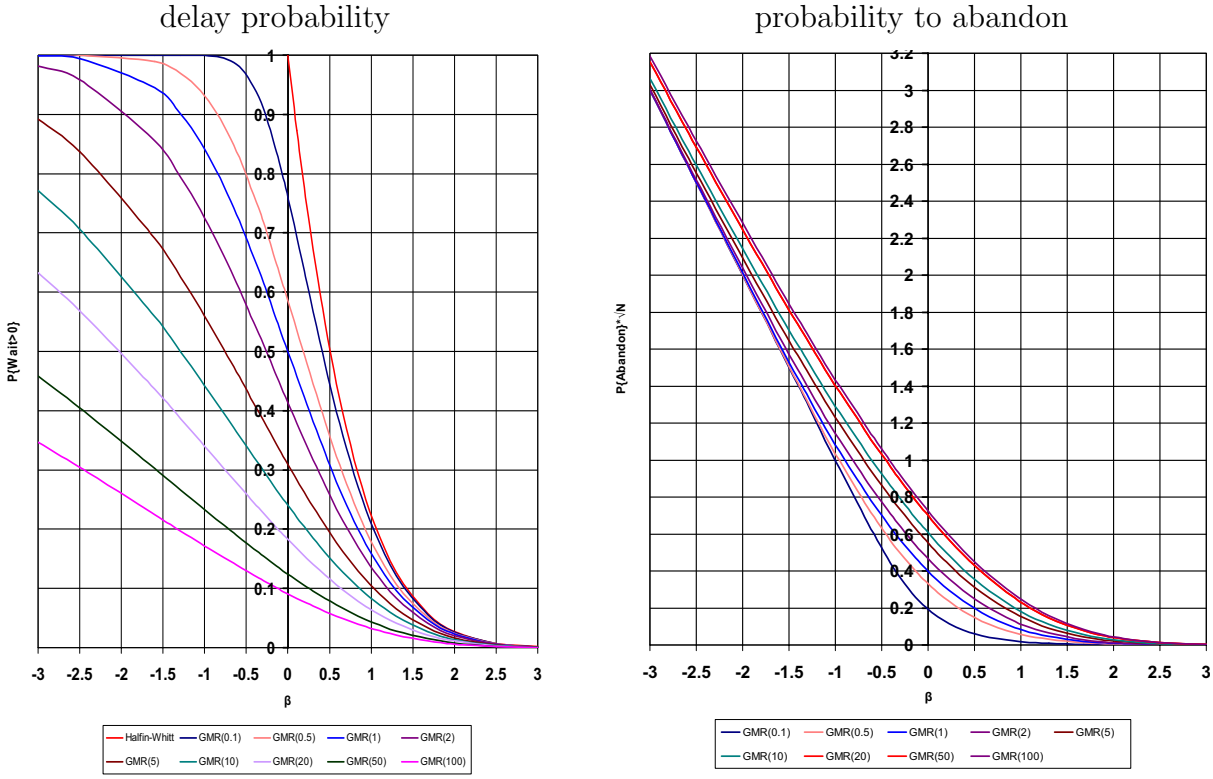
As already mentioned, the QED regime is the one to aim for, when balancing service Efficiency and Quality: hence QED = Quality- and Efficiency-Driven. The relevant history of QED Q's was described in Section 1.4.2, starting from Erlang (already 100 years ago), through the seminal paper by Halfin and Whitt [34], who characterized it in terms of delay probabilities. The reasoning is as follows, thinking Erlang-A with many servers: holding the arrival rate λ fixed, and letting the number of servers n increase indefinitely, implies convergence of $\mathrm{P}\{W > 0\}$ to 0; and letting λ grow indefinitely while maintaining n fixed implies its convergence to 1; thus, a delicate balance between the levels of arrivals and staffing is required, in order to have $\mathrm{P}\{W > 0\}$ converge to a non-degenerate limit (strictly between 0 and 1), as both λ and n increase - this is the QED, or Halfin-Whitt regime. There exist alternative characterizations of the QED regime in terms of other performance measures. The most useful one is in terms of average waiting time, being proportional to $\mathrm{E}[S]/\sqrt{n}$; in words, and assuming n not too small, this implies that average waiting time is one order of magnitude less than average service time (as, for example, in hospitals Emergency Departments where one waits hours for hospitalization that lasts days).

The QED regime goes hand in hand with square-root staffing: $n = R + \beta\sqrt{R} + o(\sqrt{R})$, where the service grade β is determined by the (limiting) delay probability, say α (and vice

versa). The function $\beta(\alpha)$, for Erlang-C, is called the Halfin-Whitt function. For Erlang-A, it is called the Garnett function, and it is computable via a spreadsheet.

Specifically, in the QED Erlang-A, the delay probability $\mathrm{P}\{W > 0\}$ converges to a constant that is a function of only the *service grade* β and the ratio μ/θ (or $\frac{1/\theta}{1/\mu}$, which measures average-patience in units of average service-time). The average wait, and hence the probability to abandon (recall formula (4)) vanish, as $\lambda, n \uparrow \infty$, both at rate $\dfrac{1}{\sqrt{n}}$. For the last two performance measure, we observe the Economies of Scale (EOS) phenomenon: service level improves with increased scale, given a fixed offered load per server. Formulae for various performance measures can be found in [31, 65]. As mentioned, QED approximations are typically extremely accurate, over a wide range of parameter values, from the very large call centers (1000s of agents) to the moderate-size (few 10s of agents) [43]; this renders QED approximations applicable to small systems, such as those found in healthcare.

Figure 28: **Asymptotic Performance in the QED Regime**



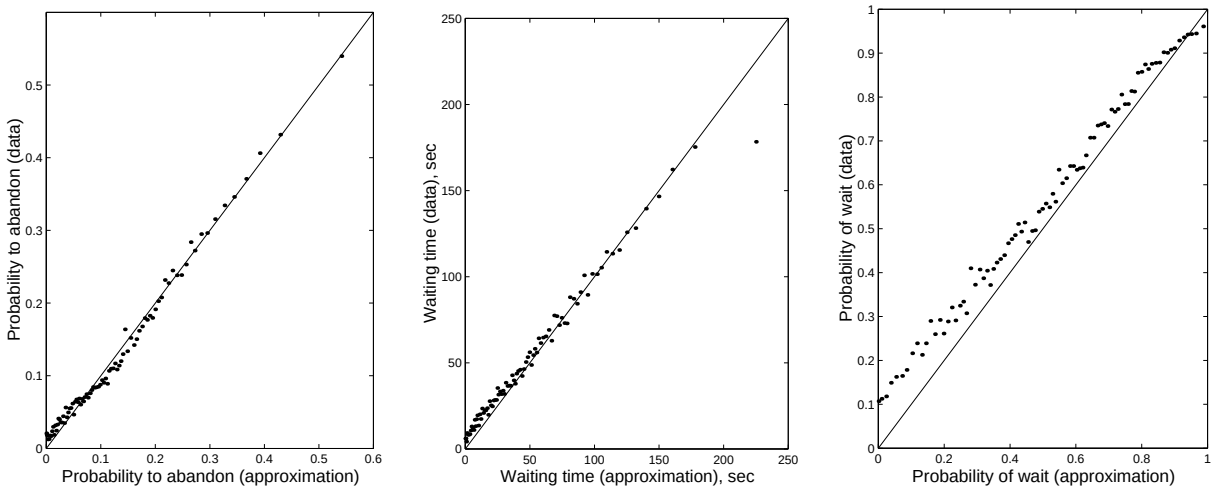
Calculating QED performance: Figure 28 illustrates the effect of β on two important performance measures (delay and abandonment probabilities), for varying values of the ratio μ/θ. In addition, we plotted the curve for the Erlang-C (Halfin-Whitt) delay probability in

the left-hand plot, which is meaningful for positive β only. Note that for large values of μ/θ (very patient customers) the Erlang-A curves approach the Erlang-C curve.

Remark on "Why Rescale - A Convincing Explanation": Figure 28 covers *all* systems (with n not too small): "all" in the sense of *all* levels of staffing and all ranges of (im)patience. This is made possible through rescaling insights, that are revealed thought QED limit theorems; specifically, the convergence of both $\mathrm{P}\{Wait > 0\}$ and $\sqrt{n}\mathrm{P}\{Abandon\}$, to non-degenerate limits that are functions of only β and μ/θ. Thus, the plot in Figure 28 displays these two limits, as functions of β, for varying values of μ/θ. For example, in order to calculate the fraction of customers who abandon, given staffing level n, one first determines the corresponding $\beta = (n-R)/R$, then reads off the function value at the right plot (on the graph with the appropriate (im)patience level), and finally one divides this value by $\sqrt{n}$, to obtain the desired fraction.

Is Erlang-A Practically Relevant?: We have already observed that the Erlang-A model is a drastic simplification of the complex reality of call centers: for example, service times and patience times are usually not exponential. Nevertheless, it turns out that if one uses historical service-time averages and estimates the (im)patience parameter via (4), QED approximations of Erlang-A provide a useful fit for actual call center data. (Some theoretical explanations of this phenomenon are discussed in [96], though a lot is left to be done in this regard. Recall also the discussion at the end of Section 4.6.2.)

Figure 29: **Erlang-A Approximations vs. Data Averages [10]**



In Figure 29, we validate the QED Erlang-A approximations against the hourly performance of the Israeli call center that is used in the course [10]. For each hour, three

performance measures are considered: probability to abandon, average waiting time and probability of wait. Their values were then also calculated, for hourly intervals, using the QED Erlang-A formulae. Then the pairs of hourly points (model, reality) were aggregated, along the same method employed in Figure 24. The resulting 86 points are compared against the line $y = x$: the better the fit the better Erlang-A describes reality. Figure 29 demonstrates a useful fit between (half-hour) data averages and their QED approximations.

4.7.3 Time-Stable Performance of Time-Varying Queues

The staffing rules (6)-(8) implicitly assume that all system parameters, in particular the offered load, are do not vary with time. This is definitely not the case in call centers and most other service systems - recall intraday arrival-rate and service-time patterns in Figures 15 and 22, respectively.

A common approach for accommodating time-inhomogeneity is to approximate a time-varying arrival-rate by a piecewise-constant function, and then apply steady-state results over the periods when the arrival rate is assumed constant. An implicit assumption here is that the arrival rate is slow-varying with respect to the durations of services. However, this approach does not always perform well in call centers (for example, during time periods when the arrival rate changes abruptly, yet predictably) and it is also inappropriate for many healthcare systems (e.g. mean sojourn time in a hospital Emergency Department is several hours, during which the arrival rate typically does change noticeably).

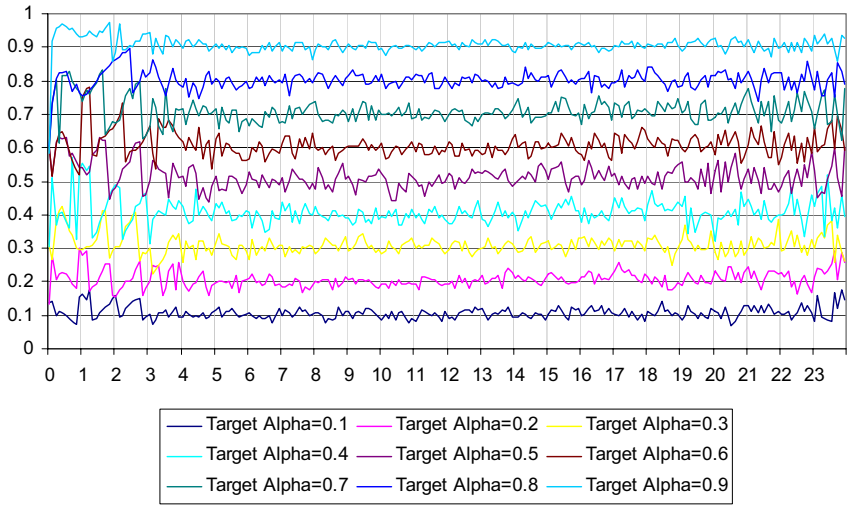
This time-inhomogeneity problem can be addressed within the framework developed in Feldman et al. [22], where the traditional staffing approach is rephrased as follows: Given a time-varying arrival process, what is the staffing level (necessarily time-varying) under which operational performance is time-stable? in other works, [22] answers the question of how to time-stabilize the performance of a time-varying system? In the answer, the time-varying Little's Law / offered-load (recall Sections 4.2.5 and 4.2.6 respectively) play a central role, as will be now explained.

Consider a time-varying queue, say $M_t/M/n_t + M$ with a time-varying arrival rate $\lambda(t), t \geq 0$, for concreteness. We are seeking a time-varying staffing levels that stabilize the delay probability at some level α. It turns out that this goal is achievable via the following time-varying analogue of the square-root staffing rule (8):

$$n(t) \;=\; R(t) + \beta(\alpha)\sqrt{R(t)}, \tag{9}$$

where $R(t)$ is the time-varying offered-load and $\beta(\alpha)$ is the Garnett function (the left-hand plot of Figure 28); the offered-load $R(t)$ is calculated either by one of its time-varying Little's Law representations (2), or by its first- or second-order approximations in Section 4.2.6. Figure 30 demonstrates the remarkable success of (9), in stabilizing the delay probability.

Figure 30: **Delay Probability under Time-Varying Square Root Staffing**



Some concluding comments are in order:

- The time-varying square-root staffing has been found extremely robust in stabilizing the probability of delay. It covers a wide range of queueing-models, including some theoretical queueing networks (single- and multiple-class), and simulated call-centers. (See Chapter 6 in the MSc thesis of Rozenshmidt [82].) Yet, there is only one case for which this startling stability can be explained theoretically - Erlang-A with $\mu = \theta$ (averages of service-duration equals average (im)patience, in which case the underlying (time-varying) Birth&Death queue becomes the easily tractable $M_t/M/\infty$; see [22]).

- Stabilizing the delay probability has the ripple effect of stabilizing other performance measures [22, 82], for example average delays and servers' utilizations, though not over all parameter ranges (e.g. abandonment probabilities over 10%, which is getting into the ED regime; see [53] for some recent developments along these lines.)

- Our stabilization of the service-level, as described above, depends on a typically unrealistic flexibility in determining staffing levels. In other words, changes in staffing can occur at any time and any level which, of course, is rarely the case. One must then apply the algorithm in [22] in order to identify staffing levels that can change, say, only once an hour or a shift. This, of course, comes at the cost of less stable of a performance.

- The present section, as well as Sections 4.2.5 and 4.2.6, underscore the fundamental importance of the notion of *workload*, which measures work within the system, mea-

sured in time units of service. (For example, saying that workload is now 5 minutes of work says that at least 5 servers must be present to accommodate this load.). The workload is a stochastic process, and its average function-of-time is the *offered-load*.

- Put simply and forcefully, and restricting to call centers and other tele-services, one should re-write all the literature that pertains to forecasting arrivals so that its focus turns to forecasting *workload* (the average of which is the offered load). Estimating arrival rates, though in addition to the offered load and not instead, remains important in systems where limited physical space is available to host delayed customers (e.g. Hospital Emergency Departments or call centers with limited trunk-capacity). In tele-services, however, queues are just ordered lists within some computer program or data-base, hence only the workload matters operationally.

