Creating operational shift schedules for third-level IT support: challenges, models and case study

Segev Wasserkrug*, Shai Taub, Sergey Zeltyn, Dagan Gilat, Vladimir Lipets and Zohar Feldman

IBM Haifa Research Lab,
Haifa University,
Mount Carmel,
Haifa, 31905, Israel
E-mail: segevw@il.ibm.com
E-mail: taub@il.ibm.com
E-mail: sergeyz@il.ibm.com
E-mail: dagang@il.ibm.com
E-mail: lipets@il.ibm.com
E-mail: zoharf@il.ibm.com
*Corresponding author

Avishai Mandelbaum

Industrial Engineering and Management,
Technion – Israel Institute of Technology,
Haifa, 32000, Israel
E-mail: avim@ie.technion.ac.il

Abstract: IT support can be divided into first-level support, second-level support and third-level support. Although there is a large body of existing work regarding demand forecasting and shift schedule creation for various domains such as call centres, very little work exists for second- and third-level IT support. Moreover, there is a significant difference between such support and other types of services. As a result, current best practices for scheduling such work are not based on demand, but rather on primitive rules of thumb. Due to the increasing number of people providing such support, theory and practice is sorely needed for scheduling second- and third-level support shifts according to actual demand. In this work, we present an end-to-end methodology for forecasting and scheduling this type of work. We also present a case study in which this methodology demonstrated significant potential savings in terms of manpower resources.

Keywords: forecasting; staffing; rostering, IT support shift scheduling; workforce management; optimisation.

Reference to this paper should be made as follows: Wasserkrug, S., Taub, S., Zeltyn, S., Gilat, D., Lipets, V., Feldman, Z. and Mandelbaum, A. (2008) 'Creating operational shift schedules for third-level IT support: challenges, models and case study', *Int. J. Services Operations and Informatics*, Vol. 3, Nos. 3/4, pp.242–257.

Biographical notes: Segev Wasserkrug is a Research Staff Member in IBM Haifa Labs, as well as the Manager of the Business Optimization Group. Segev has his PhD in information systems engineering and both a BA and an MSc degrees in computer science from the Technion – Israel Institute of Technology. Segev has practical and academic background in the areas of operations research (including optimisation, workforce predictive models and simulation), computer science and machine learning. Segev has over 13 years practical experience, including 6 years in applying analytics to solving customer problems, specifically in the area of workforce scheduling.

Shai Taub has a BA and an MSc in compute science from the Tel Aviv University. Shai has over 8 years practical experience in developing and applying advanced algorithms for real-world problems, including 2 years in the area of workforce management. Shai is a senior technical member of the SWOPS team. Shai's research interests are in the area of combinatorial optimisation.

Sergey Zeltyn is a Staff Member in IBM Haifa Research Lab, Haifa University, Haifa, Israel. He has been involved in several projects in the fields of workforce management, forecasting and statistical analysis. Before he joined IBM, he had been working as the manager of the Service Engineering Enterprise Research Center in the Technion – Israel Institute of Technology. His scientific interests include operations research, queuing theory, stochastic modelling of service systems and applied statistics. He has vast experience in data analysis, model development and consulting in the field of telephone call centres. He has MSc and PhD degrees in statistics from the Technion.

Dagan Gilat is Senior Manager of the Business Transformation & Optimization group at the IBM Haifa Research Lab, Haifa University, Haifa, Israel. He is responsible for the development of mathematical optimisation solutions and for developing methods for Business Transformation, such as CBM II, which is part of the overall Business Performance Transformation Services (BPTS) activities in IBM. He has over 15 years experience in technology research and development. He studied at the Technion – Israel Institute of Technology, where he received a PhD in information systems, M.Sc in operations research and B.A in computer science.

Vladimir Lipets is a Research Staff Member at IBM Haifa Research Lab, Haifa University, Haifa, Israel. Vladimir has a PhD in computer science from the Ben Gurion University in the area of optimisation algorithms. Vladimir also has more than 6 years programming and application development experience, and a solid background in software architecture and design. Vladimir also has more than 2 years experience in workforce management algorithms. Vladimir's interests include combinatorial optimisation, graph theory and pattern recognition.

