

Call Centers with Skills-Based-Routing: Simulation-Based State-Inference and Dynamic-Staffing

M.Sc. Research Proposal
Arik Senderovich

Advisor: Prof. Avishai Mandelbaum

The Faculty Of Industrial Engineering and Management
Area of Statistics
Technion - Israel Institute of Technology
February 2010

Contents

1	Introduction	2
2	Staffing Call Centers	4
2.1	The Workload and the Offered Load	4
2.2	Staffing Regimes	5
2.3	”Dynamic Staffing”	6
3	Skills-Based Call Centers: Overview and Modeling via Simulation	8
3.1	Introduction to SBR	9
3.2	Simulating Skills-Based Call Center	11
4	Databases for Our Simulations	15
4.1	The SEE Center and the DataMOCCA Project	15
5	Our Proposed Research	16
	References	18

1 Introduction

Many companies (in both the private and public sectors) face large daily demand for multi-type customer-services. These services can vary from complicated monetary transactions to product-sales, through dealing with unsatisfied customers that express their discontent (Customer Relations). Therefore, an efficient way of communication between the customers and the organization is called for. Here Call Centers come for the rescue, by providing an easily accessible communication channel for the customer to reach a company and, at the same time, enabling the company to respond with fast, appropriate and relatively cheap services. Many companies use, in fact, one or more Call Centers, as almost the only way to interact with their customers. (70% of customer-firm interactions occur through Call Centers [12]).

Managers nowadays are aware of the basic rule in business survival - Customer Satisfaction. This is why a great deal of resources and administrative attention is directed towards Call Centers' operation. In particular, performance measures are periodically assessed (online and off line), as a way to estimate customers' satisfaction levels. Common measures used are - Fraction of calls answered within target time, fraction of calls abandoning etc.

Call Center services can be provided either by a VRU (Voice Response Unit) or a human agent. The agents are essential, due to the fact that many calls can not be handled by standard VRU protocols. We shall, from now on, refer to the human agents as agents, servers, or service providers interchangeably.

Together with their great importance to the company's Customer Relations, service providers also bring along a substantial cost in terms of salaries paid for their staffing (around 70% of operating expenses of a Call Center [12]). Hence, the importance of activating the *right number of agents*, at the *right time*. The latter two terms were emphasized, in order to show that they are specific to each Call Center. Some managers would pay a steep price for Quality of Service (highly satisfied customers), while others would prefer to be more efficient in the utilization of their human resources (higher agent utilization requires less service providers, therefore reducing operating costs).

The timing of the arrival to and departure from the working place, of the service providers, is important. The ideal solution would be: for each time-interval of the day, staff the *exact* number of agents that is required at that time period, in order to meet the demand for service. Yet this is typically impossible due to real-life constraints. An example for such a constraint would be a minimal shift-duration for which the agents can be brought in. (Agents often can not be called over for only a fraction of the shift).

Fitting the *appropriate* number of agents, under some restrictions, will both provide the *service* the customers seek as well as reduce *operating costs*. Many mathematical models have been developed to help determine the "best" staffing levels in Call Centers. Most of them use severe assumptions on the Arrival Process (When do the calls arrive? How many calls arrive? What types of calls arrive?), and the Service Time (How long does it take to serve the calls? Is that time correlated with the call type?). Some models also assume that the Call Center is large enough [2] (Economy of Scale). The output of the models are often guidelines (rules of thumb) that can help in setting staffing levels. The question of appropriate staffing is in the center of the Service Engineering field, and will also be the main theme of our research.

Specifically, staffing is challenging in Call Centers with a large variety of customer-types, that are handled by groups of skilled agents (as introduced further on in Section 3). Some mathematical models for such Call Centers have been developed, in the past few years [6][1]. It is not always possible to accommodate the assumptions needed for such models, specifically regarding the arrival and service processes (see Section 2) in Call Centers with such heterogeneous customer types and server pools. Simplification of the problem is possible, which renders existing models of Service Engineering applicable. The cost of the simplification, for example, is that it neglects the correlation between the multiple types of incoming calls ([9]), and disregard low probability inter-pool paths (uncommon call-transactions). One of the main goals of the current research is to treat the Call Center as a complete entity, covering all its processes, and being able to estimate its current status, online. We shall also seek to determine proper staffing levels, for all agent groups at all times.

The use of *Simulation* for modeling such Call Centers is natural, in light of the diversity of customer types, and the complexity of the routing of these customers to appropriately skilled agents. In fact, an event-based simulator (see Section 3) was created for the purpose of modeling skills-based Call Centers. The integration of databases containing real Call-by-Call data (see Section 4) into the mentioned simulator will be the core of our research method (Section 5).

The research proposal is structured as follows. In Section 2, we review some aspects of Staffing in Call Centers (Offered Load, Staffing Regimes, and Dynamic Staffing). In Section 3, we elaborate on Skills-Based-Routing (SBR) in Call Centers, including a discussion on the SBR simulator. Section 4 is dedicated to our supporting Call Center database, which will feed the Simulation runs and will be used as a validation environment. Finally, Section 5 concludes with our main research topics.

2 Staffing Call Centers

The importance of appropriate staffing in a Call Center has caught the spotlight in the area of Service Engineering. Extensive research of the subject is still ongoing. The issue is also in the center of every (Call Center's) manager's agenda (due to reasons mentioned in Section 1). Several models (mostly stochastic) have been developed to help determine time-stable performance levels, in the time-varying environment of the Call Centers. We will now define the Workload process and the Offered Load function, which are the backbone of such staffing algorithms. Subsequently we discuss various static (off line) staffing regimes. We shall conclude with a short introduction to dynamic (online) staffing in Call Centers.