4.7.4 Workforce Management: Hierarchical Operational View

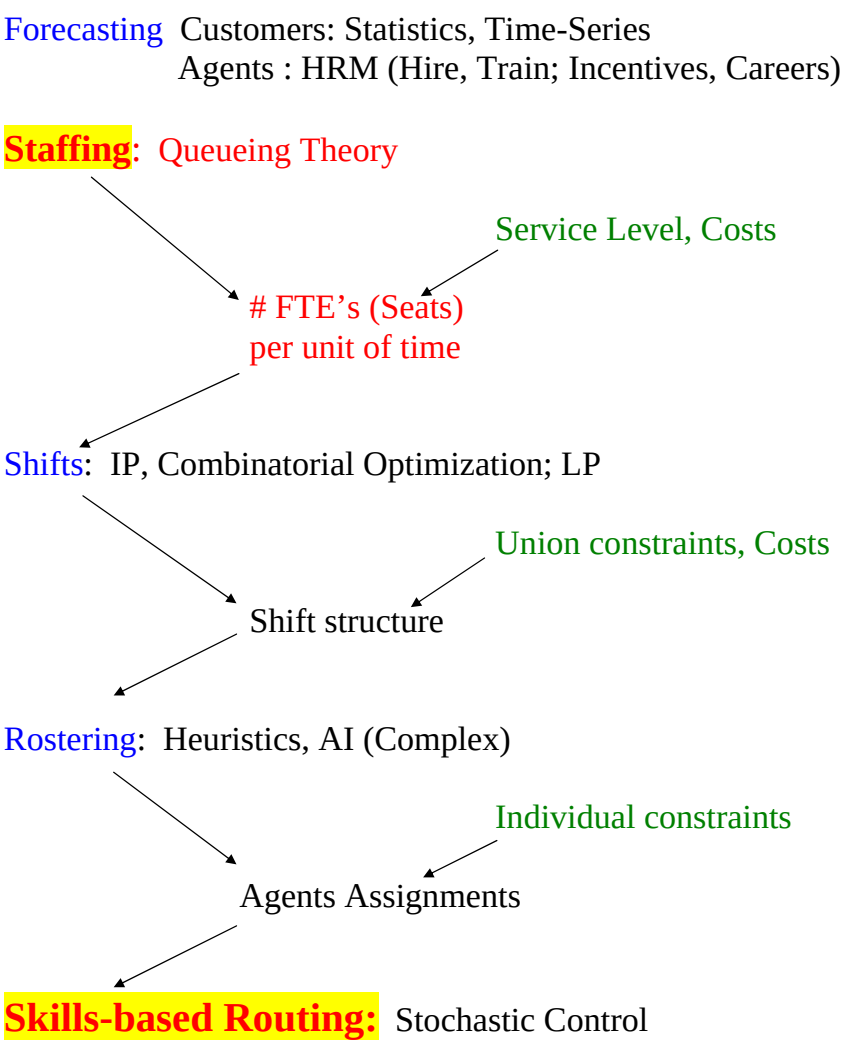
While we teach mainly staffing methods, within operational WorkForce Management (WFM), we do emphasize that the spectrum of WFM is far broader, and we do teach some elements beyond staffing the basic service station. For example, the QED regime above specifies the number of agents that should serve customers during each basic interval (say, half-hour). However, unless working from home, agents are not assigned to work at a specific half-hour. Instead, they follow some exercised shift-policy, for example eight-hour shifts with one half-hour lunch break, and two additional 15-minutes breaks, one before lunch and one after.

Figure 31 summarizes the hierarchy of decisions along the WFM chain, starting from forecasting at the top, and ending with real-time control at the shop-floor level (Skills-Based Routing, in the case of call centers; but also Judges, as we saw in 4.2.3). Each level in the hierarchy is subject to constraints, which are either exogenous or passed on from the level above it; its output then provides constraints to the level below it. Being more specific, the most basic form of the hierarchy is as follows, from top to bottom:

- *Forecasting*: One starts with forecasting of the three building blocks, mentioned in Section 4.5: arrival rates, service time and customers' (im)patience, combining the first two to create forecast of the offered-load (or workload). Significantly, one must also forecast agents' availability, based on hiring and training figures, which is made complex due to high levels of absenteeism and turnovers.

- *Staffing*: The next level accepts service-level and costs constraints. It then applies queueing models, subject to the given constraints, in order to design hourly (or half-hourly, or less) staffing levels. The output of this stage are piecewise-constant daily

Figure 31: **Call Centers: Hierarchical Operational View**



staffing functions. In is important to emphasize that the required staffing function is in fact requirements on the number of agents' positions (FTEs, where FTE = Full Time Equivalent). For example, as a consequence of managerial practice at a specific call center, one FTE in position requires the presence of 1.2 agents on premise which, in turn, requires summoning 1.5 agents to work. This is because some summoned agents do not show up to work (the second gap); and those that show up, sometimes go on breaks, attend meetings, etc. (the first gap).

- *Shifts*: Integer programs, or combinatorial optimization, is then used to aggregate say hourly demand into shifts. (The corresponding IP, in its simplest form, is of the set-covering problem type, which is taught at the course). Staffing is subject to union

constraints on working patterns, and possibly also staffing constraints. The output here is a shift-schedule, specifying how many agents should available to occupy each shift (with some of the shifts perhaps overlapping.

- *Rostering*: Out of the body of workers, who should come to work and when? this is a challenging problem in large call centers, as it could involve the individual constraints of many enough agents so that the problem becomes computationally intractable. Still, some WFM suits create reasonable to good rosters, through combining heuristics, AI and Optimization (specifically Integer Programming).

- *Skills-Based Routing, or SBR*: Given the *skills* of agents at work, and the *types* of calling customers, each call must be matched with an agent in real time (based on off-line design of the feasible (skill, type) pairs). Protocols for the rules of such matching are called SBR protocols, which we briefly describe in the next section.

The above WFM hierarchy is not unique to call centers. Indeed, nurse staffing in hospitals goes, or should go, through exactly the same steps [77, 40], as do other labor intensive services. Except for rostering, all steps of the hierarchy are covered in either lectures or recitations,with students learning and practicing forecasting, queueing models, shift scheduling (solving an IP), and some SBR principles.

4.8 Heterogeneous Customers and Servers: Skills-Based Routing (SBR)

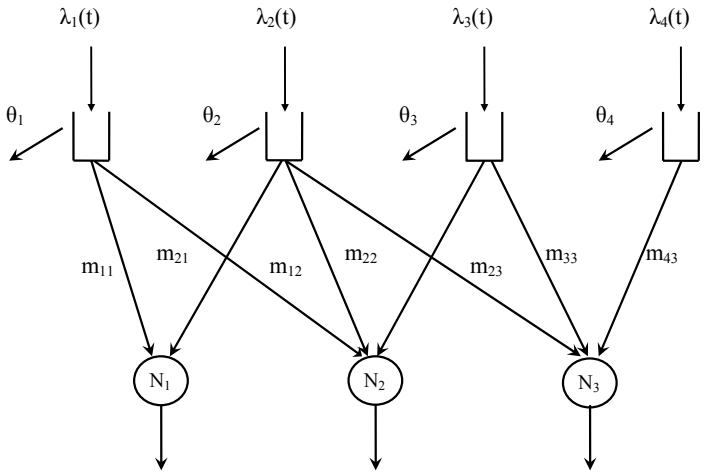
Our models have assumed so far iid customers and iid servers. A next step towards the complex reality of service systems is to allow heterogeneity of either customers or servers, or both - which amounts to studying *queueing networks* (recall Section 1.3.) To maintain some level of analytical tractability, one can proceed in one of two ways. Either render complex the topology of the network while maintaining its protocols simple, for example as in Jackson networks. Or, conversely, assume a simple network topology while allowing complex protocols, as is the case in SBR or Fork-Join networks. In view of time and scope constraints, only one of these models can be covered in ServEng - and we chose SBR: it seemed to be the more natural path when teaching services, and it was consistent with the course's emphasis on call centers. Our recent expansion into healthcare, however, suggests a corresponding expansion of modeling scope: first the addition of fork-join queues, then Jackson networks, and finally either networks that allow Markovian routing jointly with fork-join constructs or Kelly/BCMP networks. But this is yet to be pursued.

Thus, the last lecture covers SBR and related topics - which is the lowest level of the hierarchical operational view presented in Figure 31. As indicated above, SBR models are

examples of queueing networks, in the sense of having several interacting queues, and in which the network topology is simple but the service protocols could be, and are often made, complex. SBR captures the capabilities (flexibility) of agents in call centers, as well as in other service systems, to cater to more than one type of customers.

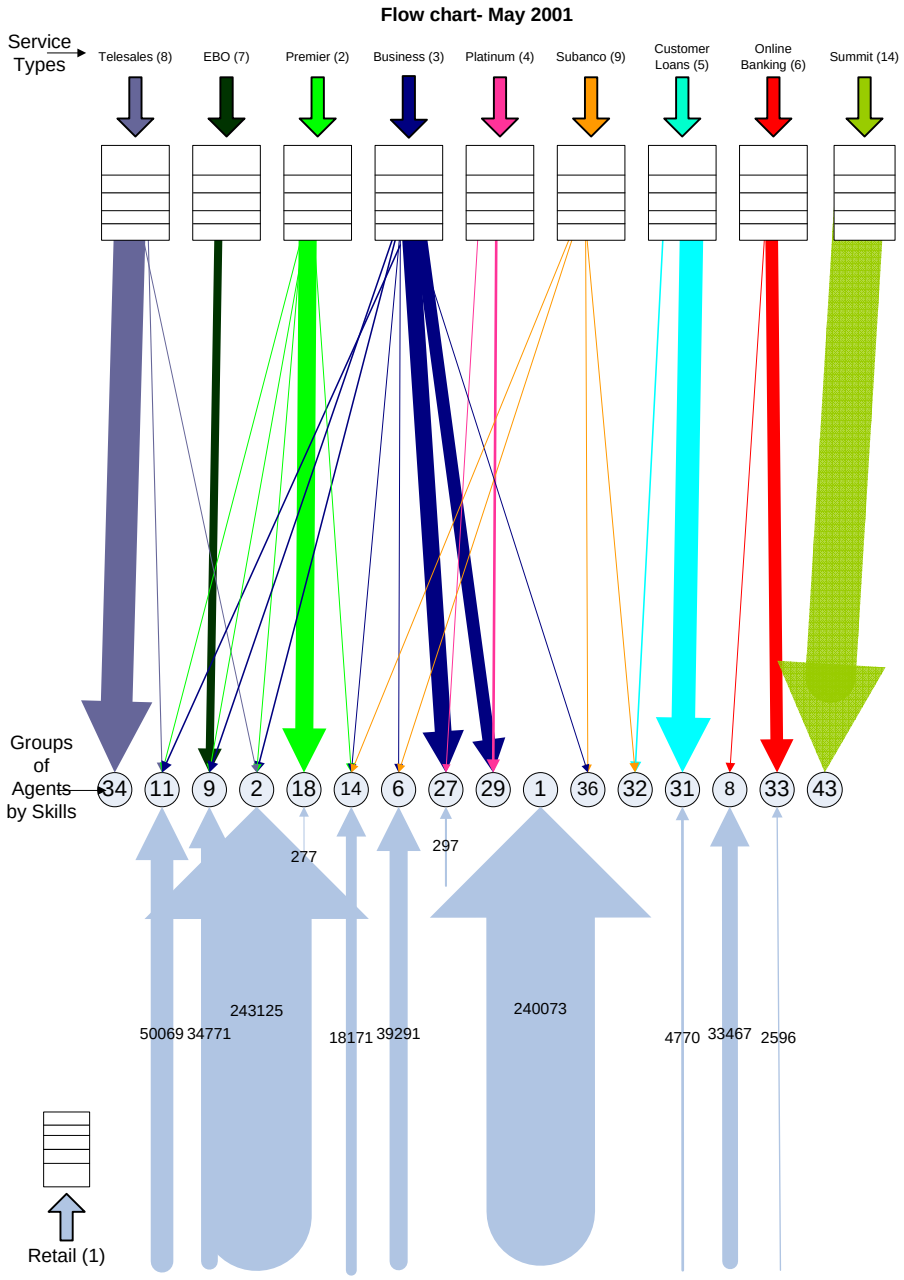
A schematic SBR diagram of a call center is presented in Figure 32. Here λ_i are arrival rates of the various customer types, θ_i are their (im)patience rates, N_j are the number of servers in the various pools (skills), and m_{ij} are average service times. (For example, if customers of type 2 are served by agents from pool 1, the average service time is m_{21}. (Each server pool consists of a homogeneous group of servers, namely with an identical skill-set.) In practice, SBR topologies vary from the very simple, e.g. having just one pool of fully-flexible (universal) agents, to the highly complex, as depicted in Figure 33, which is taken from [50] (a technical report of the SEELab).

Figure 32: **SBR - A Schematic Diagram**



Applying an SBR architecture must be an orchestrated multi-disciplinary effort, covering the design and management of Operations (and Operations Research), Marketing, Human Resources (HRM) and Information Systems (MIS). (This is clearly indicated in Figure 5). And while SBR was conceived as a technology for improving customers' flow, its benefits have turned out to go beyond operational efficiency and service quality. For example, SBR technology enables job enrichment, incentive design and the creation of career paths in call centers. These are extremely valuable options in an environment that is plugged with tele-stress, hence job dissatisfaction that culminates in high turnovers.

Figure 33: **SBR in a Call Center of a U.S. Bank**

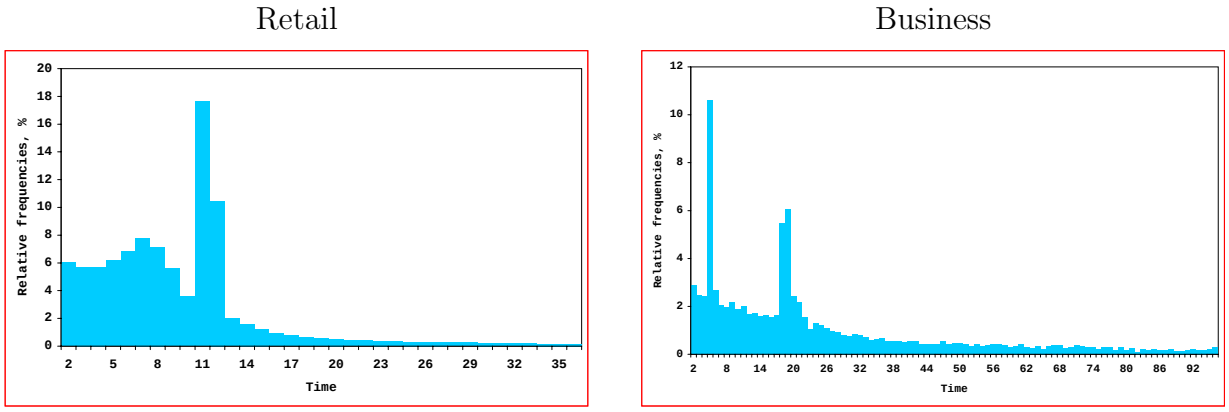


To elaborate on the multi-disciplinary pre-requisites for SBR: segmenting customers is a marketing task, agent segmentation (and hence training) is HRM, the enabling information infra-structure is MIS, and the type-skill matching is an operational challenge, both in real-time and off-line. For example, at the real-time level, one should apply algorithms for *agent selection* and *call selection*: choosing which free agent should an arriving call be routed

to, if any; and which waiting call should be attended to by an agent who becomes idle, if any. And off-line, multi type/skill environments significantly complicate the staffing and shift-scheduling problems discussed previously.