Zohar Feldman is a Research Staff Member in IBM Haifa Research Lab, Haifa University, Haifa, Israel, working in the area of services workforce management. He has over 4 years practical and academic experience in the areas of operations research (including optimisation, simulation), queuing theory, industrial algorithm development and the application of operations research techniques to services workforce management. He has also a good background in the area of computer vision. He has a BSc degree in service engineering and management, and should complete soon his MSc in operations research. His MSc thesis deals with finding optimal staffing levels in time-varying environments.

Avishai Mandelbaum is a Professor at the Technion – Israel Institute of Technology, Haifa, Israel, working mainly in the area of stochastic processes. His work has been mainly theoretical but with a clear practical inspiration, aimed at supporting decision making in complex operations. In his recent activities, through research and teaching, he has been attempting to specialise the theory of queues and queuing networks to service networks, focusing on those with high level of customer/server interaction. He has been teaching, conducting workshops and consulting numerous organisations on how to properly design, control and manage service operations, in Europe and the USA.

1 Introduction

An increasingly large number of enterprises are completely outsourcing their IT support requirements. The IT support is then provided as a service by large service providing organisations such as IBM, EDS and Accenture.

Such support services include the following three main types of support: first-level support, second-level support and third-level support. First-level support provides a rudimentary level of support requiring very little skill. Resetting a user's password is a typical problem handled by first-level support. If a first-level support representative cannot resolve the issue, it is passed on to second-level support. A second-level support representative attempts to resolve the issue by following a resolution process defined for each type of issue. If the issue cannot be resolved by the second-level support representative, it is directed to third-level support. Therefore, third-level support requires significant skill levels, as well as deep specialisation in specific product platforms. For example, updating the version of an operating system on a server is an example of a task that is carried out by third-level support personnel.

In many cases, all of the above types of support are provided by people working in large service centres, which provide support to customers located around the globe. Due to the varied geographic locations of the customers, and the resultant time zone differences, such support must be provided 24 hours a day, 7 days a week. Consequently, the people providing the support work in shifts. An additional feature of such work is that there are fluctuations in demand for these services on different days of the week and at different hours of the day. Creating shift schedules that minimise the costs of providing the service, while maintaining the required quality of service, is a challenge.

Due to the relatively low skill level required, first-level support is traditionally provided via call centres. Shift scheduling for call centres is a topic that has been extensively researched from numerous angles and perspectives (see Section 2). Therefore, a wide variety of techniques and models exist for scheduling such work. In addition, several products that support the scheduling of first-level support work are available. However, almost no research has been carried out on scheduling second- or third-level support. Moreover, significant differences exist between providing first-level support from call centres and providing second- and third-level support. Examples of such differences include a relatively low number of load items per interval, and a large amount of work required to resolve each such item. These differences violate many of the assumptions underlying the models and methods used to schedule call centre support personnel. In order to schedule third-level support, existing models need to be adapted or new models need to be created.

This paper describes a set of models and methodologies used to implement the process for scheduling third-level support, as depicted in Figure 1. This paper also details a case study that uses this implemented process to schedule a team providing third-level support. Therefore, our contribution is in both creating methodologies, models and algorithms for scheduling third-level support, and demonstrating the applicability of this approach in an actual scenario.

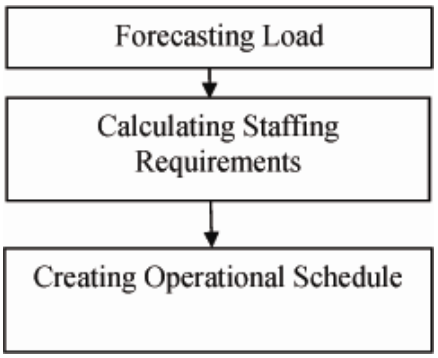
The rest of this paper is organised as follows: Section 2 discusses related work. Section 3 describes the specific third-level support characteristics that influence the forecasting and scheduling process. Section 4 details some of the reasons why existing models are inappropriate for third-level support. Section 5 outlines our methodologies and algorithms for creating shift schedules of such work. Section 6 describes the application of the results of our methodology. We close with Section 7, which summarises this work and proposes possible avenues for future work.

2 Related work

The most widely used process for creating shift schedules consist of three main steps (see Figure 1):

- *Forecasting load (or demand)*: This entails creating a forecast regarding the amount of incoming work during different times of the week/day.
- *Calculating staffing requirements*: This involves translating the amount of forecast work into the number of people (and skills) required at each point in time in order to meet a predetermined service level.
- *Creation of the actual schedule*: This process creates the actual operational schedule, while ensuring that the right number of people (and skills) is available at each point in time and that employee's personnel preferences and labour rules/regulations are satisfied.