2.1 The Workload and the Offered Load

Two basic processes occur simultaneously in a Call Center: The **Arrivals** of customers to the Call Center, and the **Service process** of these customers. These processes procreate the following two definitions:[15]

1. **Resource-k Workload:** A stochastic process over time $t \geq 0$, representing the number of customers being served by a resource of type k at time t, under the assumption that there are infinitely many resources of type k available for service and, hence, a task that requires resource k starts service immediately.
2. **Resource-k Offered Load:** A function of time $t \geq 0$, representing the average of Resource-k Workload at time t.

As the simplest example of the (stationary) Offered Load, one could consider a constant number of arrivals λ per hour, and a Service Time of random duration - S (measured in hours). The Offered Load is then simply

$$R = \lambda \cdot E(S).$$

Having the Offered Load (R) estimated, relatively simple models that can be used for staffing are: The $M/M/N$ model (Erlang-C)[2] and the $M/M/N + M$ model (Erlang-A)[7]. These models assume that there are N servers, which serve customers that arrive accordingly to a (homogenous) Poisson process (with rate λ). The service times are assumed to have exponential distribution (with mean $1/\mu$). In Erlang-C, no abandonments are allowed, whereas in Erlang-A, abandonments are accommodated, each after an exponential period (with mean $1/\theta$) of (im)patience of the corresponding customer.

These models have two major flaws:

1. The models impose restrictive assumptions on its building blocks (Exponentially distributed service times, inter-arrival times and (im)patience). In large Call Centers these models perform reasonably well, and the approximations are rather useful [3].
2. The models disregard time-varying environments common in Call Centers. More complex models allow R that is time-varying ($R(t)$), and under certain assumptions, $R(t)$ can be estimated [15]. This is of course the more realistic scenario, and in our research we will aim at covering time-varying environment.

2.2 Staffing Regimes

Once we have the Offered Load (R) estimated, the Call-Center's manager is to decide on a strategy: Quality of Service (QoS) vs. Efficiency of the human resource. On one hand, every manager would prefer customers at their maximum satisfaction levels, therefore choosing the QoS approach (90% of U.S. consumers form a company's image via their Call Center experience). On the other hand, given natural financial constraints, the cost of low utilization (high staffing levels) is high.

We now present three common heuristic approaches (rules of thumb), which essentially transform the estimated Offered Load into recommended staffing, per time period. These approaches turn out to provide time-stable results, especially for moderate-to-large Call Centers. For simplicity, let us assume that the Offered Load is constant (Time-homogeneous poisson arrivals)[2]:

1. **The Efficiency-Driven (ED) regime** - Given an Erlang-A model, as an example, ED staffing N would be:

$$N = R - \gamma \cdot R, \quad 0.1 \leq \gamma \leq 0.25.$$

This essentially means that we are to allocate fewer servers than R . The ED regime focuses on, agent utilization, at the cost of reduction in QoS. The larger the γ the higher is the utilization of agents, but the trade-off is a higher probability to abandon. Indeed, the abandonment probability is approximately γ .

Under the ED regime, customers will almost surely have to wait for service, abandonment risk are not low (10-20%), but the Call Center will run more economically. This method of staffing is relevant for organizations who face little to no competition and/or companies with a very attractive product.

2. **The Quality-Driven (QD) regime** - For managers who will not settle for low QoS and who can not afford losing any customer to competition, the ED regime would be disastrous. Quality driven managers are to operate their Call Center according to the following rule of thumb (for both the Erlang-A and Erlang-C models):

$$N = R + \gamma \cdot R, \quad 0.1 \leq \gamma \leq 0.25.$$

This regime aims at high probability of starting service without any prior delay, hence negligible abandonment rate and very high customer satisfaction levels. The problem with this method is that slacking might occur, especially in realities that are time-varying but assume constant arrival rate: during "slow" hours, agents will be idle, and idleness costs will not be covered via customers satisfaction.

3. **The Quality- and Efficiency-Driven (QED) regime** - This regime attempts to rationalize between the first two approaches. In most Call Centers, managers strive to simultaneously achieve low operating cost, jointly with high customer satisfaction. This is achievable in large-enough Call Centers, via the following staffing rule (Erlang-A):

$$N = R + \beta \cdot \sqrt{R}, \quad -1 \leq \beta \leq 1.$$

In practice, under the QED regime, 25-75% (β dependant) of the customers are delayed prior to service, abandonment rate is between 1% and 5%; and the average wait time (for service) is an order of magnitude smaller than the service time (seconds vs. minutes). For more accurate QED applications, β could be derived from a desired probability to wait target.

In our research, we shall typically assume that the QED approach would be appropriate. Especially, due to the fact that we will base our research on the database of a large organization (high operating cost - efficiency needed), with a large variety of competing companies (QoS is called for).

The staffing models and regimes, mentioned so far, were all static in the sense of being pre-determined. For example, while planning daily staffing for a Tuesday, managers make decisions on a previous Monday or earlier. One of our research themes will be using simulated Offered Load for online real-time *dynamic staffing*: for example, deciding during Tuesday morning on changes of staffing schemes for Tuesday afternoon.