In conventional heavy-traffic, and in the absence of abandonment (e.g. email-based call centers, or the "production" of justice from Section 4.2.3), SBR protocols are well understood and easily explained: they are mere generalizations of the classical $c\mu$ rule, as discovered in [63]. These (asymptotically) optimal algorithms can be thus clearly taught to ServEng students. But adding abandonments or turning to the QED regime present novel challenges, which has triggered extensive research in recent years. This territory is far from being fully charted, yet it has already generated insights (conceptual, structural, algorithmic) that are simple and useful enough to be exportable to the ServEng classroom. Until this is done, we have been taking a practically-oriented approach, covering parts of the teaching note of Garnett and Mandelbaum [30], and expanding with some industry papers and recent research results when feasible, and as time permits.

Figure 34: **Histograms of Waiting Times in a U.S. Call Center - Influence of Routing Protocols**



We also demonstrate several data-based scenarios that bring out the importance of SBR and corresponding routing (load-balancing) protocols. For example, Figure 34 shows the histogram of waiting times for two customer types in a large U.S. call center. There are several noticeable peaks, for the following reasons. The call center in consideration consists of four smaller call centers in as many cities, which are interrelated through the following dynamics: an inbound call first seeks service in its local call center, which follows some local SBR protocol; if delayed in its local queue for over a specified time (e.g. around 10 seconds for Retail calls), the call is placed in an inter-queue (multi-location queue) which is served by all call centers, following another global protocol. The success of these protocols

is manifested by the Retail (left) histogram in Figure 34, which clearly demonstrates a peak above 10 seconds or so, and a sharp drop to relatively few delays beyond 11 seconds. (A similar story can be told about Business customers, with a threshold of 5 minutes, though we have not founded explanation for the peak above 20 seconds.)

5 ServEng Homework and Exams: A Data-Based Approach

In concert with the way the course is taught, both in lectures or recitations, the course assignments and exams are also data-based, when feasible. ServEng students are thus exposed to various real-world data sets, with which they apply theoretical and practical tools that they were taught. The data-based exercises still go hand in hand with ample interesting theoretical homework, so that understanding the material goes beyond the "recipe level". Homework weighs about 50% towards the final grade, while the rest is determined by a final exam (3-3.5 hours). In this section we describe both homework and exams.

Data-based homeworks and exams are difficult to create, hence the support of the SEELab is essential here; and, even so, the homeworks must serve the course over a relatively long time, and writing an exam takes many hours and iterations (a joint Instructor+TA effort).

Data analysis tends to require relatively many hours of work from students, some of it not directly rewarding, yet it is very important for the learning experience of ServEng. Hence homeworks are carried out in groups (up to 4 students in earlier generations of the course, and student pairs in recent generations), and their weight in the final grade is substantial - around 50%, as already mentioned. Group assignments also save on feedback and grading resources, which are not in abundance. (High quality feedback is a must for the hard-working students, in order to help them both learn from their mistakes as well as sustain their motivation.) To achieve all the above, the support staff of the ServEng course (TAs, homework graders) must be dedicated and of the highest quality, which has indeed been the case - we return to describing their essential contributions in Section 7.

5.1 Homework Assignments

We first present a complete list of the homework assignments, with accompanying brief descriptions. Then we elaborate on a couple of the assignments that nicely represent the "data-based spirit" of the course.

5.1.1 List of Assignments

Recitations and assignments, as well as additional related materials, can be downloaded from the ServEng website. Note that one can also download partial solutions for most of the assignments - these were installed when we moved from groups of 4 to pairs, in order to reduce students' workload. (Professors who would like to experiment with teaching ServEng could write to the first author (AM), regarding further information about the homework.)

1. **On queues.** This introductory assignment requires from students to read some textbook/background materials and present a brief executive summary on "The Future of Queues in Service Systems".

2. **Capacity Analysis. Little's Law.** The assignment contains mostly theoretical questions on Little's Law; some were in fact motivated by empirical scenarios.

3. **Empirical Models.** This is the first data-based homework assignment. Data of a U.S. Banking call center, exported via DataMOCCA [87], is prepared as convenient Excel files that students then use. Answering the questions requires the application of lecture material on the Little's Law, capacity analysis and fluid models.

4. **Processing Networks.** We ask students to create a model of a process network (recall Section 4.3) with which they are familiar. First, an activity precedence diagram should be described, then a resource-based representation should be given and, last, the two representations must be integrated into the combined diagram of the activities and resources. An information flow diagram has been optional in recent years. In addition, students are required to speculate on the Service Engineering of their model, based on their diagrams.

5. **SEEStat Analysis.** Students download an educational version of SEEStat (the GUI of DataMOCCA), with which they explore part of the U.S. Bank database. They search for answers to several practically-oriented questions, redoing (at the push of a button) some of the exercises that they worked hard over, in the previous Empirical Models homework. We elaborate on this homework later on in Section 5.1.2.

6. **Fluid Models.** The students address several questions, related to call-center applications, via the fluid approach: they write the proper differential equations, solve them in Excel using the finite-difference method, and finally apply the Solver to optimize some aspect of the operation (e.g. SBR, overtime management). The homework is covered in detail in Section 5.1.2.

7. **Statistical Analysis of Arrivals.** This homework includes some statistical question (e.g. practical testing of the Poisson hypothesis) and introductory-level forecasting problems.

8. **Service Processes and Empirical Analysis of Customers' (Im)Patience.** Some questions here are purely theoretical and others, on abandonment behavior, require the interpretation of real-data from call center.

9. **Gazolco's Call Center.** As mentioned, 4CallCenters (4CC) is software [26] that supports the application of various Markovian queueing models (Erlang-C, Erlang-B, Erlang-A and others). 4CC is a valuable tool, serving as a personal WFM tool that students actually use also after graduation. It is practiced via operational problems of the fictitious Gazolco call center, which is an assignment that helps students develop their intuition on practical call center challenges.

10. **Theoretical Analysis of Service Stations in Steady State; Priority Queues.** This homework includes a small case study on pizza delivery, adapted from [23] . It is followed by some queueing-theoretical questions.

11. **Staffing of Call Centers.** This last homework (in essence, the final project) is based on actual call center data (we have used various call centers across different semesters). It asks the students to apply a variety of tools they acquired during ServEng lectures. In one version of the homework, data from three call centers was used: a small Israeli call center, for applying standard queueing models; a large U.S. call center, that calls for QED analysis; and a medium Italian call center, which went through an interesting process of "hiring and firing".

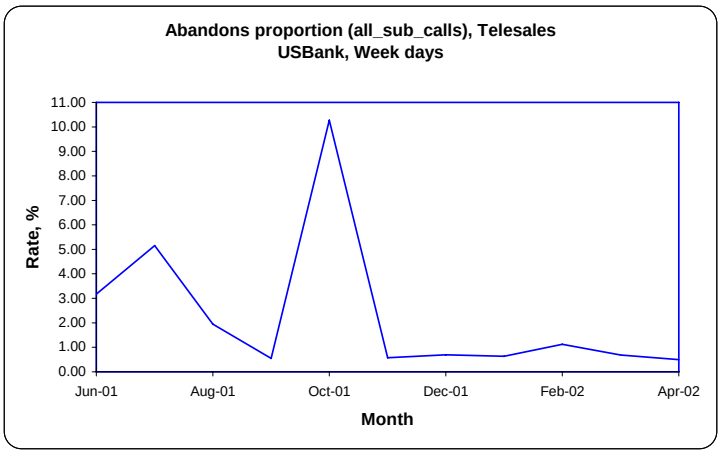
5.1.2 Expanding on Some of the Homeworks

This section contains a detailed coverage of two homeworks: SEEStat Analysis of USBank and Fluid Models. The first homework is chosen since it provides an excellent illustration to our data-based approach. The homework on Fluid Models teaches very useful techniques that are rarely covered in the courses on Services or Operations Research (Queueing Theory).

Example of a Data-Based Homework: SEEStat Analysis

Homework 5, on SEEStat Analysis of USBank, provides students with the opportunity to dive into real data from a (relatively complex) service system, explore problems that arise in call-center practice and apply knowledge that they acquired in lectures. We now review some questions from this assignment.

Figure 35: **Monthly Abandonment Rate of Telesales Calls**



Question 1. Undesirable Service Level. Figure 35 depicts monthly abandonment rates in U.S. bank. It shows a peak at the Telesales queue, in October 2001. We observe that the average abandonment rate of this month reaches 10% - several times higher than the yearly average. Students are requested to use SEEStat in analyzing this peak, showing all the relevant graphs that they use in the analysis, and explaining the information they deduce from each graph.

Solution of Question 1. In order to explain the phenomena, observed in Figure 35, one should explore operational characteristics of the call center at finer time scales. First, using SEEStat, we check the daily abandonment rate of the Telesales calls during October 2001; see Figure 36. Unusually large abandonment rates on Tuesday, October 9 and on Wednesday, October 10, are observed; for example, the fraction abandoning exceeds 45% on October 10.

In the next step of the analysis, October 2001 daily arrival rates of Telesales calls are shown (Figure 37). We note a very large number of calls on October 9 and 10: the usual weekday arrival rate has approximately doubled. One of explanations of this phenomena is that October 9 and 10 take place after an "extended weekend" (Saturday, Sunday and Columbus day on Monday, October 8). There can be also other unknown reasons for the peak arrival-rate, say a large-scale tele-marketing promotion.

Let us check if the number of agents on these heavily-loaded days was increased accordingly. Figure 38 shows the daily number of agents in October 2001: we observe that between October 9 and October 11, it is higher than on other days of this month. However, this increase is not significant enough to compensate for the double-fold increase in the arrival rate.

Figure 36: **Daily Abandonment Rate of Telesales Calls: October 2001**

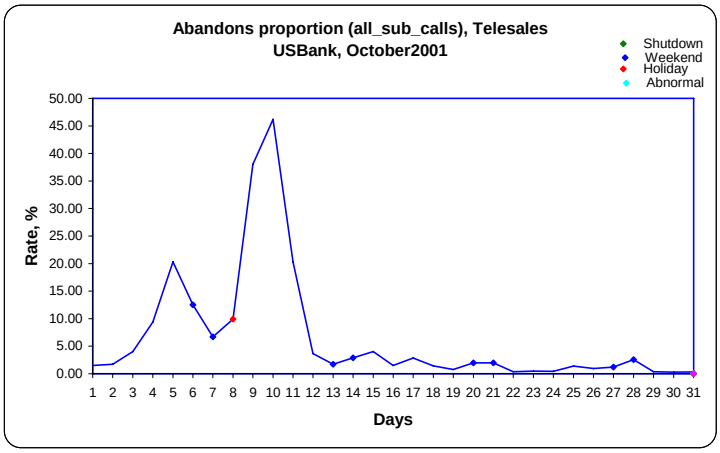


Figure 37: **Daily Arrival Rate of Telesales Calls: October 2001**

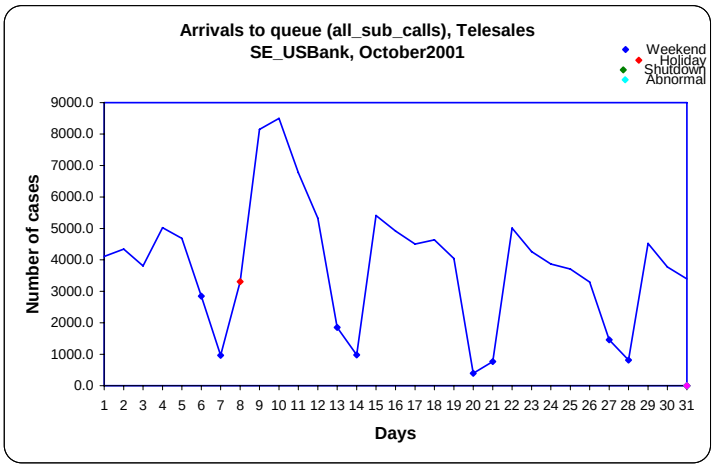


Figure 39 demonstrates the outcome: abandonment rate on October 10 is close to 50% most of the day. Hence, the somewhat increased number of agents on October 9-10 is insufficient for sustaining a reasonable service level.

Question 2. The Influence of an Irregular Event. Figure 40 shows the overall intraday call volume on the Tuesdays in September 2001. Naturally, we know the cause of the small number of arrivals on September 11, 2001; students should try to find out in the database the reason for the discrepancies between the other three days. Next, based only on Septembers' Tuesdays information, students should create (simple) forecasts for the number of calls on

Figure 38: **Daily Number of Agents for Telesales Calls: October 2001**

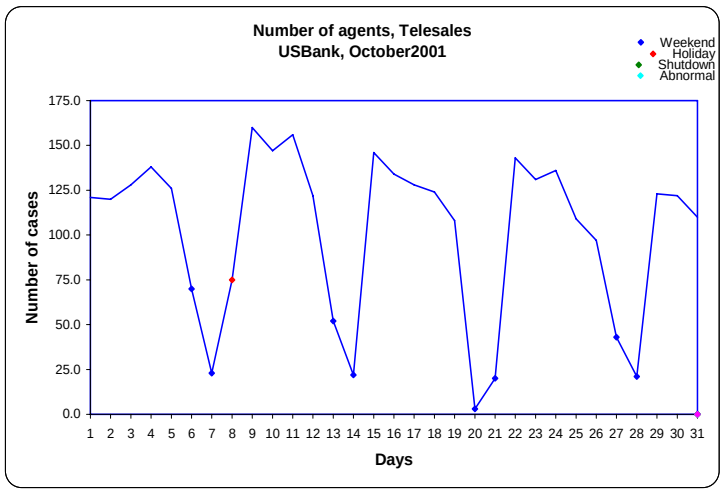
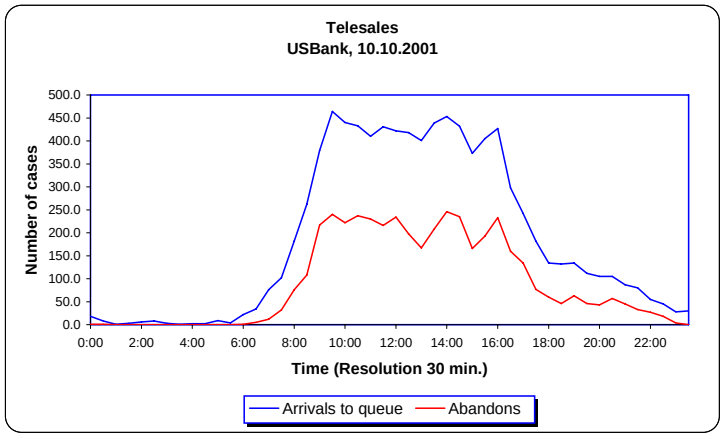


Figure 39: **Intraday Arrival and Abandonment of Telesales Calls: Wednesday, October 10, 2001**



October 2, 2001, and October 9, 2001 (the first and second Tuesdays of October 2001), during the time period 10:30-11:00.