Figure 1 Scheduling process



As discussed in the introduction, very little work exists for scheduling third-level support. Therefore, two types of related work will be surveyed. The first are generic techniques that can be used to implement the separate process steps depicted in Figure1. The second type are existing models and techniques used in call centres to implement shift schedule creation.

2.1 Generic techniques

Forecasting load is a process used to generate a model of the load expected in the future. This forecast must include information such as the number of load items that will occur in each interval, the skill(s) required to handle each load item and the amount of work required for each work item. In order to generate operational schedules, such a forecast must usually be generated two to six weeks in advance. (This period is referred to as the forecast horizon.) This forecast may be periodically updated as new information arrives.

Two main types of techniques are generally used for load forecasting. The first involves statistical techniques such as time series analysis, based on models such as moving average, ARIMA (Brockwell and Davis, 2002), or regression techniques. The second uses machine learning techniques, based on models such as decision trees and statistical learning algorithms (Russel and Norvig, 2003). Both of the above types of models are data-driven methods, i.e. they generate a forecast of the demand based on historical data.

Translating the demand into staffing requirements to get the number of workers required at each point in time is usually carried out based on queuing network models. In some cases, analytical models, such as those based on M/M/N queues (Gross and Harris, 1998) are used, while more complex cases (e.g. Avramidis et al., 2007) use discrete event simulation of queuing networks.

Creating an operational schedule is usually carried out based on heuristic algorithms, mixed integer programming optimisation or a combination of the two.

The challenge lies in combining the above techniques to create an end-to-end scheduling solution for a specific type of work and finding a model that is suited to this type of work for each of the stages outlined above. In some cases, such as the work in call centres, standard models for all of the above exist and are widely used in commercial products. This will be further elaborated upon in a subsequent section. For third-level support work, however, there are no widely known appropriate models.

2.2 Models for call centre shift scheduling

For each of the scheduling components in Figure 1, the most widely used models in call centres are the following:

- *Forecasting load*: The forecast horizon is divided into short *forecasting intervals*, typically 15 or 30 minutes each. In each forecasting interval, arrivals are assumed to follow a homogeneous Poisson process. Under this model, forecasting predicts the average arrival rate in each interval of the forecast horizon. The average interval arrival rate is usually calculated using simple time series techniques such as moving average or single exponential smoothing (Brockwell and Davis, 2002).
- *Calculating staffing requirements*: In most cases, the required number of people is calculated based on analytical queuing models and steady-state analysis, although in a small number of cases, simulation modelling may be used. These analytical models are either M/M/N when customers do not abandon, or M/M/N+M (Mandelbaum and Zeltyn, 2007) when abandonment needs to be taken into account. Based on these models, the number of people is calculated per staffing interval. This approach assumes that a staffing interval corresponds to a forecasting interval. For each such staffing interval, the minimum number of people is calculated so it satisfies some

service-level constraint. An example of such a constraint is '80% of all calls must be answered within 20 seconds' or 'at most 8% of the calls may be abandoned'. Note that this method assumes that the above service-level constraint must be satisfied in each staffing interval (which is the same length as the forecasting interval).

- *Creating operational schedules*: The actual operational schedules are created using either heuristics or mixed integer programming. (See e.g. Mason et al., 1998; Brusco and Jacobs, 2000.) In both cases, the problem is formulated based on scheduling work relevant to call centre work.

In all of the above components, use of the above set of models implies, either explicitly or implicitly, the existence of specific assumptions. Examples of such assumptions are piecewise homogeneous Poisson arrivals, the need to maintain service levels in each forecasting interval and the applicability of steady-state analysis to obtain the staffing requirements in each staffing interval. It is possible that such assumptions may even not hold for some call centres. Due to this fact, together with the growing complexity of such call centre operations, load forecasting, staffing and scheduling of such operations are still very much an active field of research. An example of such research is request routing and agent scheduling in the presence of heterogeneous skill sets. For an overview of the research in the call centre area, as well as the current challenges, see Avramidis et al. (2004), Gans et al. (2003) and Green et al., (2007).

3 Characteristic of third-level support work

In this section, we describe the characteristics of third-level support work relevant to workforce scheduling. These characteristics are as follows:

- an arrival process that includes both the characteristics of load item arrivals and the amount of work required by each such item
- the service-level constraints that have to be maintained
- a process workflow for handling the load items
- specific scheduling rules relevant to this type of work.