2.3 "Dynamic Staffing"

"Dynamic Staffing" means adjusting the number of agents available to provide service, during the present shift (online) according to the Call Center's current state. Since we expect that there will be some deviations from a

priori estimated Offered Load, it would be natural to modify our staffing accordingly. The "Dynamic Staffing" reduces uncertainty, allows accurate shift-planning and optimizes both Efficiency and QoS. We shall divide our "Dynamic Staffing" into two parts: short term (minutes) and medium term (hours, shifts).

The first part is thoroughly discussed in [18]. The article presents an algorithm that determines short term staffing, online. According to the author, the key to such dynamic staffing is exploiting knowledge of the current state of the Call Center, along with having accurate estimates (at least mean and variance) of service demand, for the near future. The author also provides a recipe for immediately answering all calls. He does so by using the $M/G/\infty$ model, which assumes an infinite number of agents. The available number of agents is assumed to be fully flexible (activated and deactivated on demand). The model specifies how many agents are to be activated, in the short term, under these assumptions.

Another attempt to produce dynamic staffing was made in [14], here in the context of an Emergency Department (ED). In this article, the authors describe how using a suitable simulator, that correctly captures the complex environment of an ED, one can extract (using relatively few assumptions on the processes) appropriate staffing levels. The article discusses a method similar to Whitt's in [18], except for not presuming to have infinite number of human agents. This, of course, implies that there are customers who are not immediately served. The presented method for estimating demand and service in [14] is based on feeding the simulator with real-time data. Some of it will be accurate (arrivals of patients), and some will be inaccurate (departures, or future patient transactions). One of the proposed ways of actual staffing for the ED is using the Offered Load $R=R(t), t \geq 0$. The QED regime is applied, along with time-varying approximations for $R(t), t \geq 0$. In our research, a similar approach will be taken (assuming a time-varying R , and the QED environment). Our simulator will be presented in the Section 3.2 below.

3 Skills-Based Call Centers: Overview and Modeling via Simulation

Consider the following scheme, which represents a small part of a Call Center's operational reality:

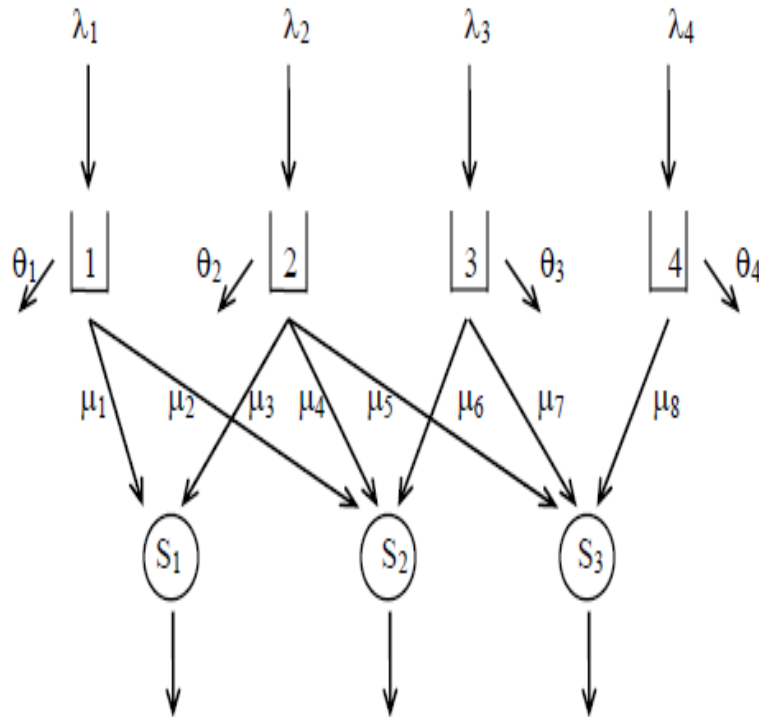


Figure 1: "Multi-queue multi-skill system = schematic depiction of a telephone call-center"

Here the λ 's designate the arrival rates for various customer types, the μ 's service rates, the θ 's individual abandonment rates and the S 's are the number of agents in each agent pool (group of agents, sharing the same skill).

Note that analytical representation of even this limited diagram is very complicated. Advanced tools were developed in order to help managers with handling such complex scenarios. One of the tools is Skills-Based-Routing (SBR) protocols [6].

In the next two subsections, we focus on SBR, continuing with simulating Skills-Based environments in Call Centers.

3.1 Introduction to SBR

SBR is a protocol for online routing of the customers that arrive to some service-providing system [11]. In a Call Center, this means providing the caller with the most appropriate service (quality-and speed-wise). Doing so is possible by routing the call to a group of appropriately skilled agents. The routing protocol can be either Static (pre-determined to address a regular scenario) or Dynamic (also pre-determined, but adoptable, in response to non-regular scenarios). Static protocols are dependant upon the natural assignment of calls to agents. For example, we plan the routing protocol so that it directs the Arabic speaking customers to an Arabic speaking agents. Dynamic protocols are created to deal with online "problems" that might occur during the day. It is obvious that good planning requires both protocols to work hand in hand. The co-existence of these protocols requires the planners of the Call Center to create routing schemes and algorithms that are completely automated and flexible (easily altered over time, in response to changes of a Call Center's state).

When implementing a skills-based system, the major decisions to be made are:

1. Who are the customers - defining customer types (*off line*).
2. Who are the servers - their skills and numbers (*off line*).
3. How are customers routed to servers - the control policy (*static- or dynamic - online*).