Solution of Question 2. Figure 41 shows that Monday, September 3, 2001, was a holiday (the red point indicates that). Therefore, Tuesday, September 4, 2001, was the first workday after a long weekend, and this is the reason for the heavy load on September 4.

In order to forecast the arrival rate on the first and the second Tuesdays in October 2001, one must first check the calendar for any holidays on these days or prior to them.

Figure 40: **Arrivals Rate on Tuesdays in September 2001**

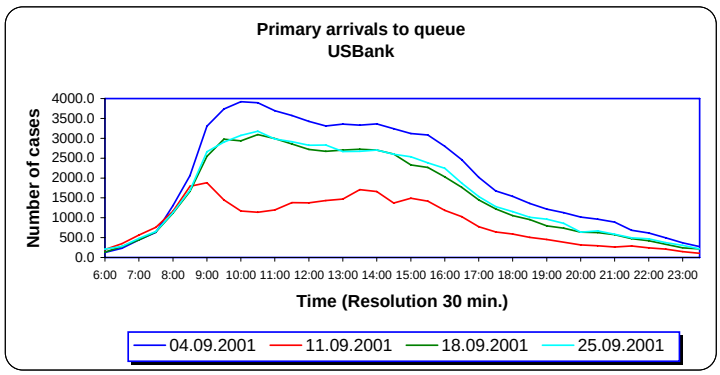
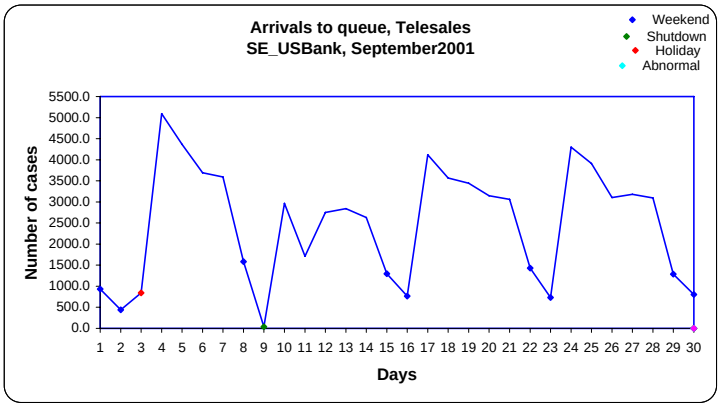


Figure 41: **Daily Call Volumes in September 2001**



SEEStat provides monthly calendars; see Figure 42. We observe that there are no special days on or before Tuesday, October 2, 2001, so our forecast is based on normal Tuesdays in September 2001, namely September 18 and September 25. (Here we actually *assume* that these Tuesdays are normal - after all, the September with September 11 in it, is not a normal September.) The average arrival volume for the 10:30-11:00 interval is 3137. In contrast, Tuesday, October 9, is the first workday after a "long weekend". Therefore the data of a similar Tuesday, September 4, can be used for the forecast: the expected arrival volume is 3897 during 10:30-11:00.

Question 3. Service Time Distribution. Students are requested to draw a histogram of the Customer Service Time for Retail calls, during the weekdays of February 2002. They should identify a theoretical distribution that can fit best the actual distribution of these services times. If the empirical distribution has significant differences with the suggested distribution, they should explain why these discrepancies are or are not relevant.

Figure 42: **Calendar, October 2001**

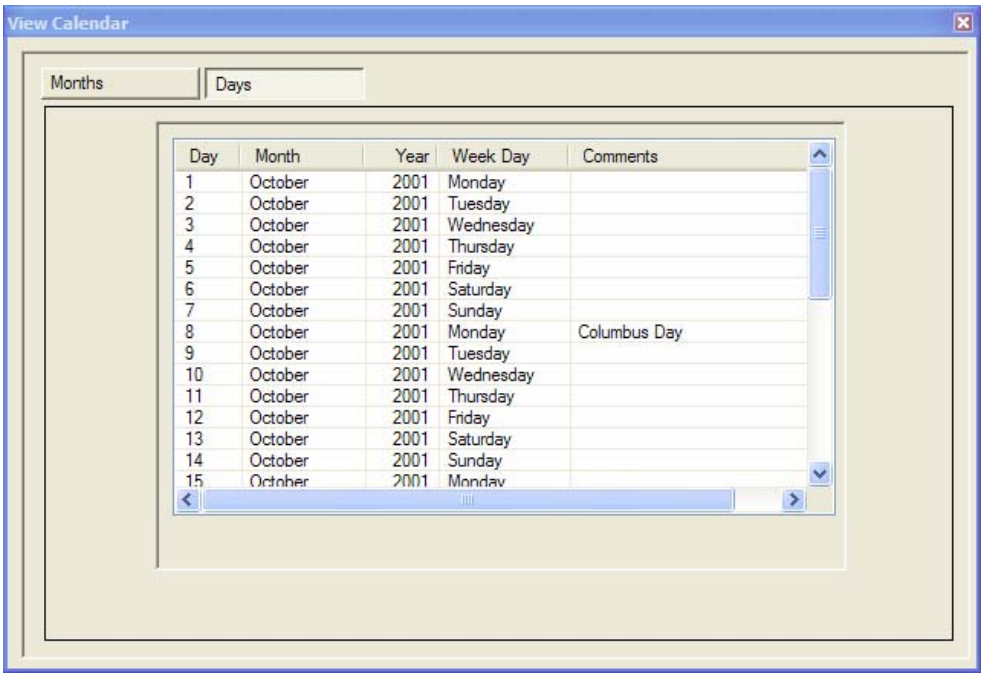
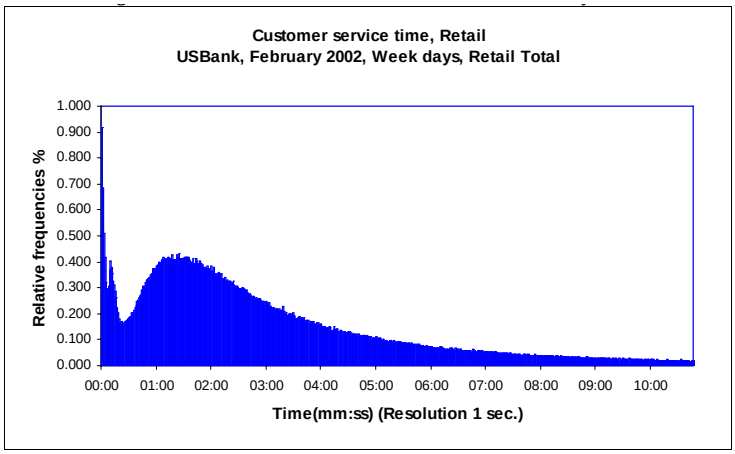


Figure 43: **Histogram of Customer Service Times. Retail Calls**

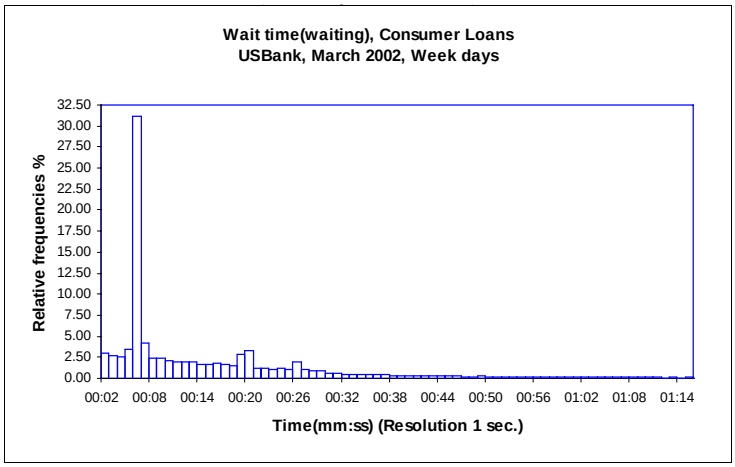


Solution of Question 3. Figure 43 displays the requested service-time histogram. Ignoring service times of duration less than 15 seconds, the empirical distribution is very similar to a lognormal distribution. (This can be checked by SEEStat.) The reason for ignoring small service times is that a service that lasts less than 15 seconds is reasonable to attribute to some

outlier phenomenon (recall Section 4.5.2) and not to a normal service; indeed, 15 seconds is too short time for a voice service.

Question 4. Network Balancing Protocol and Performance Level. With the goal of balancing the queues across the nodes of the (integrated) call center (New-York, Boston, Philadelphia), the ACD routes callers according to the following protocol. (Recall Section 4.8.) When a call has waited a pre-determined time in its original node, it is placed in an inter-queue, from which the call can be answered at any node. The waiting time distributions during March 2002 should be used in order to identify these pre-determined times, for the Consumer Loans calls.

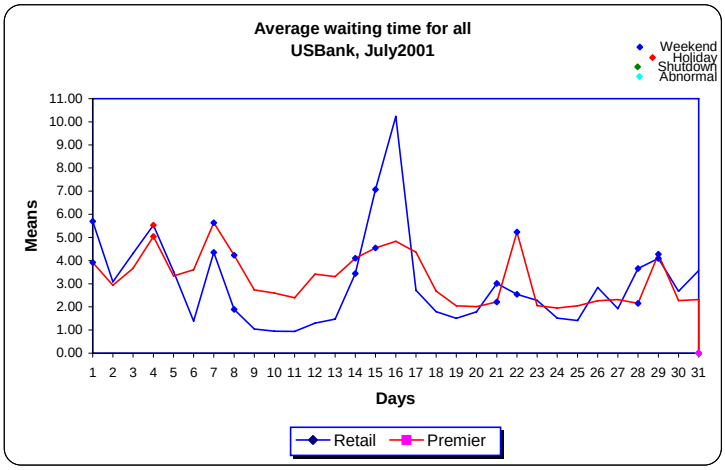
Figure 44: **Average Waiting Time for Customer Loans Calls: Week Days, March 2002**



Solution of Question 4. Figure 44 presents the distribution of waiting time for the Customer Loans calls on weekdays, during March 2002. We observe a peak at 7 seconds. Therefore, after 7 seconds of wait, the Customer Loans customers were placed in the inter-queue.

Question 5. Service Levels and Priority Queues. Retail and Premier calls belong to the same service type, but the Premier calls are considered VIP calls. Despite the fact that the agents of Retail and Premier services have the same skills, the agents defined as "Retail agents" are not qualified to serve Premier calls; but "Premier agents" are in fact qualified to serve Retail calls.

Figure 45: **Average Waiting Time for Retail and Premier calls, July 2001**



The marketing manager of the call center claims that service levels are not satisfactory on July 2001. Students should explain the reasons for the marketing manager claim, prepare the graphs to support their explanation and suggest one or several ways to solve the problem.

Solution of Question 5. Compare Retail and Premier service levels, considering average waiting times and probability to be served immediately; see Figures 45 and 46. We observe that the lower-priority customers (Retail) enjoy better service, which is certainly not predesigned. Focusing on the number of agents of both service types (Figure 47), we observe that there are less than 70 Premier's agents, and more that 600 Retail's agents on weekdays. Hence, lack of EOS (Economies-of-Scale) seems to be responsible for the insufficient service level of VIP customers.

Potential ways to improve the situation include:

- Enable all Retail and Premier agents to serve all Retail and Premier calls. This change must be integrated with a corresponding SBR Policy and, probably, training courses.
- Transfer some of the "Retail agents" to "Premier agents". This change must be integrated with a corresponding SBR Policy.
- Change the SBR policy. For example "do not route Retail call to Premier agents if there are less than 10 idle Premier agents".

Figure 46: **Fraction of Immediately Answered for Retail and Premier Calls, July 2001**

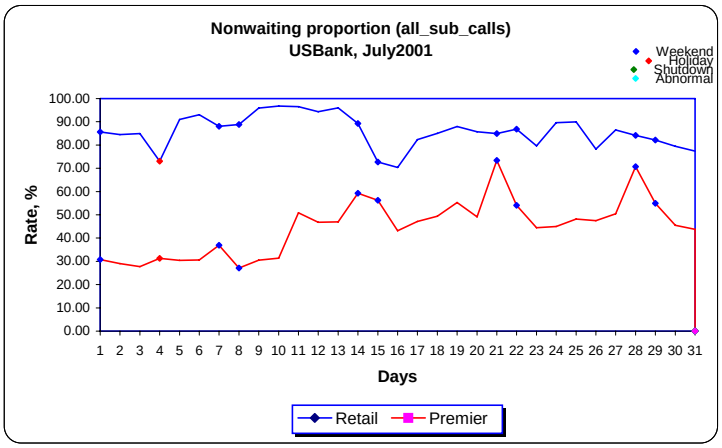
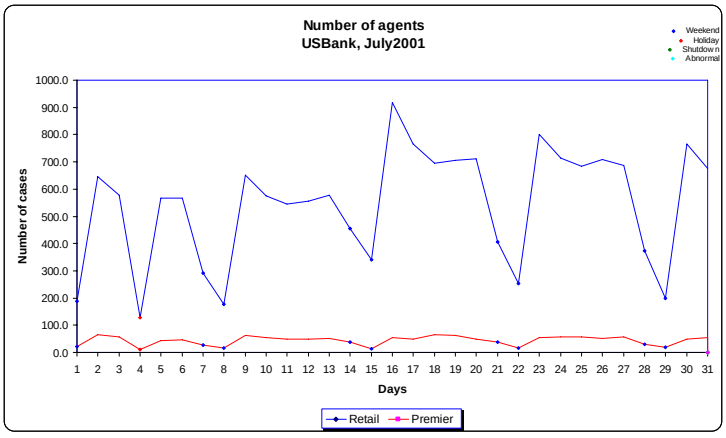


Figure 47: **Number of Agents for Retail and Premier calls, July 2001**



Fluid Analysis, or "Simple Models at the Service of Complex Reality"

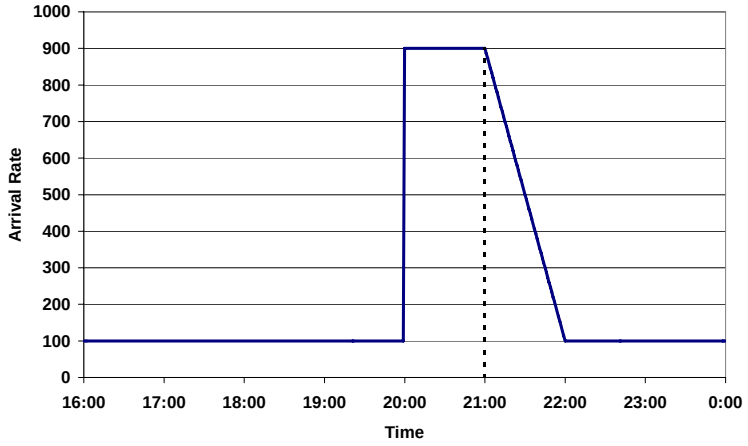
This homework trains students in the ODE/Spreadsheet approach to Fluid Models, as discussed in Section 4.4. Below we cover most questions of this homework.

Part 1

In Part 1 of the homework we introduce Tele-SHOP, a commercial channel dealing with online sales, which is operated by a call center. Assume that, in order to increase profits, it was decided to place a short (30 seconds) daily advertisement on national TV, at 20:00.

Consequently, a sharp rise in the arrival rate after 20:00 is expected. Assume that the arrivals to the call center are well approximated by an inhomogeneous Poisson process, with the arrival rate function (customers per hour) given by Figure 48.