The following sections detail each of these items. In these sections, as well as in the remainder of this work, the characteristics and data of the team that participated in the case study are used for illustration and clarification. This team consisted of approximately 25 agents, providing third-level support 24 hours a day, 7 days a week.

3.1 Arrival process

The type of load item being handled by third-level support work is called a problem ticket, or 'ticket' for short. Due to the nature of the work carried out by such support, the arrival process has the following characteristics:

- on average, a small number of tickets arrive per time interval
- a large amount of work is required per ticket
- there is a large demand variance in the arrival pattern.

For example, for the team that participated in the case study, about one or two problem tickets arrived per hour, but the average amount of work required per problem ticket was 1 hour and 47 minutes. In addition, there was a large variance in the number of problem tickets arriving per hour. There were many hours in which no new problem tickets arrived and there were also several hours in which more than ten tickets entered the system.

3.2 Service level

With regards to service level, each problem ticket has an associated severity. For each such severity, a different service-level constraint has to be maintained. For the team participating in our case study, five severities existed, numbered 1 to 5. Severity 1 tickets required the highest service level, while Severity 5 tickets did not have an associated service level (i.e. best efforts). The exact definition of the service-level constraints associated with Severities 1 through 4 are as follows:

- *Severity 1*: 90% of the tickets must be resolved within 4 hours.
- *Severity 2*: 85% of the tickets must be resolved within 8 hours.
- *Severity 3*: 80% of the tickets must be resolved within 3 days.
- *Severity 4*: 80% of the tickets must be resolved within 5 days.

An unusual aspect regarding the above Service-Level Agreements (SLAs) is that due to the large difference in SLAs between Severities 1 and 2 tickets (which are specified in hours) and Severities 3 and 4 (specified in days), a *mixed preemption* service policy is used. In other words, when either Severity 1 or Severity 2 ticket arrives, if no service representatives are available, a service representative working on a Severity 3, Severity 4 or Severity 5 ticket stops working on the low severity ticket and begins working on the newly arrived ticket. However, a Severity 3 ticket will not interrupt the work of service representatives working on Severity 4 or Severity 5 tickets.

An important point to note regarding these SLAs is that they need to be maintained at the monthly level. For example, it is acceptable to have a time interval in which less than 90% of the Severity 1 tickets are resolved within four hours, as long as the required percentage is maintained at the monthly level.

3.3 Workflow of problem ticket handling

A distinctive characteristic of third-level support is that such support may follow a complex process. Although a problem ticket may sometimes be completely handled by the team for which it was initially opened, in many cases, it may be routed to other parties. A ticket may be routed to one of the following possible parties:

- The customer (i.e. the party relying on the support services). Such routing may take place, for example, if the customer needs to provide additional information about the problem or carry out some action as part of the problem resolution.
- Another team from the same service provider, i.e. the service provider for the team for which scheduling is carried out. An example of such routing is a case in which the same service provider provides third-level support for both a specific database management system (e.g. Oracle or DB2) and a specific operating system

(e.g. Windows or Linux). Since each type of support requires distinct knowledge, this support is provided by two separate teams, which we will call the DB team and the OS team, respectively. A problem ticket may first be classified as a problem associated with the installation options of the operating system and routed to the OS team. After carrying out some work, the problem may be found to be associated with the configuration of the database management system. At this point, the problem will be routed to the DB team.

- An additional support provider. This may occur, for example, if correcting a problem in an operating system requires a patch from the provider of that operating system.

The above routing may be arbitrarily complex. A problem ticket may go through several such routing steps, where at each step it is routed to one of the parties described above. In addition, the process a ticket undergoes impacts the SLA for the ticket. The time a ticket spends either assigned to the customer or to an additional service provider is not counted towards the service-level constraint defined in the previous section. To clarify, consider the case in which there are 8 hours between the time that a Severity 1 ticket is opened and until it is closed, but the ticket spent five of those hours waiting for additional information to be provided by the customer. In this case, the time counted is only 3 hours, so this ticket would not violate the service-level constraint.

3.4 Scheduling rules

Several scheduling rules had to be maintained in the schedules generated for the case study team. A thorough description of all of these rules is beyond the scope of this work. However, in this section several example rules will be described.