These decisions typically require the involvement of separate divisions in the Organization: Marketing (for no. 1), Human Resources (for 2) and Operations (for no. 2 and 3), all supported by IT (To make the routing infrastructure automated). All these factors underscore the complexity of designing and maintaining SBR protocols.

A Call Centers' SBR *design* consists of classifying the customers and determining the servers' required skills. A particular single design can have alternative interpretations, for example:

1. Different customer-types can represent customers requiring different services (e.g. technical support vs. billing) or customer priorities (VIP vs. Members).
2. Separate server-pools can be due to servers' level of Capabilities-Training-Experience (e.g. Hebrew-English speaker vs. Arabic-Russian-Spanish speaker, generalist vs. specialist, expert vs. novice).

Even with simple designs there can be associated many natural different control (routing) policies. The two decisions to be made when routing customers are:

1. Whenever a service ends and there are queued customers, which customer (if any) should be routed to the server just freed. This is also called: "Pull decision policy"[11].

2. Whenever a customer arrives and there are idle servers, to which one of them (if any) should the customer be routed. This is also called: "Push decision policy"[11].

Here is a chart that demonstrates a flow scheme in a large Israeli Telecom Call Center (during all of March of 2008)[10]:

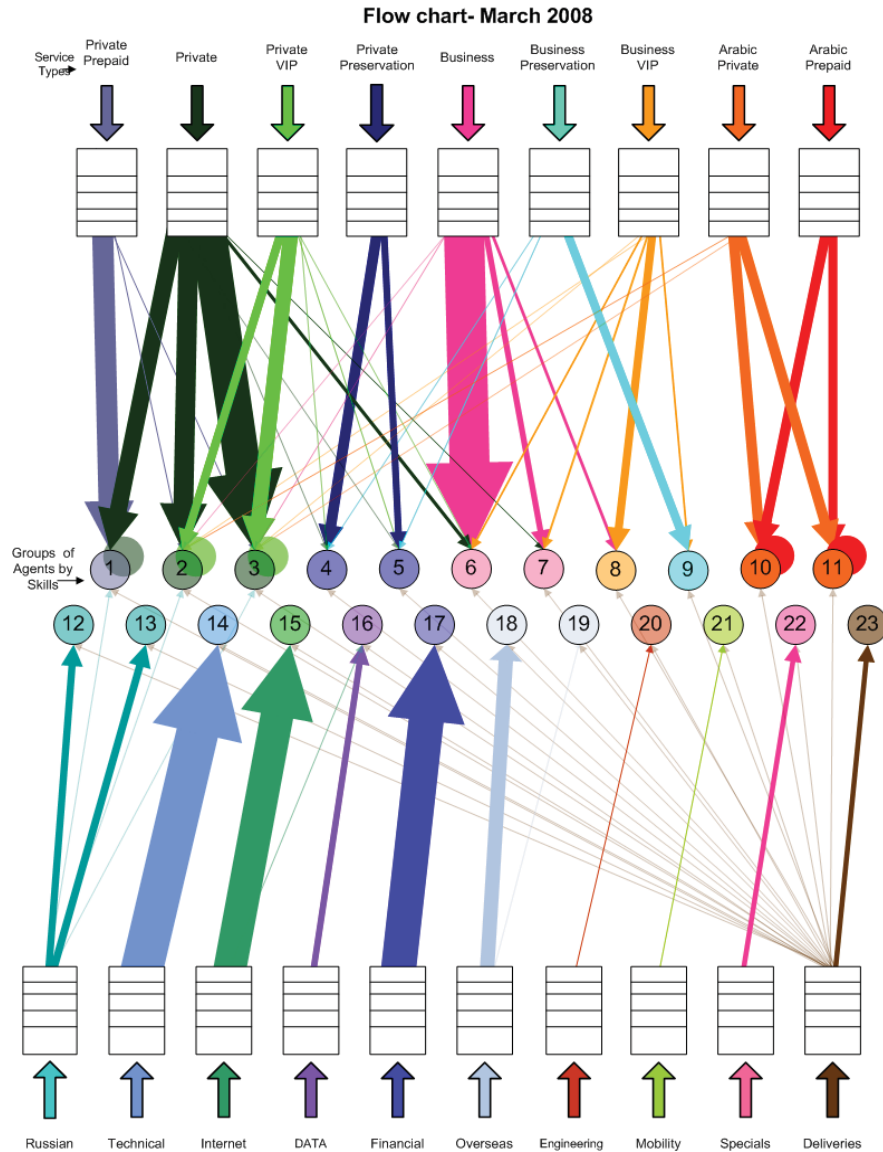


Figure 2: "Skills-Based-Routing scheme - Large Israeli Tele-company"

In Figure 2, we see arrivals (types of customers) entering queues. Then these calls are routed (designated with wide or narrow arrows, corresponding to the rate of flow: high rate with wide arrow) to the pools of servers (that historically served these customers during March 2008).