Figure 48: **Arrival Rate Function to Tele-SHOP Call Center**



The call center operates from 16:00 till 24:00. The service duration of each incoming call has an average of 12 minutes. Assume that the working hour of a service representative costs 36 shekel[1], while a minute of waiting of a customer costs 1 shekel. (A lower bound for the latter cost would be toll-free costs.)

Then the cost of running the call center is given by

$$C^{(N)} = \int_0^T \left[h_t(Q(t) - N(t))^+ + c_t N(t)\right] dt\,, \tag{10}$$

where

- $Q(t)$ - number in system (served + queued) at time t, $0 \le t \le T$,
- $N(t)$ - number of servers at time t,
- c_t - staffing cost per unit of time per server[2],
- h_t - waiting cost per unit of time per customer[3].

[1] Israeli monetary unit

[2] The units of c_t are monetary units per one time-unit of work of one server (salary)

[3] The units of h_t are monetary units, eg. shekels, per unit of waiting time, i.e. $\frac{\text{shekels}}{\text{waiting customer} \times \text{time unit}}$

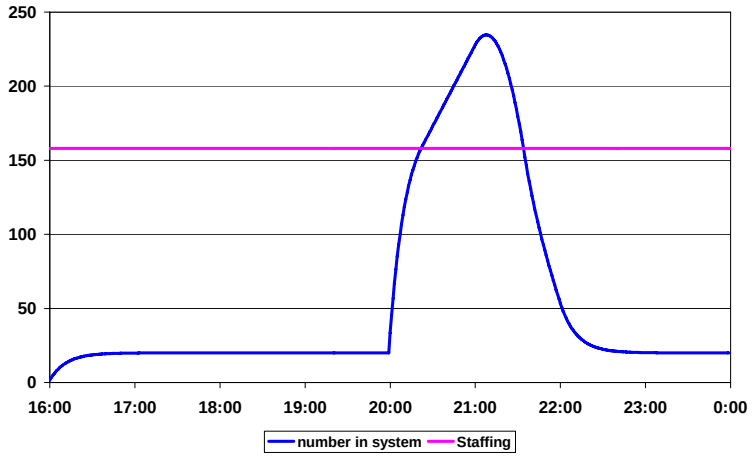
The system starts empty and in all questions of the homework we assume that the queue must be empty at the end of the day. Students should answer most of the following questions using Excel Solver.

Question 1. Assume that the number of servers must be fixed during the working day and that the tele-customers do not abandon. Find the optimal staffing (i.e. fixed staffing that minimizes the costs of running the call center) and calculate the minimal cost.

Solution of Question 1. Consider a queueing system with one class of customers and one service station. Define:

- $\lambda(t)$ - instantaneous arrival rate at time t,
- μ - service rate (service capacity of each server), constant in time,
- $N(t)$ - number of servers in the system at time t,
- $Q(t)$ - total number of customers in the system, (being served + queued) at time t.

Figure 49: **Number in System under Optimal Fixed Staffing**



Then, according to the fluid approach, the number in system satisfies the following ODE:

$$\frac{d}{dt}Q(t) = \lambda(t) - \mu \cdot \Big(Q(t) \wedge N(t)\Big), \quad Q(t_0) = Q^0, \quad t \in [t_0, T]. \tag{11}$$

The corresponding finite-difference approximation is constructed in the following way. Choose time resolution Δt (we use one minute). For the time interval in consideration (16:00-24:00), let $t_0 = 16{:}00$, $t_n = t_{n-1} + \Delta t, \quad n = 1, 2, ...$ Then

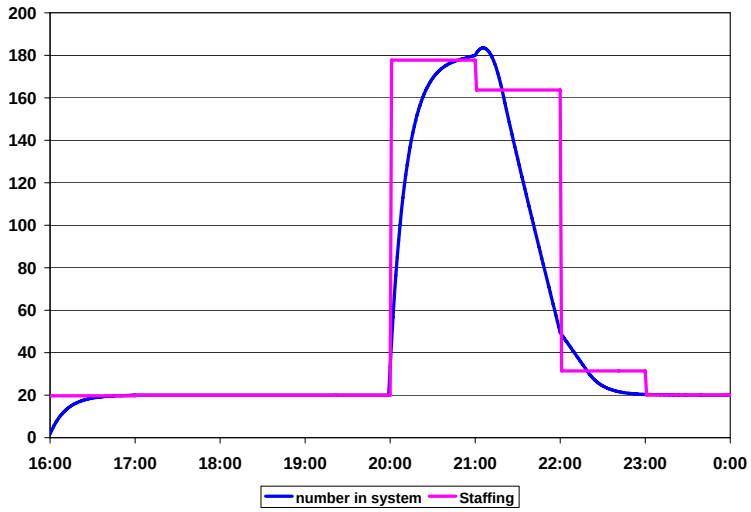
$$Q(t_n) = Q(t_{n-1}) + \lambda(t_{n-1}) \cdot \Delta t - \mu \cdot \Big(Q(t_{n-1}) \wedge N(t_{n-1})\Big) \cdot \Delta t. \tag{12}$$

This recursive equation can be easily implemented in EXCEL. It approximates the cost (10) (taking the staffing cost per minute $c = 36/60 = 0.6$), which can now be optimized via the EXCEL Solver.

Optimal solution: $N^* \approx 158$, Cost=48,763, Figure 49 shows the dynamics of number-in-system.

Question 2. Solve Question 1 under the assumption that the manager is allowed to change the staffing at the beginning of each hour (*shift staffing*).

Figure 50: **Optimal Shift Staffing**



Solution of Question 2. Use (12) where $N(t)$ can be changed each hour. Optimize the vector of eight hourly staffing levels via EXCEL Solver.

The optimal staffing $N^* \approx (20, 20, 20, 20, 178, 164, 31, 20)$ (see Figure 50), Cost=17,502.

Note that the shift staffing (staffing flexibility) enables a drastic cost decrease. The optimal staffing increases with the arrival peak at 8pm, remains almost at the same level between 8 and 10pm, and decreases significantly at 10pm. Note that during five hours the staffing remains at the equilibrium level $N = 20$, with the arrival rate equal to the total service rate.

Question 3. From now on assume that upon each service completion, the system gets a reward of r shekel. Assume that waiting customers can abandon the tele-queue and the abandonment rate equals θ (i.e. each waiting customer abandons after the average time of $1/\theta$ hours, if not admitted to service before). In addition, assume that, *instead of waiting costs,* there exists a penalty of s shekel per each abandoning customer. If not stated otherwise, we assume that $s = 10$ shekel and $\theta = 3$.

Write the expression for the total profit of the system.

Solution of Question 3. The total profit is equal to

$$C^{(N)} = \int_0^T \Big[r \cdot \mu \cdot (Q_t \wedge N_t) - s \cdot \theta \cdot (Q_t - N_t)^+ - cN_t \Big] dt.$$

Question 4. Determine the optimal *fixed* staffing and the maximal profit for $r = 50, 100, 200$. What kind of behavior do you expect from the *fixed* optimal staffing level N^* as r increases to infinity? Does N^* converge to some limit? If it does, what is the limit?

Solution of Question 4.

The differential equation for the number in system $Q(t)$ is given by

$$\frac{d}{dt}Q(t) = \lambda(t) - \mu \cdot \Big(Q(t) \wedge N(t) \Big) - \theta \cdot \Big(Q(t) - N(t) \Big)^+.$$

We transform it to the finite-difference equation in the same manner that (11) was transformed to (12).

For $r = 50$: optimal staffing $N^* = 151$, Profit=50,434. See Figure 51.

For $r = 100$: optimal staffing $N^* = 171$, Profit=148,040.

For $r = 200$: optimal staffing $N^* = 177$, Profit=346,350.

What if we take $r = 10,000$? Then $N^* = 179$. See Figure 52.

How to characterize the limit? As the reward-per-call r increases, then one **can not afford to miss anyone**. The optimal **fixed** staffing is the minimal staffing that enables each incoming customer to be immediately admitted to service:

$$N^* = \max_{t \in [0,T]} R(t), \quad \text{where} \quad \frac{d}{dt}R(t) = \lambda(t) - \mu \cdot R(t).$$

In other words, $R(t)$ is the number of people in a system with an infinite number of servers.

Question 5. Assume $r = 60$ and $s = 0$ (no penalty for abandoning customers). Compute the optimal *fixed* staffing and the maximal profit. Comment on your results.

Solution of Question 5. We get optimal staffing $N^* = 151$, Profit=70,318. Note that the optimal staffing is identical to the case $r = 50$ and $s = 10$. This equivalence prevails since in both cases we lose (or, which is the same, do not earn) 60 shekel per each abandoning customer. It can be proved rigorously that if the system is empty at the beginning and at the end, optimal staffing depends on $(r + s)$ only.

Question 6. Instead of giving one 30-second advertisement at 20:00, it was decided to place two 15-second advertisements at 18:00 and at 21:00. The resulting arrival rate is shown in Figure 53. (The arrival rate starts to decrease at 18:30 and 21:30.) Assume $s = 10$ again.

Figure 51: **Reward-per-Call $r = 50$**

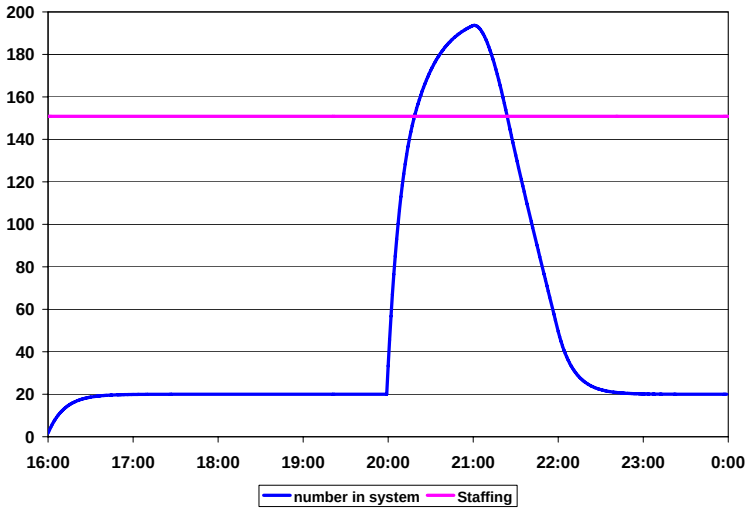
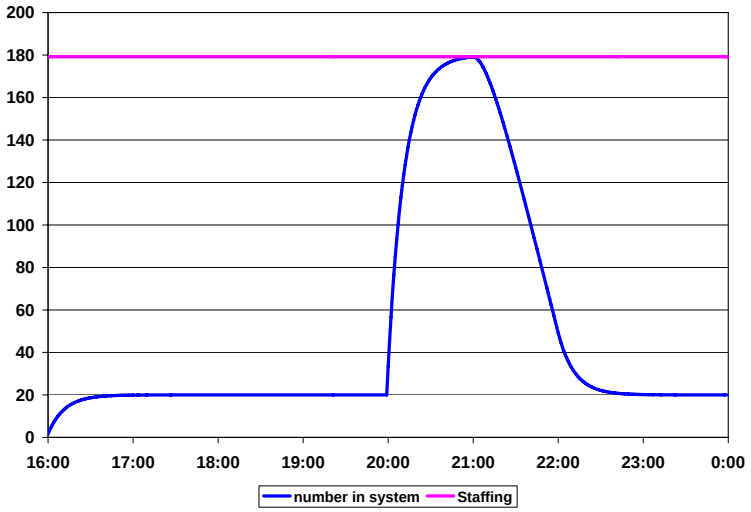


Figure 52: **Reward-per-Call $r = 10000$**



For the new arrival rate, determine the optimal *fixed* staffing N^* for $r = 100$ and 200. Compare the maximal profits for the two advertising policies. Try to explain your findings.

Solution of Question 6.

For $r = 100$: optimal staffing $N^* = 156$, Profit=152,314. See Figure 54.

For $r = 200$: optimal staffing $N^* = 164$, Profit=350,784.

We observe that the optimal profit is larger than in the case of one long arrival peak. It turns out that less servers are needed in order to deal with short arrival peaks.

Figure 53: **New Arrival Rate Function**

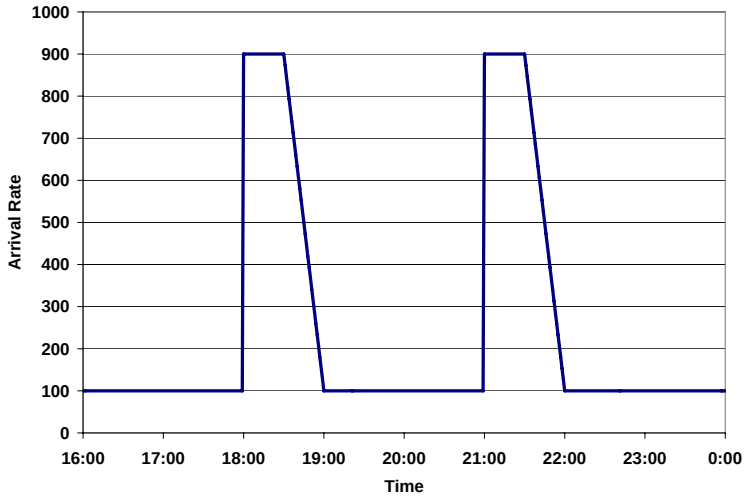
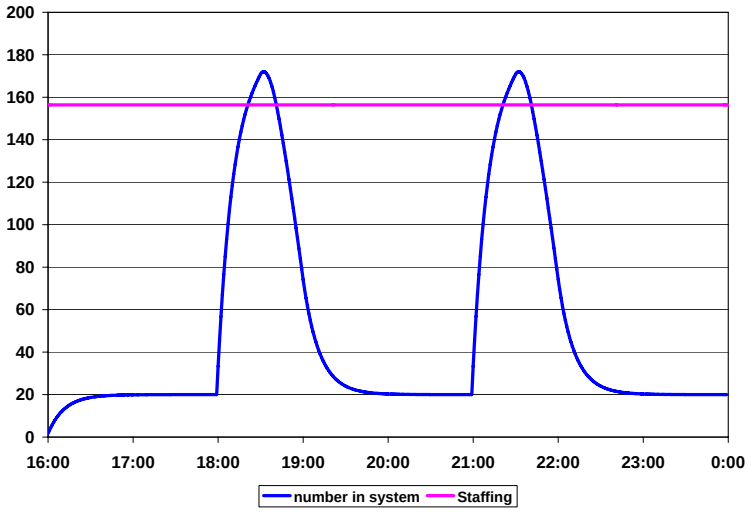


Figure 54: **New Arrival Rate Function. Reward-per-Call $r = 100$**



Question 7. In Question 6, what is the limit of the optimal staffing level N^* as r increases to infinity? Compare your results with Question 4. Do you think that $N^* = \max_t \frac{\lambda(t)}{\mu}$ is a correct answer? If not, can you provide a theoretical expression for the limit value of N^*?