One of the scheduling rules is the overlap rule. The overlap rule results from the fact that the length of time from the point at which the problem ticket is opened and until it is closed may be several hours or even days. The reasons for these large time periods may include relatively large amount of work required per problem ticket, the service-level constraint that allows a large amount of time to pass until the ticket is resolved and the fact that a significant proportion of this time may pass while the ticket is assigned to a party other than the team being scheduled. Therefore, it is quite common that a ticket will start being handled by one agent (denoted by Agent A), and will then have to be handled by another agent (denoted by Agent B) when the shifts change. In order to enable such a transition, there needs to be an overlap between the times of the shifts. The shift of Agent B needs to start before the shift of Agent A ends, in order to enable an orderly transition of tickets between the agents. The overlap rule ensures such an overlap between the shift times.

Another relevant rule is regularity. The regularity rule is intended to ensure that a person will be scheduled for work during the same hours for some predefined time period such as a week or a month. Under this rule, for example, a person may be scheduled to work during a day shift or a night shift. However, if a person is scheduled to work during the night shift on Monday, for example, the person should also be scheduled to work during night shifts for the rest of that week, Tuesday to Friday. The reasons for the need for the regularity rule can be widespread, and include both taking into account the time it takes a person to adjust to the schedule and transportation issues in developing countries such as India, where allowances must be made for very lengthy round trips (up to 6 hours) between a person's work and home.

Another rule we look into is *one shift per day*. This rule states that a person should be scheduled to at most one shift during a 24 hour period.

4 Inappropriateness of existing models

Due to the unique characteristics of third-level support work, there are no known methods and models appropriate for scheduling such work. This section details some of the reasons why the use of models for scheduling call centre work is problematic for this domain.

In call centres, steady-state analysis is carried out per staffing interval using M/M/N or M/M/N+N queues to derive staffing requirements from load forecasting. However, the use of this model assumes the following:

- the arrivals in each staffing intervals are homogenously Poisson.
- the system reaches steady state quickly in each such staffing interval (i.e. steady state is reached close to the beginning of the interval)
- the staffing is derived by requiring that the service-level constraint is met in each such staffing interval. For third-level support, the service level is not only required to be met only at a monthly level, but also specified in term of the sojourn time (i.e. from the time the request arrives and waits for service until it is resolved). In most call centre scenarios, the SLA is specified in terms of waiting time (i.e. the time until service *starts*).

When analysing the arrival patterns of the problem tickets, we ascertained that staffing intervals must be no longer than 1 hour in order for the assumption of a homogenous Poisson arrival to be reasonable. For third-level support, because very few tickets arrive per hour and because each such ticket requires a large amount of handling time, it is unreasonable to expect that within an hour – a short enough amount of time – the system will reach a steady state. In addition, the service-level constraint for third-level support is defined at a monthly level, rather than at the staffing interval level. As a result, using M/M/N steady-state models to derive staffing does not seem to be suitable.

Classical staffing models are also inappropriate because they assume that each incoming load item consists of a single piece of work, which is handled by the agent responding to this call. These models do not take into account the possibility of workflows in which a load item may be routed (and worked on) by several parties, only one of which is the party for which the staffing is being derived. In addition, call centre models do not address the possibility that only some of the time required to handle the load item is counted towards the service-level constraint.

The mixed preemption policy poses another challenge for analytical models. Existing models assume either a fully preemptive, or a non-preemptive policy, but not both.

Furthermore, unlike call centres, in which customers which experience excessive waits may give up and disconnect, there is no abandonment in third-level support work; once a problem ticket is opened, it remains open until it is being handled. Therefore, M/M/N+M are not relevant in such a case. Moreover, additional care needs to be taken when translating the load forecast into staffing requirements, since understaffing can result in an unstable system when abandonment does not exist.

5 Scheduling third-level support work

In our methodology, we translated the forecasted load into staffing requirements using a simulation model. The decision to use a simulation model was made because state-of-the-art analytical queuing methods cannot appropriately describe the complexities associated with such work. It is important to note that the use of a simulation model also relaxes the requirements associated with creating a load forecasting model. If analytical methods are used, we need a load forecasting model that correctly describes the arrival process and can be analysed using analytical queuing methods. Using a simulation model only requires simulation of the forecasted load.

The remainder of this section describes the implementation of the three components required to schedule agents providing third-level support.