Figure 2 is a representative example of how complex an SBR algorithm can become in large Call Centers. It gives

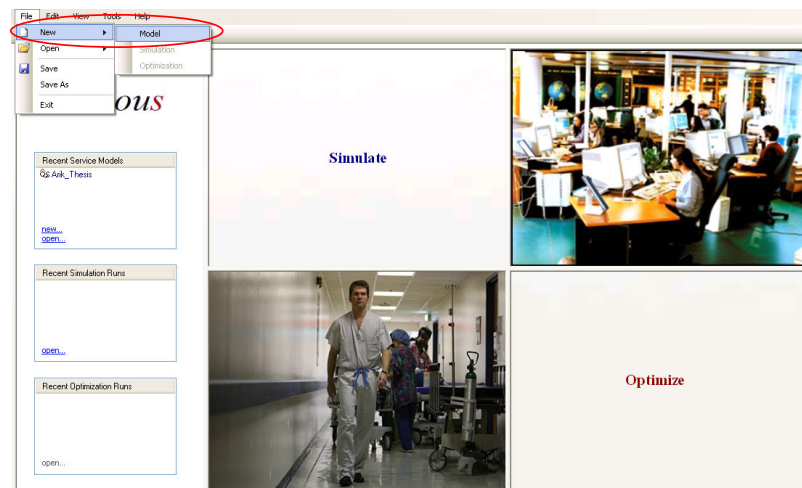
strong support to the intuition that modeling such Call Centers, using *analytical models*, is nearly impossible, and that Simulation here is called for. Moreover, our experience in this large Israeli Call Center (where SBR algorithms have been applied), estimation of required staffing, using software that is based on these *models*, rarely fit the reality needs. Thus, we are hoping that a reliable database, and a suitable representation of the processes that take place in the entire Call Center (via the SBR simulator, to be introduced in section 3.2), will produce a better solution to the problem of setting up adequate staffing levels (relatively to some performance measures).

3.2 Simulating Skills-Based Call Center

A central part of our research will build upon a simulator, which is capable to capture the complex environment of a Call Center. The simulator that we shall use was written as part of an M.Sc. thesis [5] (Operation Research program, IE&M faculty, Technion), by Zohar Feldman.

When simulating a Skills-Based Call Center, the following factors (that were discussed in Section 3.1) must be considered: control policies, types of arriving calls, skills of the groups of agents and so on. These factors serve as basic input to the simulator. The program allows its users to perform evaluations of the Call Center, relatively to some performance measures. For example, after feeding in relevant inputs, one can analyze the abandonment rate, given certain routing policies. Also, having a certain model of a Call Center initialized, an optimization (built-in) tool for staffing levels is also available. To conclude this section, we now present some screen-shots that demonstrate the simulator's modeling and analysis capabilities:

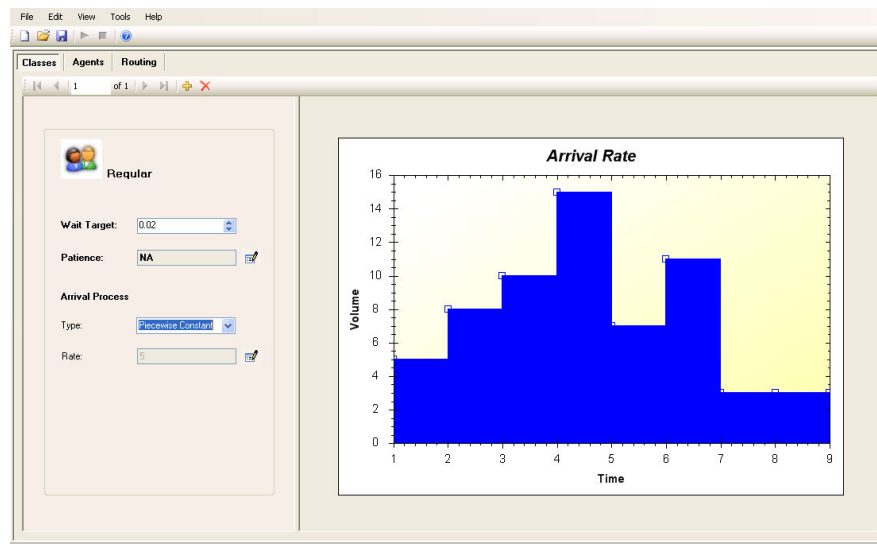
The main screen of the simulator:



At the marked place on-screen, we can choose the File-New-Model option. This will create a new model of a Call Center that will eventually be the basis for our simulations.

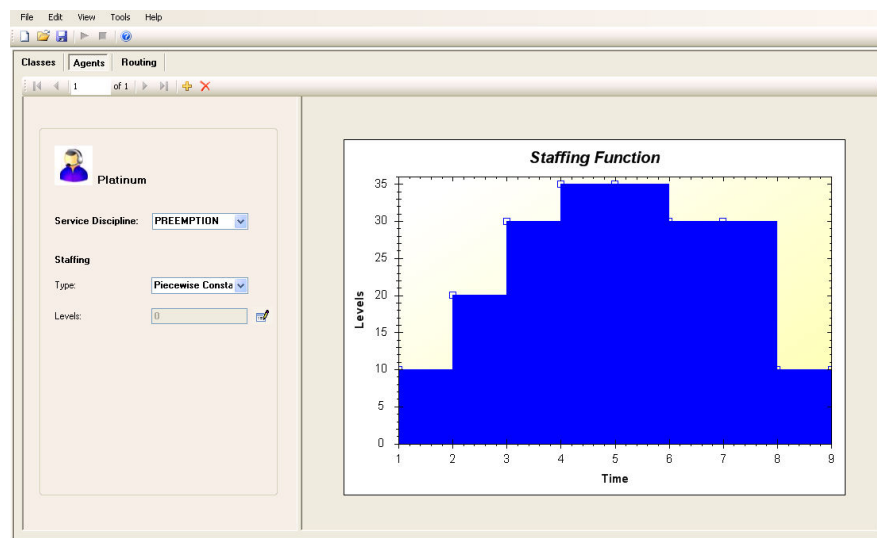
Once inside the "Model" module (after choosing a suitable name for the model), the following screen appears-

The "customer type" screen:



Here we can choose: name of the customers' type (Regular in our example), the target time for serving a customer of this type (one minute in our example), and the arrival rate (in our example - poisson arrivals with piecewise constant rate per hour). We add more customer types, using the yellow '+' sign on the upper part of the screen. Once all desired customer types are chosen, we can move on to the agents screen.