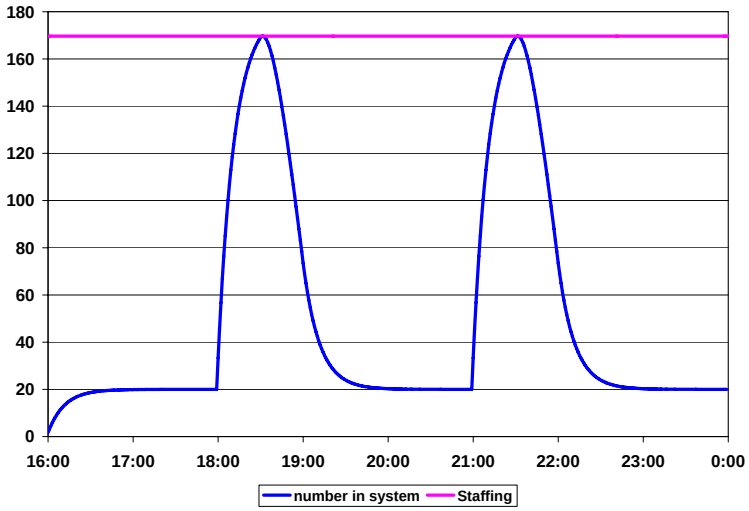
Solution of Question 7.

First, take $r = 10,000$ and get the optimal staffing $N^* = 170$ (Figure 55). As explained in the solution of Question 4,

$$N^* = \max_{t\in[0,T]} R(t), \quad \text{where } \frac{d}{dt}R(t) = \lambda(t) - \mu \cdot R(t).$$

For the new (double peak) arrival rate, N^* is significantly smaller than $\max_t \frac{\lambda(t)}{\mu} = 180$: there is no need to fix the staffing level at the maximal instantaneous load, since relatively short peaks can be served by a smaller number of agents.

Figure 55: **New Arrival Rate Function. Reward-per-Call $r = 10,000$**



Question 8. Assume $r = 100$. Compare the two advertising policies for one-hour shift staffing.

Solution of Question 8.

For the old arrival rate:

$N^* \approx (20, 20, 20, 20, 179, 176, 39, 20)$, Profit=180,798, see Figure 56.

For the new arrival rate:

$N^* \approx (20, 20, 168, 55, 20, 168, 54, 20)$, Profit=179,845, see Figure 57.

In the second case, one-hour shift staffing implies four hours with an increased staffing (two for each peak) versus three hours in the first case. Therefore, in contrast to Question 6, optimal profit is larger for the old arrival rate.

Question 9. For the optimal advertising policy and the optimal one-hour shift staffing computed in Question 8, calculate the proportion of customers, that abandoned the system without being served.

Solution of Question 9:

Figure 56: **Old Arrival Rate. Optimal Shift Staffing**

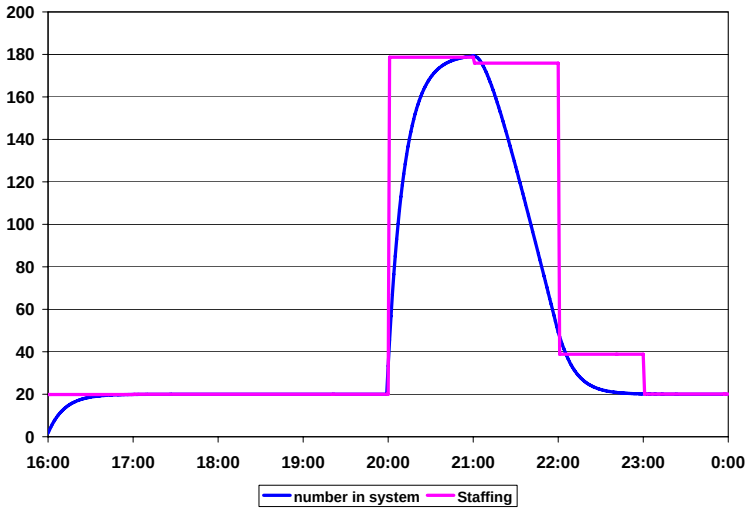
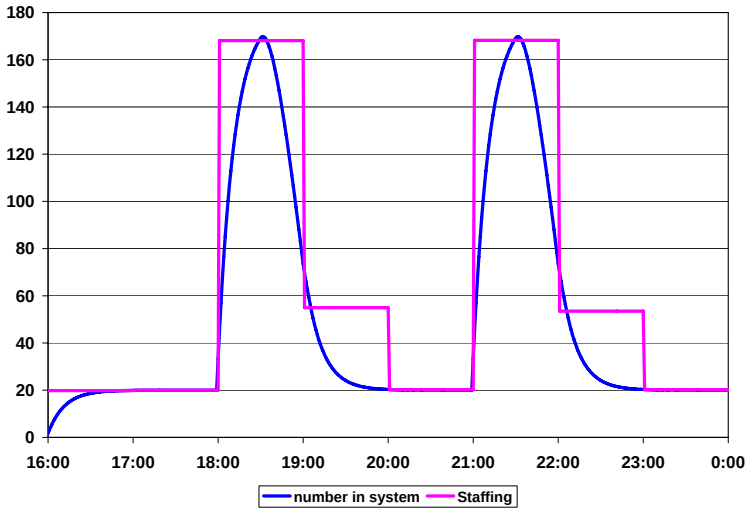


Figure 57: **New Arrival Rate. Optimal Shift Staffing**



Consider the old arrival rate.

$$\text{Prob}(abandon) = \frac{\text{total abandoned}}{\text{total arrived}} = \frac{\int_0^T \theta(Q(t) - N(t))^+ dt}{\int_0^T \lambda(t) dt} \approx \frac{2.5}{2007} = 0.12\%.$$

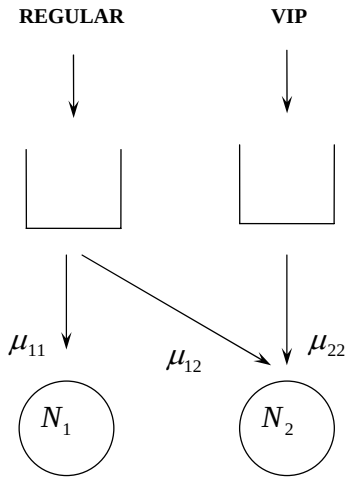
Part 2. N-model

From now on assume that there are two classes of customers (calls): regular (class 1) and VIP (class 2), that arrive to the call center. The arrival rates of two classes $\lambda_1(t)$ and $\lambda_2(t)$

are given by Figure 48 (for regular) and Figure 53 (for VIP). There is no abandonment in the system.

In addition, the call center is virtually divided into two stations 1 and 2 with the corresponding fixed staffing levels N_1 and N_2 (see Figure 58).

Figure 58: **The Structure of the N-Model**



Servers at station 1 can handle only regular customers with the average service duration of 15 minutes. Servers at station 2 can handle both classes with the average service duration of 12 minutes for class 1 and 10 minutes for class 2. A working hour of a service representative costs 36 shekel for station 1 and 48 shekel for station 2. A minute of waiting of a customer costs 2 shekel for VIP and 1 shekel for a regular customer.

Assume that the call center works in the *preemptive-resume regime,* i.e., at every moment a service to a customer can be interrupted (in this case a customer goes back to queue of its class) and resumed at a later time. Moreover, assume that the VIP customers are *high priority customers*, which means that no regular customer can be in service at station 2 while VIP customer is waiting.

Define $Q_i(t)$ to be the *total* number of class - i customers in the system, $i = 1, 2$. Let $Q(t) = Q_1(t) + Q_2(t)$. Assume $Q(0) = 0$.

Question 10. Assume two operating policies for regular customers:

- **First policy.** If both stations can serve regular customers, they are routed to Station 1 (both from the queue and from service at Station 2),

- **Second policy.** If both stations can serve regular customers, they are routed to Station 2 (both from the queue and from service at Station 1).

Write the differential equation for $Q_1(t)$ and $Q_2(t)$ for both policies.

Solution of Question 10:

First policy:

$$\frac{dQ_1}{dt} = \lambda_1 - \mu_{11}\Big(Q_1 \wedge N_1\Big) - \mu_{12}\Big[(Q_1 - N_1)^+ \wedge (N_2 - Q_2)^+\Big]$$

$$\frac{dQ_2}{dt} = \lambda_2 - \mu_{22}\Big(Q_2 \wedge N_2\Big).$$

Second policy:

$$\frac{dQ_1}{dt} = \lambda_1 - \mu_{12}\Big(Q_1 \wedge (N_2 - Q_2)^+\Big) - \mu_{11}\Big[\Big(Q_1 - (N_2 - Q_2)^+\Big)^+ \wedge N_1\Big]$$

$$\frac{dQ_2}{dt} = \lambda_2 - \mu_{22}\Big(Q_2 \wedge N_2\Big).$$

Question 11. For the two policies, defined in Question 11, determine the optimal fixed staffing for both stations (i.e., two staffing levels N_1^* and N_2^* that minimize the costs of running the call center). Which policy implies the lower operating costs?

Solution of Question 11:

Figure 59: **First Policy: regular customers are routed to station 1**

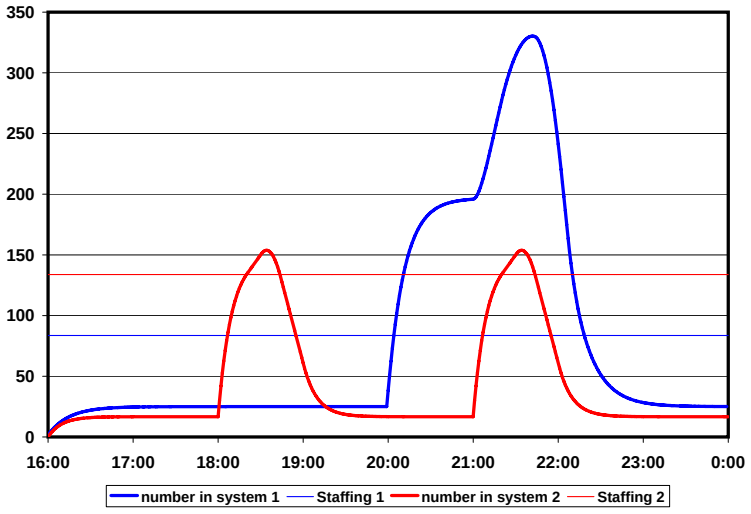
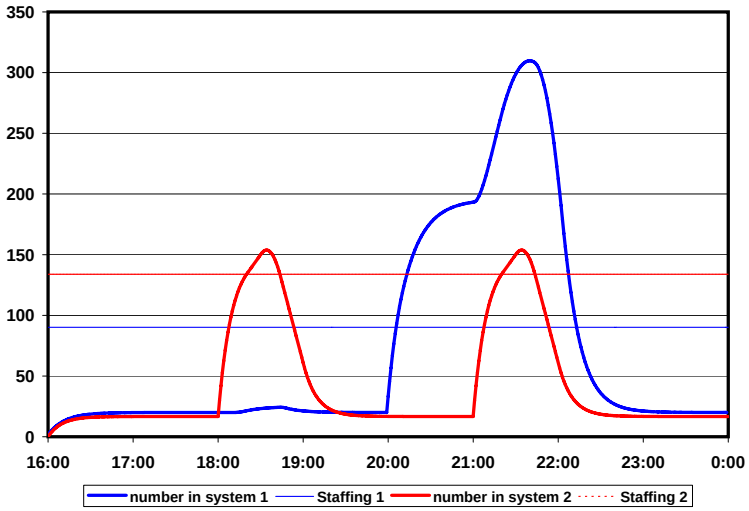


Figure 60: **Second Policy: regular customers are routed to station 2**



For the first policy we get $N_1^* = 84$, $N_2^* = 134$ and overall cost is 94,735, see Figure 59.

For the second policy $N_1^* = 90$, $N_2^* = 134$ and overall cost is 94,006, see Figure 60. The second policy turns out to be better.

5.2 The Final Exam

Until several years ago, exams were very comprehensive, requiring a labor-intensive preparatory process. All exams+solutions appear on our website, but they are written in Hebrew. An English-version example of such an exam, with its solution, appears in [19].

The second generation of our exams is of a different form, which is more structured and hence easier to prepare. The exam consists of four questions (for a total of around 50 points = the weight of the exam in the final grade):

1. **Homework Question:** taken from the homework, and worth about 10-15 points. This question verifies that students were not free-riders when preparing their group homework.

2. **Recitations+Lectures Question:** taken from lecture-notes provided in recitations and lectures, and worth about 10-15 points. This question encourages class attendance, which enhances the learning experience (and which students too often tend to neglect).

3. **Practical Question:** asking the students to actually analyze data, e.g. identifying operational regimes and some of their properties, which is worth about 15 points. (The

students do not need a computer to work on this question: it is composed to be solved with paper and pencil.) Since in their future work most students will encounter real data, this question in fact tests what they are actually carrying with them after graduation.

4. **Theoretical Question:** that asks students to prove or interpret part of theoretical fact (e.g. Khinchine-Pollatzchek, or Biased Sampling from PASTA, or something similar to a proof from class), worth about 10 points. This question is a pre-requisite for getting one's final-grade up to the level of 90-100. Students who perform well in this question often continue to graduate school (or are already there).

6 Service Engineering: Fusing Research and Teaching

During the years that the ServEng course has been taught, fruitful and bilateral relations were created and sustained between various research projects and our teaching program. First, our course is supported by a large body of practically oriented research. Material from academic papers is adapted to the needs of undergraduate students, and frequently used in lectures. In addition, quite a few ServEng students continued to pursue ServEng related work: either in undergraduate projects, or at work after graduation or, in the best of all worlds, within MSc or PhD theses. We now provide a sample of such research and projects, classified by topic. Cited papers, theses and projects are all downloadable from the "References" section of our course website [83].

Design of Call Centers: Erlang-A and Extensions. This research builds on works of Palm [76] and Halfin and Whitt [34]. It was started by Garnett, Mandelbaum and Reiman [31] with Erlang-A analysis in the QED regime, then generalized in Zeltyn and Mandelbaum [96] to M/M/n+G. Borst, Mandelbaum and Reiman [9] show optimality of the QED regime for Erlang-C under broad assumptions and Mandelbaum and Zeltyn [67], continuing this research, solve service-level constraint satisfaction problems for M/M/n+G.

Skills-Based Routing. A challenging line of research is that on matching customers and agents, taking into account agent capabilities (cross-trained, specialized) and customers profiles (VIP, regulars). This research includes Mandelbaum and Stolyar [63] (a model with efficiency-driven services), Gurvich, Armony and Mandelbaum [33] (threshold policies in the QED regime), Atar, Mandelbaum and Reiman [5], and Atar, Mandelbaum and Shaikhet [6].

Behavioral Operational Models. Data-Based (Empirical) research in Operations Management is gaining importance, and rightly so. This goes hand in hand with acknowledging psychological aspects that are significant operationally, for example customer-patience, mechanisms that trigger abandonment, preferences as to what information customers seek etc. A fundamental issue here, for which we believe no definite answer is yet available, is the understanding (quantification) of the "Cost of Delay". This is especially significant in (phantom) tele-queues such as waiting at the phone, "conversing" with an IVR or a computer terminal. A related question is the following: given the individual cost of waiting and abandonment triggers, predict the ensuing system (Nash) equilibrium, in particular accounting for learning due to accumulated experience. The above has been explored in Mandelbaum and Shimkin [62], Shimkin and Mandelbaum [85] and Zohar, Mandelbaum and Shimkin [97].