5.1 Staffing simulation model

The simulation model we created was designed to take into account the unique characteristics of third-level support work. One of the most important characteristics of this work is the fact that, during its lifetime, a problem ticket may be assigned to one of several parties. As detailed information regarding the routing of tickets to different parties could not be obtained, a relatively simple model was created. This model is described in Figure 2.

Figure 2 Ticket handling process model

According to the process described in Figure 2, a ticket is modelled by a sequential process of tasks. The description of these tasks is as follows:

- Task1: The amount of work required on this ticket before being transferred to a party other than the team being scheduled. Task1 does not include the time the ticket waits in the queue before work on it has begun.
- Task2: The total amount of time the ticket is assigned either to the customer or to an additional service provider. This includes the time the ticket spends waiting for the customer or service provider.
- Task3: The total amount of time the ticket is assigned to another team from the same service provider. This includes the time the ticket spends waiting for this team.
- Task4: Once the ticket is routed back to the team being scheduled (i.e. the same team that carried out the work in the Task1 phase), this is the remaining amount of work required on this ticket.

As described above, the amount of time for Task2 is not taken into account when deciding whether a ticket meets the service-level constraint. Formally, let T_i denote the amount of time specifying the service-level constraint for a given ticket of severity i. For example, for all Severity 1 tickets, $T_i = 4$ hours. Equation (1) describes the condition that must hold in order for the constraint to be met:

$$W_{\text{Task1}} + \text{Task1} + \text{Task3} + W_{\text{Task4}} + \text{Task} \leq T_i \quad (1)$$

where:

- W_{Task1} and W_{Task4} is the time the ticket waits for the team being scheduled to start working on Task1 and Task4, respectively.
- Task1 and Task4 are the working (service) times required for Task1 and Task4, respectively.
- Task3 is the time the ticket spends being assigned to another team from the same service provider. This includes the queuing time for that team.

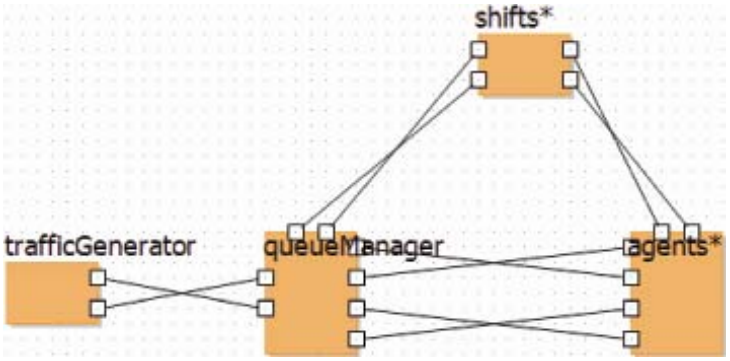
Not all tickets are assigned to all four types of parties; therefore, it is possible that for a specific ticket, either Task2 or Task3 (or both) is zero.

We created a simulation model based on the above flow of a problem ticket, using the Anylogic 5.5 simulation engine by XJ Technologies (http://www.xjtek.com). The model also had to take into account individual behaviour. This was due to the additional characteristics of third-level support, for example the fact that for non-critical severity tickets (i.e. tickets of Severities 3–5) the tickets are not passed between agents. Therefore, at the end of a service representative's shift, all Severities 3–5 tickets assigned to this service representative are put 'on hold', and work on them only resumes in the next working shift of this service representative. In order to model such individual behaviour, the agent-based modelling paradigm was used.

Figure 3 depicts the main components of this simulation model. These components are as follows:

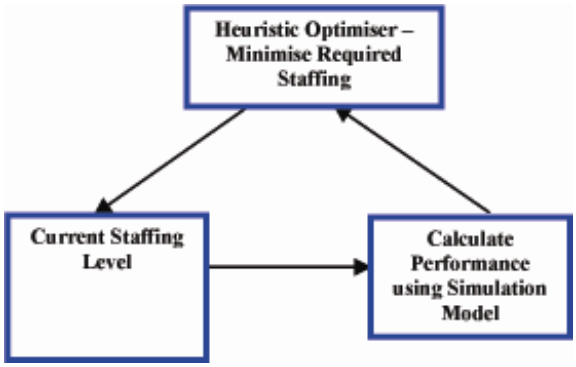
- *trafficGenerator*: This is the component responsible for generating the problem tickets, as well as the service process followed by each ticket (i.e. the service process in Figure 2).
- *queueManager*: This component models the service policy and assigns the requests to agents according to the priority of the requests. This component also manages the mixed preemption policy.
- *agents*: This component is responsible for modelling the relevant aspects of the individual service representatives when providing the service.
- *shifts*: This component is responsible for notifying agents when they are scheduled for work.