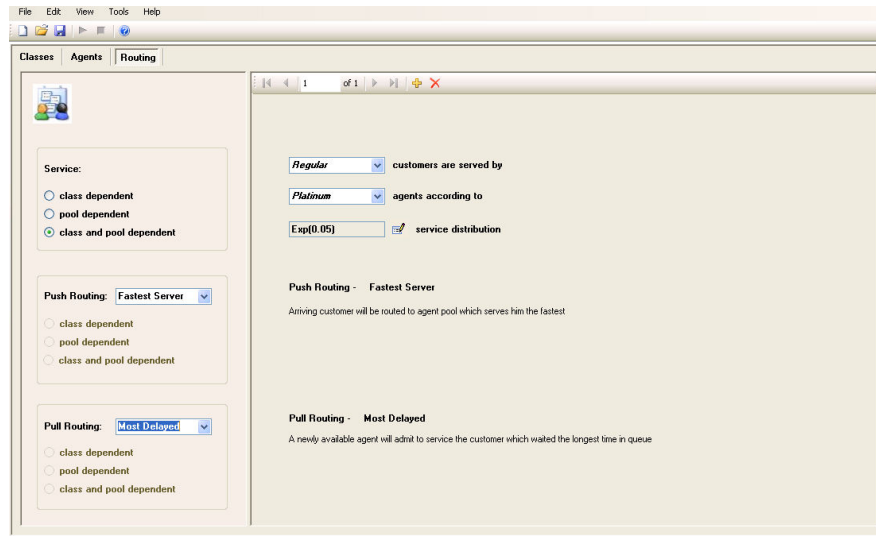
The "agent group" screen:



In this screen, we can choose the name for the agent group (Platinum Service, in our example). We must also choose the service discipline - PREEMPTIVE/EXHAUSTIVE (Exhaustive means that the agent will first finish service, therefore going on a break. Preemptive means that the agent will transfer the customer, to another agent, in case of a break. Also, staffing levels can be chosen (either constant, or piecewise constant). In our example,

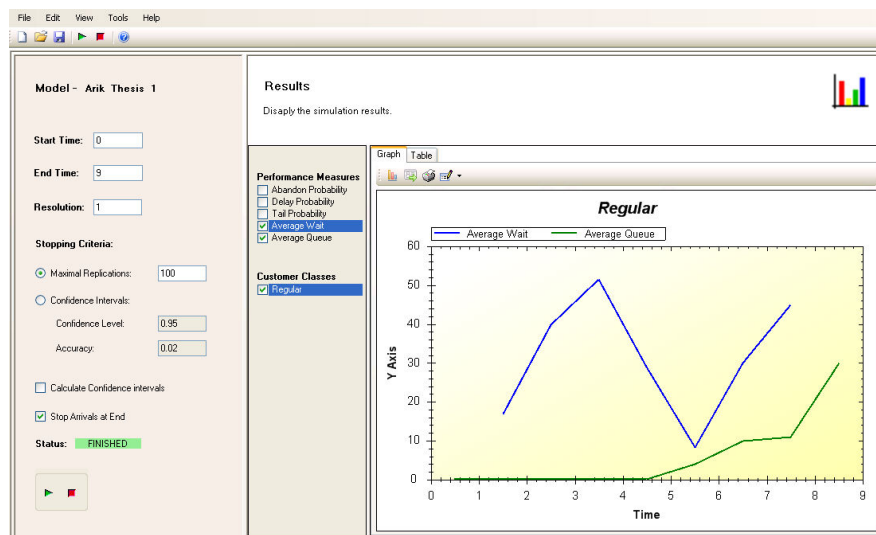
we have determined a piecewise constant staffing levels. After setting all groups of agents, we can move on to the "routing" screen.

The "routing" screen:



In this screen, we match the customers to their main group of agents. A group of agents can serve multiple types of customers, and vice versa. Service time is to be determined (either exponential, log-normal or uniform), for each pair(customer type, agent group). It is also possible to select N/A in the service time option, thus stating that the mentioned group of agents does not serve the selected customer type. One is also to choose the Push and Pull policies (discussed in Section 3.1). Once all routing options are set, we can move to the simulation part, and run realizations of our selected model. In order to get to the "Simulation" module, one should press New-Simulation, while being in the "Model" module.

The "simulation" module:



Here we can choose the desired length of one run, resolution, number of runs and termination conditions. We can also choose to calculate confidence bounds for performance measures, and parameters (such as Average Wait Time, Average Number in Queue and so on). We have chosen a five hour simulation, in an hourly resolution. We see graphs of waiting time, and number of people in queue, as functions of runtime.

4 Databases for Our Simulations

In order to perform a suitable Simulation of a Skills-Based Call Center, we need a comprehensive database. This database will provide us with the data needed to build the Call Center model (Arrival processes, Service times and SBR schemes). Another use for the data can be validation of the Simulation mode, by comparing the Simulation output with the history of what actually happened at the Call Center. Therefore, the application of the databases mentioned in Section 4.1, for our purpose, is ideal.