Design of Interactive Voice Response Units. In many call centers, most of customers get end-to-end service via IVR and the others engage IVR at the initial stage of service. Hence, the IVR design is a very important research problem. In Khudyakov, Feigin and Mandelbaum [47], IVR is mathematically modeled, exactly analyzed and finally approximated in the QED regime for additional insights. This work continues at the SEELab, based on extensive IVR data from two call centers.

Predictable Variability and Time-Varying Queues. Many service systems operate over a finite horizon, during which operating characteristics vary predictably with time. In order to account properly for this predictable variability, models must sometimes be transient, which renders impossible their exact analysis. One thus resorts to alternative models, based on the fluid view and its diffusion refinement. This work started in Mandelbaum and Massey [56] and continued in Mandelbaum et al. [59]. One way to cope with predicable variability is to predictably vary staffing levels. It is then possible, via a surprisingly simple adaptation of the square-root staffing rule, to achieve time-stable performance in the face of time-varying demand. This was first done in Jennings et al. [44]; recently, the research has been significantly expanded in Feldman et al. [22], recall Section 4.7.3.

Applications in Health Care. Recently, an Open Collaborative Research project was launched, jointly by the Rambam hospital in Israel, the Technion, and IBM Haifa Research Lab. This is a multi disciplinary project aimed at significant clinical, operational and financial improvement of hospital patient care processes. The outputs of this project include Marmor et al. [70, 69], M.Sc. thesis of Yulia Tzeytlin [88] and more is yet to come. This project helps expand the ServEng course to cover healthcare

applications. (Recall the Emergency Department flowcharts in Section 4.3.)

Statistical Analysis and Forecasting. Call centers have been accumulating vast quantities of data (operational, marketing, survey), but these have been *inaccessible* to academic research, at least at the level of the individual transaction (call-by-call data). The first of a kind comprehensive research of a call center transactional data was performed by Mandelbaum, Sakov and Zeltyn [61] and continued by Brown et al. [10]. Currently, the DataMOCCA software enables such research for several large call centers. Forecasting arrival rate and offered load is another important research issue; here, in addition to [10], we should mention Weinberg, Brown and Stroud [90], and Aldor-Noiman, Feigin and Mandelbaum [3]. A recent M.Sc. thesis of Shimrit Maman [55], advised by the two authors, studies the important problem of staffing under uncertain arrival rate, combining queueing theory research with statistical modeling and insights.

7 Acknowledgements, and a Little More History

The present Service Engineering course at the Technion is the culmination of a long continuous-improvement process. One can view this process as progressing along several fronts:

- *Teaching material*: Most of the material is original, based on research papers of the authors and coauthors, customized teaching notes, graduate theses and students' projects.

- *Recitations and Homework*: These have been developed originally by the authors, and then enhanced, expanded and innovated by the many excellent teaching assistants (TAs) that the course and its students have enjoyed. As already mentioned, homework are central to the learning experience, which is manifested by a weight of about 40-50% of the final grade.

- *Students' Projects*: Undergraduate IE&M student at the Technion must take part in a yearly project, typically carried out during one's last (4th) year of school, with the goal of applying theory to a real-world setting. Similarly, graduate students in the IE&M Operations Research program take a project-course, with the same goal. A full list of projects that the first author has advised appears in [15]. Many of these projects apply tools acquired in the Service Engineering course (mostly to call centers and hospitals) - then excerpts from such projects are often used to demonstrate Service Engineering applications. Other projects constitute pilots that develop into research projects or graduate theses.

For example, Figures 8 – 11 describe the routing process of hospital patients from an emergency department to internal wards. These were first generated within an undergraduate project at the Rambam hospital in Haifa, then continued as the MSc thesis [88], and used as an example for the homework where students are asked to create their own processing network. As another example, Figure 21 was first created within a project at a call center in a large Israeli Bank, it has become part of a research paper that is now in the writing, and it.

- *Website*: Maintenance of the course's website has been the responsibility of the course's TA. While being a high toll in terms of time and effort, the website has been a central enabler for development and maintenance. Indeed, the website serves as an organized repository of course material, active and archival, for the benefits of students, instructor and TA. Also, through the website, course material is accessible to teachers and research-colleagues world-wide (which is the reason that the material has been developed in English, while the Technion course has been taught in Hebrew);

Course TAs: The Service Engineering course owes much of its success to its excellent TAs. These have been carefully chosen and then well-groomed for the job (for example, by re-attending lectures as a future TA, and learning "trade secrets" from the present TA). Our TAs have all been top graduate students, doing research directly related to some course subjects, and willing to undertake the challenging job of a ServEng TA. This entails first the routine yet time-consuming chores of conducting recitations, maintaining the course website, having office hours (which are usually attended due to the challenging homework) and, all in all, help the course maintain a service-level that a Service Engineering course is worthy of. In addition, there is also the creative part of a TA work (which distinguishes the excellent from the merely very-good) - that of generating new material for recitations and homework, and writing questions for exams, with an emphasis given to incorporating the TA's own research into course material. One of the authors (S.Z.) was the first TA for the course in its present form. Then, in chronological order, we would like to acknowledge the significant contributions of Itay Gurvich, Gennady Shaikhet, Shimrit Maman and Galit Yom-Tov.

Students' Contribution: There is a long list of (mostly graduate) students who have contributed to the course material development (sometimes while serving as homework graders). Among these, we would like to especially acknowledge Sivan Aldor-Noiman, Yonit Baron, Izik Cohen, Ofer Garnett, Zohar Feldman, Eva Ishay, Polina Khudyakov, Pablo Liberman, Yariv Marmor, Michael Reich, Luba Rosenshmidt, Arik Senderovich, Yulia Tzeytlin and Asaf Zviran.

The SEELab: Permanent and temporary members of the SEELab [84] have supported the course in various ways, mostly related to data and SEEStat. Here we would like to acknowledge Valery Trofimov, Ella Nadjarov, Igor Gavako, Katya Kutssy and Arik Senderovich.

Research Partners: When teaching, especially a young subject, it is useful for teachers to reflect and students to study the subject's evolution. This is also an opportunity to acknowledge those who contributed to that evolution. An example is the overhead in Figure 61, taken from a lecture on Queues with Impatient Customers: it describes the "Modeling History" of such queues, as inspired by Call Centers; it also lists the models and their developers, who have been our coauthors and partners, and to whom we are highly grateful. Names of graduate students are highlighted in the slide, this in order to stimulate the curiosity and appetite of course students, towards projects and graduate studies in the field of Service Engineering.

Adminstration Support: Special thanks is due to the current dean of the IE&M Faculty at the Technion, Professor Boaz Golany, who followed closely the development of the course and spared no resources to nurture it.

Mini-Courses: Mini-versions of the course have been taught at several universities: Wharton (hosted by Larry Brown, Noah Gans and Morris Cohen), Columbia GSB (hosted by Awi Federgruen and Ward Whitt), Stanford GSB (Mike Harrison and Sunil Kumar) and INSEAD (hosted by Zeynep Aksin). These mini-courses gave us the opportunity to share and learn from our hosts and their students - their generosity, along all dimensions, is greatly appreciated.

Subsequent Courses to Service Engineering: Advanced Service Engineering courses are occasionally taught at the Technion (thought not at the frequency we desire, due to scare resources). They either focus on a specific application (a recent seminar was dedicated to healthcare), or on advanced theoretical techniques (e.g. fluid and diffusion approximations of queueing systems). In addition, the first author (AM) is presently leading a multidisciplinary seminar in Services, attended by both faculty and (mostly honor) students. (IBM provided summer scholarships for some of the students, who were engaged in preparing the seminar.) In this seminar, faculty from various disciplines, e.g. Marketing and Psychology, will lecture on "Services" from the view-point of their discipline. The seminar is then planned on giving rise to multi-disciplinary undergraduate projects: for example, jointly Operations & Information System (on RFID-based measurements in hospitals), Operations & Psychology (on actual and perceived workload in services), and more. These projects will be jointly supervised by faculty from the corresponding disciplines.

Figure 61: **Acknowledgement to Partners, Coauthors and Students**

Call Centers = Q's w/ Impatient Customers 14 Years History, or "A Modelling Gallery"

1. Kella, Meilijson: Practice ⇒ Abandonment important
2. Shimkin, Zohar: No data ⇒ Rational patience in Equilibrium
3. Carmon, Zakay: Cost of waiting ⇒ Psychological models
4. Garnett, Reiman; Zeltyn: Palm/Erlang-A to replace Erlang-C/B as the standard Steady-state model
5. Massey, Reiman, Rider, Stolyar: Predictable variability ⇒ Fluid models, Diffusion refinements
6. Ritov; Sakov, Zeltyn: Finally Data ⇒ Empirical models
7. Brown, Gans, Haipeng, Zhao: Statistics ⇒ Queueing Science
8. Atar, Reiman, Shaikhet: Skills-based routing ⇒ Control models
9. Nakibly, Meilijson, Pollatchek: Prediction of waiting ⇒ Online Models and Real-Time Simulation
10. Garnett: Practice ⇒ 4CallCenters.com
11. Zeltyn: Queueing Science ⇒ Empirically-Based Theory
12. Borst, Reiman; Zeltyn: Dimensioning M/M/N+G
13. Kaspi, Ramanan: Measure-Valued models and approximations
14. Jennings; Feldman, Massey, Whitt: Time-stable performance (ISA)

"Customers" or "Products": Is the process of teaching Service Engineering a production process (contributing to the production of Service Engineers) or a service process (serving students)? This is a question worthwhile discussing in a first lecture of a Service Engineering course, but we shall not pursue it here. We shall only say that, as either "Service-Providers" or "Producers", in short Teachers, we are grateful to the many students of Service Engineering, who have learned and labored through the course material and homework: undergradu-

ates, graduates, postdocs, researchers, practitioners and colleagues. The Technion, and the above mentioned hosting institutions for the mini-courses, are blessed with students of high quality and drive, which makes the teaching experience (more often than not), and hopefully the learning experience as well, challenging, enjoyable and rewarding. Indeed, learning that a course-graduate keeps the lecture notes handy on an office shelf, or having the course be the trigger for a student's successful academic career, is all that a teacher can hope for.

References

[1] Adler P.S., Mandelbaum A., Nguyen V. and Schwerer E. (2005) From project to process management: An empirically-based framework for analyzing product development time. *Management Science*, 41, 458-484. 4.3.1

[2] Aksin, Z., Armony, M., Mehrotra, V. (2007) The Modern Call-Center: A Multi-Disciplinary Perspective on Operations Management Research. *Production and Operations Management*, Special Issue on Service Operations in honor of John Buzacott (ed. G. Shanthikumar and D. Yao) 16(6), 655-688. 4.5.3

[3] Aldor-Noiman S., Feigin P.D. and Mandelbaum A. (2009) Workload Forecasting for a Call Center: Methodology and a Case Study. To be published in the *Annals of Applied Statistics*. 6

[4] Asmussen, S. (2003) Applied Probability and Queues, 2nd Edition, Springer. 4.6.3

[5] Atar R., Mandelbaum A. and Reiman M. (2004) Scheduling a Multi-Class Queue with Many iid Servers: Asymptotic Optimality in Heavy-Traffic. *The Annals of Applied Probability*, 14(3), 1084-1134. 6

[6] Atar R., Mandelbaum A. and Shaikhet G. (2006) Queueing systems with many servers: null controllability in heavy traffic. *The Annals of Applied Probability*, 16(4), 1764-1804. 6

[7] Barron Y. (1996) Performance Analysis of Dynamic Stochastic PERT/CPM Networks. M.Sc. Thesis, Technion. Available at
http://ie.technion.ac.il/serveng/References/thesis_Yonit_Heb.pdf,
in Hebrew, and
http://ie.technion.ac.il/serveng/References/thesis_Yonit_Eng.pdf,
English Summary. 4.3.1

[8] Bertsimas D. and Mourtzinou G. (1997) Transient laws of non-stationary queueing systems and their applications, *Queueing Systems: Theory and Applications (QUESTA)*, 25(5), 115-155. 4.2.5

[9] Borst S., Mandelbaum A., and Reiman M. (2004), Dimensioning large call centers, *Operations Research*, 52(1), 17-34. 1.4.2, 4.7.1, 6

[10] Brown L.D., Gans N., Mandelbaum A., Sakov A., Shen H., Zeltyn S. and Zhao L. (2005) Statistical analysis of a telephone call center: a queueing science perspective. *Journal of the American Statistical Association (JASA)*, 100(469), 36-50. 3, 4, 4.5.1, 4.5.2, 4.5.2, 4.5.3, 29, 4.7.2, 6

[11] Chen, H. and Yao D. (2001) Fundamentals of Queueing Networks: Performance, Asymptotics, and Optimization. New York, NY: Springer-Verlag. 4.6.3

[12] Cohen, I. (2003) Multi-Project Scheduling, Technion Ph.D. thesis, a summary of which is available at
http://ie.technion.ac.il/serveng/Lectures/MultiProject.zip 4.3.1

[13] Corbett, C.J., van Wassehhove, L.N. (1993) The natural drift: what happened to Operations Research?, *Operations Research*, 41, 625-640. 1.5.4

[14] Cox D.R. and Oakes D. (1984) Analysis of Survival Data, Chapman and Hall. 4.5.3

[15] CV of A.M., available at
http://ie.technion.ac.il/serveng/References/AviM_cv_08.pdf. 7

[16] Dai J.G. A fluid limit model criterion for instability of multiclass queueing networks, *Annals of Applied Probability*, 6, 751-757. 4.4.2

[17] Eick S.G., Massey W.A. and Whitt W. (1993) The physics of the Mt/G/1 queue, *Operations Research*, 41(4), 731-742. 4.2.5, 4.2.6

[18] Erlang A.K. (1948) On the rational determination of the number of circuits. In *The life and works of A.K.Erlang.* Brockmeyer E., Halstrom H.L. and Jensen A., eds. Copenhagen: The Copenhagen Telephone Company. 1.4.1

[19] Example of a ServEng exam. Available at
http://ie.technion.ac.il/serveng/Lectures/Exams/Moed_A_2004_Eng.pdf. 5.2

[20] Federgruen A., and Samojednik A. (2008) VONAGE: Designing a Service Strategy. Columbia University case. Available at
http://www.docstoc.com/docs/14384169/VONAGE. 4.1

[21] Feldman Z. (2008) Optimal Staffing of Systems with Skills-Based-Routing. Technion M.Sc. thesis. Available at http://ie.technion.ac.il/serveng/References/Zohar_Thesis.pdf. 1.3.2

[22] Feldman Z., Mandelbaum A., Massey W. and Whitt W. (2008) Staffing of time-varying queues to achieve time-stable performance. *Management Science*, 54(2), 324-338. 4.7.3, 4.7.3, 6

[23] Fitzsimmons J. and Fitzsimmons M. (2004) Service Management: Operations, Strategy, Information Technology McGraw Hill, 4th Edition. 1.1, 4.1, 5.1.1

[24] Flanders S. (1980) Modeling court delay. *Law & Policy Quarterly*, 2, 305-320. 4.2.3, 4.2.4

[25] Fluid systems in practice. Movies on transportation systems. Available at http://ie.technion.ac.il/serveng/Lectures/MITSIM.zip. 4.4.1

[26] 4CallCenters Software (2002). Available at http://ie.technion.ac.il/serveng/4CallCenters/Downloads.htm. 4.6, 4.6.2, 5.1.1

[27] Fralix B., Riano G. and Serfozo, R. (2007) Time-Dependent Palm Probabilities and Queueing Applications. Technical Report. Available at http://alexandria.tue.nl/repository/books/631529.pdf 4.2.5

[28] Frei F. X., Harker P. T., Hunter L. W. (1998) Innovation in Retail Banking. Report, Wharton's Financial Institutions Center. Available at http://ie.technion.ac.il/serveng/Lectures/Retail.pdf, or http://fic.wharton.upenn.edu/fic/papers/97/9748.pdf. 1.1, 4.3

[29] Gans N., Koole G. and Mandelbaum A. (2003) Telephone call centers: a tutorial and literature review. Invited review paper, *Manufacturing and Service Operations Management*, 5(2), 79-141. 4.5.3

[30] Garnett O., and Mandelbaum A. (2000) An Introduction to Skills-Based Routing and its Operational Complexities. Teaching note, Technion, Israel. Available at http://ie.technion.ac.il/serveng/Lectures/SBR.pdf. 4.8

[31] Garnett O., Mandelbaum A. and Reiman M. (2002) Designing a telephone call-center with impatient customers. *Manufacturing and Service Operations Management*, 4, 208-227. 1.4.2, 4.5.3, 4.6.2, 4.7.1, 4.7.2, 6

[32] Green L.V., Kolesar P.J. and Whitt W. (2007) Coping with time-varying demand when setting staffing requirements for a service system. *Production and Operations Management*, 16(1), 13-39. 4.2.6

[33] Gurvich I., Armony M. and Mandelbaum A. (2008) Service Level Differentiation in Call Centers with Fully Flexible Servers. *Management Science*, 54(2), 279-294. 6

[34] Halfin S. and Whitt W. (1981) Heavy-traffic limits for queues with many exponential servers. *Operations Research*, 29, 567-588. 1.4.2, 4.7.1, 4.7.2, 6

[35] Hall R.W. (1991) Queueing Methods for Services and Manufacturing, Englewood Cliffs, New Jersey, USA: Prentice Hall. 1.4.1, 4.1, 4.4.2, 4.6.3

[36] R.W. Hall (Editor) (2007) Patient Flow: Reducing Delay in Healthcare Delivery (International Series in Operations Research and Management Science). 1.4.1

[37] Harrison M.J., Loch C., and Grant D.P. (1993) Manzana Insurance: Fruitvale Branch. Stanford University Graduate School of Business School Case S-DS-87. 4.1

[38] (2005) Hazard Rate Functions: Examples via Phase-Type Distributions. Teaching Note, Technion IE&M. Available at http://ie.technion.ac.il/serveng/Recitations/hazard_phase_type.pdf 4.5.2

[39] Helber, S. and Stolletz,, R. (2004) Call Center Management In Der Praxis: Strukturen Und Prozesse Betriebswirtschaftlich Optimieren. Springer. 4.6

[40] Hershey J., Pierskalla W. and Wandel S. (1981) Nurse staffing management. In: D. Doldy (ed.), *Operational Research Applied to Health Services*, Croon Helm, London, 189-220. 4.7.4

[41] Herman, R. (1992) Technology, human interaction, and complexity: reflection on vehicular traffic science. *Operations Research*, 40(2), 199-212. 1.5.4

[42] Hopp W. and Spearman M. (2000) Factory Physics. Second Edition, McGraw-Hill. 4.5.3

[43] Janssen A.J.E.M., van Leeuwaarden J.S.H. and Zwart B. (2008) Refining square root safety staffing by expanding Erlang C, to appear in *Operations Research*. 1.4.2, 4.7.2

[44] Jennings O., Mandelbaum A, Massey W. and Whitt W. (1996) Server Staffing to Meet Time-Varying Demand. *Management Science*, 42(10), 1383-1394. 6

[45] Jennings O. and de Vericourt F. (2007) Nurse-to-patient ratios in hospital staffing: a queueing perspective. Working Paper, Duke University. 1.4.2

[46] Khudyakov P. (2006) Designing a Call Center with an IVR (Interactive Voice Response). M.Sc. Thesis, Technion. Available at http://ie.technion.ac.il/serveng/References/thesis_polyna.pdf. 1.4.2

[47] Khudyakov P., Feigin P.D., and Mandelbaum A. (2009) Designing a Call Center with an IVR (Interactive Voice Response). Submitted for publication. 6

[48] Kopach-Konrad R., Lawley M., Criswell M., Hasan I., Chakraborty S., Pekny J., and Doebbeling B.N. (2007) Applying Systems Engineering Principles in Improving Health Care Delivery. *Journal of General Internal Medicine*, 22 (Suppl 3), 431-437. 4.2.3

[49] Larson R.C., Cahn M.F., and Shell M.C. (1993) Improving the New York City Arrest-to-Arraignment System. *Interfaces*, 23(1), 76-96. 4.3.1, 7

[50] Liberman P., Trofimov V. and Mandelbaum A. (2008) Skills-Based-Routing in US-Bank. Technion, Israeli Institute of Technology, Technical Report. Available at http://ie.technion.ac.il/Labs/Serveng. 4.8

[51] Little J.D.C. (1961) A Proof of the Queueing Formula $L = \lambda W$, *Operations Research*, 9, 383-387. 4.2.2

[52] Litvak E., McManus M.L., and Cooper A. (2002) Root cause analysis of emergency department crowding and ambulance diversion in Massachusetts, Boston University Program for Management Variablity in Health Care Delivery. 4.2.6, 4.5.2

[53] Liu Y. and Whitt W., (2009) Stabilizing Customer Abandonment in Many-Server Queues with Time-Varying Arrivals. Submitted to Operations Research. Available at http://www.columbia.edu/~ww2040/LiuWhittStabilizing0606.pdf 4.7.3

[54] Lovelock. C.G. (1992) Managing Services: Marketing, Operations and Human Resources, Prentice-Hall. 4.1

[55] Maman S. (2009) Uncertainty in the Demand for Service: The Case of Call Centers and Emergency Departments. M.Sc. Thesis, Technion, April 2009. 6

[56] Mandelbaum A. and Massey W.A. (1995) Strong approximations for time-dependent queues. *Mathematics of Operations Research*, 20, 33-64. 4.4.2, 6

[57] Mandelbaum A., Massey W.A. and Reiman M. (1998) Strong Approximations for Markovian Service Networks. *Queueing Systems: Theory and Applications (QUESTA)*, 30, 149-201. 1.4.2, 4.4.2

[58] Mandelbaum A., Massey W.A., Reiman M. and Rider B. (1999) Time Varying Multiserver Queues with Abandonment and Retrials. ITC-16, Teletraffic Engineering in a Competitive World, Editors P.Key and D.Smith, Elsevier, 355-364. 4.4.2

[59] Mandelbaum A., Massey W.A., Reiman M., Rider B. and Stolyar A. (2000) Queue Lengths and Waiting Times for Multiserver Queues with Abandonment and Retrials. *Selected Proceedings of the Fifth INFORMS Telecommunications Conference.* 4.4.2, 6

[60] Mandelbaum, A. and Reiman, M. (1998) On Pooling in Queueing Networks. *Management Science*, 44, 971-981. 4.5.2

[61] Mandelbaum A., Sakov A., and Zeltyn S. (2000) Empirical Analysis of a Call Center. Technical report, Technion. Available at http://ie.technion.ac.il/serveng/References/ccdata.pdf. 3, 6

[62] Mandelbaum A. and Shimkin N. (2000) A model for rational abandonment from invisible queues. *Queueing Systems: Theory and Applications (QUESTA)*, 36, 141-173. 6

[63] Mandelbaum A. and Stolyar A. (2004) Scheduling Flexible Servers with Convex Delay Costs: Heavy-Traffic Optimality of the Generalized $c\mu$-Rule. *Operations Research*, 52(6), 836-855. 4.8, 6

[64] Mandelbaum A. and Zeltyn S. (1998) Estimating characteristics of queueing networks using transactional data. *Queueing Systems: Theory and Applications (QUESTA)*, 29, 75-127. 3

[65] Mandelbaum A. and Zeltyn S. (2005) The Palm/Erlang-A Queue, with Applications to Call Centers. Teaching note to *Service Engineering* course. Available at http://ie.technion.ac.il/serveng/References/Erlang_A.pdf. 4.5, 4.5.3, 4.6.1, 4.7.2

[66] Mandelbaum A. and Zeltyn S. (2007) Service engineering in action: the Palm/Erlang-A queue, with applications to call centers. In: Spath D., Fähnrich, K.-P. (Eds.), *Advances in Services Innovations*, 17-48, Springer-Verlag. 4, 4.5.3, 4.5.3, 4.6.1

[67] Mandelbaum A. and Zeltyn S. (2009) Staffing many-server queues with impatient customers: constraint satisfaction in call centers. To be published in *Operations Research*. 4.5.3, 6

[68] Mandelbaum A. and Zeltyn S. (2009) Service Engineering: Data-Based Course Development and Teaching. An abbrerivated version of the present document. *

[69] Marmor Y. , Mandelbaum A., Vasserkrug S., Carmeli B., Greenshpan O., Vortman P., Schwartz D. and Basis F. (2009) InEDvance: Advanced IT in Support of Emergency Department Management. *In proceedings of 2009 NGITS conference.* 6

[70] Marmor Y. , Shtub A., Mandelbaum A., Vasserkrug S., Zeltyn S., Carmeli B., Greenshpan O. and Masika Y. Toward Simulation Based Real-Time Decision Support Systems For Emergency Departments. Accepted to *2009 Winter Simulation Conference.* 6

[71] Munichor N. and Rafaeli A. (2007) Number or apologies? Customer reactions to telephone waiting time fillers. *Journal of Applied Psychology*, 92(2), 511-518. 4.5.3

[72] Newell G.F. (1980) Traffic flows on transportation networks, MIT Press. 4.4.1, 4.4.2

[73] Newell G.F. (1982) Application of Queueing Theory, Chapman and Hall. 1.4.1, 4.4.1, 4.4.2

[74] Oliver R.M. and Samuel A.H. (1962) Reducing letter delays in post offices. *Operations Research*, 10, 839-892. 4.4.2

[75] Palm C. (1953) Methods of judging the annoyance caused by congestion. *Tele*, 4, 189-208. 4.5.3

[76] Palm C. (1957) Research on telephone traffic carried by full availability groups. Tele, vol.1, 107 pp. (English translation of results first published in 1946 in Swedish in the same journal, which was then entitled *Tekniska Meddelanden fran Kungl. Telegrafstyrelsen.*) 1.4.1, 4.5.3, 6

[77] Pierskalla W.P. and Brailer D.J. (1994) Applications of Operations Research in Health Care Delivery. In: S.M.Pollock et al., Eds., *Handbooks in OR & MS, Vol.6.* 4.7.4

[78] Paxson V. and Floyd S. (1995) Wide area traffic: the failure of Poisson modeling. *IEEE/ACM Trans. Netw.*, 3(3), 226-244. 4.5.1

[79] Porteus E.L. (1989) The Case Analysis Section: National Cranberry Cooperative. *Interfaces*, 19, 29-39. 4.1, 4.4.2

[80] Porteus E.L. (1993) Case Analysis: Analyses of the National Cranberry Cooperative: 1. Tactical Options. *Interfaces*, 23, 21-39. 4.1, 4.4.2

[81] Porteus E.L. (1993) Case Analysis: Analyses of the National Cranberry Cooperative: 2. Environmental Changes and Implementation. *Interfaces*, 23, 81-92. 4.1, 4.4.2

[82] Rozenshmidt L. (2007) On Priority Queues with Impatient Customers: Stationary and Time-Varying Analysis. M.Sc. Thesis, Technion. Available at http://ie.technion.ac.il/serveng/References/thesis_Luba_Eng.pdf 4.7.3

[83] "Service Engineering" course website, Technion, http://ie.technion.ac.il/serveng. (document), 1.1, 2, 4.1, 6

[84] SEE: Website of the Technion's "Service Enterprise Engineering" Research Center, http://ie.technion.ac.il/Labs/Serveng/. (document), 3.1, 7

[85] Shimkin N. and Mandelbaum A. (2004) Rational Abandonments from Tele-Queues: Nonlinear Waiting Costs with Heterogeneous Preferences. *Queueing Systems: Theory and Applications (QUESTA)*, 47, 117-146. 6

[86] Simon H.A. (1996) The Sciences of the Artificial. The MIT Press; third edition. 1.5.1

[87] Trofimov V., P.D. Feigin, Mandelbaum A., Ishay E., and Nadjharov E. (2006) DATA MOdel for Call Center Analysis: Model Description and Introduction to User Interface. Technion, Israeli Institute of Technology, Technical Report. Available at http://ie.technion.ac.il/Labs/Serveng. 3.1, 5.1.1

[88] Tseytlin Y. (2009) Queueing Systems with Heterogeneous Servers: On Fair Routing of Patients in Emergency Departments. Technion M.Sc. Thesis, April 2009. Available at http://ie.technion.ac.il/serveng/References/thesis-yulia.pdf. 6, 7

[89] U.S. Bank SEELab Report (2006) DataMOCCA: DATA MOdel for Call Center Analysis - The Call Center of "US Bank". Prepared by Olga Donin, with the support of SEELab researchers. Available at http://ie.technion.ac.il/Labs/Serveng/files/The_Call_Center_of_US_Bank.pdf. 3

[90] Weinberg J., Brown L.D. and Stroud J.R. (2007) Bayesian forecasting of an inhomogeneous Poisson process with applications to call center data. *J. Amer. Statist. Assoc.*, 102, 1185-1199. 4.5.1, 6

[91] Whitt W. (2005) Engineering Solution of a Basic Call-Center Model. *Management Science*, 51(2), 221-235. 4.6.3

[92] Whitt W. Research site. http://www.columbia.edu/~ww2040/allpapers.html. 1.4.2, 1.4.3

[93] R.W. Wolff. (1982) Poisson arrivals see times averages. *Operations Research*, 30, 223-231. 4.5.1

[94] H. Xie, T. J. Chaussalet and P. H. Millard. (2203) A continuous time Markov model for the length of stay of elderly people in institutional long-term care. *Journal of the Royal Statistical Society Series A*, 168(1), 51-61. 4.5.2

[95] Yom-Tov G. (2007) Queues in Hospitals: Semi-Open Queueing Networks in the QED Regime. Ph.D. Research Proposal, Technion. Available at http://ie.technion.ac.il/serveng/References/proposal_Galit.pdf. 1.4.2

[96] Zeltyn S. and Mandelbaum A. (2005) Call centers with impatient customers: many-server asymptotics of the M/M/n+G queue. *Queueing Systems: Theory and Applications (QUESTA)*, 51, 361-402. 1.4.2, 4.6.2, 4.7.1, 4.7.2, 6

[97] Zohar E., Mandelbaum A. and Shimkin N. (2002) Adaptive behavior of impatient customers in tele-queues: theory and empirical support. *Management Science*, 48, 566-583. 6

[98] Zviran A. (2008) Fork-Join Networks in Heavy Traffic: Diffusion Approximations and Control. M.Sc. Research Proposal, Technion. Available at http://ie.technion.ac.il/serveng/References/Asaf_research_proposal.pdf. 1.4.3, 4.3.1