4.1 The SEE Center and the DataMOCCA Project

The Service Engineering Enterprise (SEE) Center was established in 2007, within the Faculty of Industrial Engineering and Management, at the Technion. The goal of SEE is the development of engineering and scientific principles that support modeling, design and management of Service Enterprises, for example financial services (banking, insurance), health services (hospitals, clinics), government and tele-services (telephone, internet). Presently, SEE's main activity is designing, maintaining and analyzing an accessible repository of resources and data from telephone call-centers, which has been called Project DataMOCCA.

DataMOCCA (Data Models for Call Center Analysis) is a universal model for call center data that, together with its graphical user interface SEESTAT, enables real-time statistical analysis, spanning seconds-to-months resolutions (See also [17]). Currently, DataMOCCA covers call-by-call data of some large call centers (a U.S. bank, an Israeli Telecom company, an Israeli Bank and so on) over periods of 2.5-3 years each. For example, the U.S. bank data has close to 220 million calls, out of which about 40 million were served by agents and the rest by the Interactive Voice Response (IVR) system.

We will focus on a small part (three months) of the Israeli Telecom database, together with Medium Israeli Bank database (which is only now being added to the lab). These databases fit our needs since they include data on arriving customers (including type) and their routing between the different skill-pools, of agents. This data is suitable to serve as input for our simulator. From now on, we shall refer to these databases as our Basic Database.

5 Our Proposed Research

Our research will cover the following topics:

1. *Simulation and Validation*: Constructing (and feeding with data) our *Simulation Model* - We shall use our Basic Database, discussed at the end of Section 4.1, as the foundation to a suitable simulation model. This model will represent both the Israeli Telecom Call Center and the Israeli Medium Bank. We plan to use the simulator, discussed in Section 3.2, throughout our entire research. Validating these models, against the Basic Database, will be our first step. The simulated model will have to be consistent with the reality of our Call Center: Appropriate estimates of Call Center's processes (arrival, abandonment, service) will be obtained, as well as inserting all possible paths (agent pools), for arriving calls.
2. *Current State Inference* - The next step is to develop means for fully understanding the current state of the system, in terms of parameters and performance measures. In most modern Call Centers, shift-managers are aware, at least partially, of their current performance measures and service times. But the majority of data remains unsupervised and unknown. Some data can not be extracted immediately, and there is a need to apply tools (such as simulation) in order to estimate it. We shall develop an algorithm that will use known recent history (customer arrivals and their service process), as well as (partial) information on the current calls within the system (number of calls in queue, served calls etc.). The output of the algorithm will present an estimate of the Call Center's present state. The known data (historical and current) will be easily and automatically transferred, as input to our *Simulation Model*. (This stage can also be useful in practice, for managers who wish to closely monitor their system's performance.) Hopefully, this will be a generic algorithm that can be also used on other Call Centers in the future. Our methodology will follow that of [14], but modifications and new developments might be called for.
3. *Short term Dynamic Staffing* - Having real-time feed of data, along with recent historical data, can serve as a basis for online dynamic staffing. We shall use the *Simulation Model* once again, but this time to forecast future evolution. Furthermore, the *Current State* will be combined with historical data, and an estimation of the time-varying Offered Load ($R(t)$), over t future time units, would become available. Then, under the QED regime, we shall provide good enough assessments, for the number of human agents to be activated, per near future period of time t (the same time for which we will have the Offered Load calculated). Goodness of fit will be determined, relatively to some predetermined targets (relatively to performance measures). One (ideal) target can be, for example, immediately answering incoming calls (as attempted in [18]). Consequently, the levels of staffing can be tested against our Basic Database. (This can also serve as a practical

tool for the Call Center's decision makers.)

4. *Online control* - After inferring the current state of the Call Center, we shall try to build an online SBR control regime, that will use the current state as an input. This online control, along with the mentioned dynamic staffing, will give an appropriate response to unexpected occurrences during the current shift. Suitable dynamic SBR protocols have been developed ([8] for example), yet we hope that using the real-time information and applying simulation will give us the fine tuning that is often missing in off-line models.
5. *Interactions and mutual influences* of customers (and their type) and skilled agents, in our Skills-Based Call Center: In this part, we hope that exploring the Basic Database, using the SEESTAT software (Section 4) and the simulation runs, will give us new insights, regarding various processes and their interactions. Later, we shall make an attempt to match suitable analytical models to what we shall witness empirically.
6. *Static staffing* over longer periods of time, for our Skills-Based Call Center - Having tools for short term staffing (Part 3), models that can predict certain behaviors (Part 4), and our trustable Basic Database, we can make an attempt to deduce suitable staffing protocols, over longer terms (Shifts, days etc.). We could perhaps even try to develop staffing plans over yet longer horizons (aggregate planning). Our experience suggests that, in practise, such plans lack scientific support. This part will conclude the set of tools, which we plan to develop, in order to perform full analysis on a Skill-Based Call Center.
7. *Call Center tomography* - In this part of our research we shall attempt to create a full map of the workflow within our Skills-Based Call Center (hence tomography). Workflow in our case means all possible paths for a call starting with its arrival, continuing with possible sub-stations (Voice-Response-Unit, recorded messages etc.) that it might pass along the way prior to leaving the system (served or unserved). This is not an easy task even in our Call Center basic database, for which we at least theoretically have complete data, including sub-calls per each call (call-by-call database). Some records might contain data that we can not assign to sub-sections (lack of business documentation and human knowledge). Other records might have been lost due to log-systems' shutdown that we know has occurred numerous times in companies which databases we use. In healthcare environments for example the data is even less complete, and process mining techniques are required in order to understand the careflow and improve performance measures [13]. We can try to apply some of these techniques on our data, in order to complete what is missing. Once we will have the full mapping of our Call Center, we then shall be able to use techniques similar to the ones mentioned in [16] and [4] in order to analyze the behavior of our arrivals, service times and delays (waiting times). The

most important advantage of the mentioned analysis is that we perceive the Skills-Based Call Center, and all its processes and paths as a whole. In our opinion, this part may turn out essential for both Staffing (static/dynamic) and Online Control, since it will complete some missing elements that are essential for supporting the latter activates. .

It is important to clarify that our first attempts, for the mentioned above, will begin by separating the Call Center into sub-groups. For example: using only two pools of agents that serve three types of customers. Thus, we shall have a small sub-Call-Center. This will make our experiments easier to conduct and control. Then, hopefully, having positive results for the sub-Call-Center, we shall be able to extend these results to the entire Basic Database, and perhaps to other Skills-Based Call Centers as well.

References

- [1] Aksin, Armony, and Mehrota. The modern call center: A multi-disciplinary perspective on operations management research. *Production and Operations Management*, 16(6):665–668, 2007. Available from: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.129.4795&rep=rep1&type=pdf>. 3
- [2] Borst, Mandelbaum, and Reiman. Dimensioning large call centers. *Operation Research*, 52:17–34, 2004. Available from: <http://iew3.technion.ac.il/serveng/References/Dimensioning.pdf>. 3, 4, 5
- [3] Brown, Gans, Mandelbaum, Sakov, Zeltyn, Zhao, and Haipeng. Statistical analysis of a telephone call center: A queueing-science perspective. *Journal of the American Statistical Association*, 100:36–50, 2005. Available from: http://iew3.technion.ac.il/serveng/References/JASA_callcenter.pdf. 5
- [4] Castro, Coates, Liang, Nowak, and Yu. Network tomography: Recent developments. *Statistical Science*, 19(3):499–517, 2004. Available from: http://projecteuclid.org/DPubS/Repository/1.0/Disseminate?view=body&id=pdfview_1&handle=euclid.ss/1110999312. 17
- [5] Feldman. Optimal staffing of systems with skills-based-routing (sbr). Master’s thesis, Technion, 2008. Available from: http://iew3.technion.ac.il/serveng/References/Zohar_Thesis.pdf. 11
- [6] Gans, Koole, and Mandelbaum. Telephone call centers: Tutorial, review and research prospects. *MSOM*, 5(2):79141, 2003. Available from: <http://iew3.technion.ac.il/serveng/References/Gans-Koole-Mandelbaum-CCReview.pdf>. 3, 8
- [7] Garnett, Mandelbaum, and Reiman. Designing a call center with impatient customers. *MSOM*, 4(3):208–227, 2002. Available from: <http://iew3.technion.ac.il/serveng/References/Garnett.pdf>. 4
- [8] Gurvich, Armony, and Mandelbaum. Service level differentiation in call centers with fully flexible servers. *Management Science*, 54(2):279–294, 2008. Available from: http://iew3.technion.ac.il/serveng/References/Vdesign_pub.pdf. 17
- [9] Gurvich, Luedtke, and Tezcan. Staffing call-centers with uncertain demand forecasts: A chance-constrained optimization approach. 2009. Available from: http://www.kellogg.northwestern.edu/faculty/gurvich/personal/Robust_05_09_09-21_names.pdf. 3
- [10] Liberman. Empirical analysis of service centers: A queueing-science perspective, 2009. Available from: http://iew3.technion.ac.il/serveng/References/Buenos_Aires_040809.pdf. 10

- [11] Mandelbaum and Garnett. An introduction to skills-based routing and its operational complexities. Technical report, Technion, 2000. 9, 10
- [12] Mandelbaum, Sakov, and Zeltyn. Empirical analysis of a call center. Technical report, Technion, 2000. Available from: <http://iew3.technion.ac.il/serveng/References/ccdata.pdf>. 2
- [13] Mans, Schonenberg, Song, and van der Aalst. Process mining in healthcare: A case study. Available from: <http://rseguel.googlepages.com/03-mans2008.pdf>. 17
- [14] Marmur. Simulation-based real-time decision-support system for emergency departments. 2009. Available from: http://iew3.technion.ac.il/serveng/References/Yariv_WSC_seminar.pdf. 7, 16
- [15] Reich. The workload process: Modelling, inference and applications. Master’s thesis, Technion, 2010. 4, 5
- [16] Serban and Zhang. Discovery, visualization and performance analysis of enterprise workflow. *Computational Statistics & Data Analysis*, 51(5):2670–2687, 2007. Available from: <http://www2.isye.gatech.edu/~nserban/pave.pdf>. 17
- [17] Trofimov, Feigin, Mandelbaum, Ishay, and Nadjharov. Datamocca - model description and introduction to user interface. Technical report, Technion, 2004. Available from: http://iew3.technion.ac.il/serveng/References/DataMocca_February_2008_CC_Forum_Lecture.pdf. 15
- [18] Whitt. Dynamic staffing in a telephone call center aiming to immediately answer all calls. *Operations Research Letters*, 24:205–212, 1999. Available from: <http://www.columbia.edu/~ww2040/DynamicStaffing.pdf>. 7, 16