Figure 3 Staffing simulation model (see online version for colours)



Using this simulation model, the required staffing levels that maintain the severity differentiated service-level requirements can be obtained, for example, by coupling this simulation model with an optimisation technique suitable for simulation (e.g. Tabu/Scatter search). This can be done using a process such as the one depicted in Figure 4.

Figure 4 Finding required staffing levels (see online version for colours)



5.2 Load forecasting model

The forecast model has to include the following components:

- arrival process
- amount of work (i.e. Task1 + Task4) required by each ticket
- amount of time each ticket spends assigned to either the customer or an additional service provider (Task2)
- amount of time each ticket spends assigned to another team from the same service provider (Task3).

Based on initial analysis, it was decided that for the case study team, a separate forecasting model had to be created for each severity due to the difference in arrival characteristics for the different severities. We created this forecasting model based on a large amount of historical data that was available for the case study team.

To model the arrival process, we used the following methodology:

- the number of arrivals per day was forecasted
- distribution of the arrivals within a given day was found.

The form of the model for forecasting the number of arrivals per day appears in equation (2). In this equation, X_t is the number of arrivals in day t, M_t is a deterministic function of $X1, \ldots, X_{t-1}$, and Y_t is a random variable with some distribution.

$$X_t = M_t + Y_t. \tag{2}$$

We tried several time series methodologies for obtaining M_t, including moving average, simple and double exponential smoothing and time series techniques that take seasonality into account. The simple moving average was found to be the best technique.

We obtained Y_t by fitting a distribution to the residuals $X_1 - M_1$, $X_2 - M_2$, …. This technique resulted in a high correlation (0.85) between the forecasted values and the actual values.

Given the number of arrivals per day, no theoretical distribution was found that could represent the arrival distribution within the day. Therefore, we used the empirical arrival distribution, which is sufficient when used in conjunction with the simulation-based staffing models discussed in Subsection 5.1.

With regards to the amount of work required by each ticket, it was found that the work distribution was subexponential (Goldie and Kluppelberg, 1997). An example of a subexponential distribution is the lognormal distribution. Once the distribution was chosen, maximum likelihood estimators were found for the parameters of the distribution. The fact that the work time distribution is subexponential contradicts another assumption underlying the use of M/M/N models – the assumption that the service time distribution is exponential.

Finally, the durations of Task2 and Task3 were also found using distribution fitting techniques.

5.3 Creating operational schedules

Once the staffing requirements are determined from the forecasted demand and the service-level constraints, the actual operational schedules were generated using mixed integer programming. The scheduling had to be articulated as a set of linear constraints in order to enable such schedule generation.

In general, the problem is formulated as an assignment problem. For each agent i and each possible shift j for this agent, there is a binary variable X_{ij} that has the value 1 if agent i is assigned to shift j and 0 otherwise. Then, for each rule there is a set of linear constraints that ensure this rule is being fulfilled. The objective function is defined to optimise objective measures such as costs, and subjective measures such as employee preferences. While a full description of the model is beyond the scope of this work, we exemplify the specification by an illustrative formulation of the overlap rule.

Let there be two shifts s_1 and s_2 such that shift s_2 starts an hour before the end of shift s_1. Assume that in order for the overlap rule to be satisfied, if an agent is assigned to shift s_1, an agent must be assigned to shift s_2. Additionally, assume there are ten agents in the team. For each such agent, there is a binary variable X_{ij} that is 1 if agent i is assigned to shift j and 0 otherwise. In order to ensure that if an agent is assigned to shift s_1, an agent must be assigned to shift s_2, additional binary variables S_1 and S_2 are defined. Given this, equations (3)–(5) ensure that the required conditions are maintained.

$$\forall i,\ X_{i1} \le S_1 \tag{3}$$

$$S_2 \le \sum_{i=1}^{10} X_{i2} \tag{4}$$

$$S_1 \le S_2. \tag{5}$$

The required conditions are maintained as follows. Equation (3) ensures that if an agent is assigned to s_1, (i.e. one of the variables $X_{i1},\ \ i = 1 \cdots 10$ is 1) then variable S_1 will be 1. Equation (4) ensures that if S_2 is 1, then an agent must be assigned to shift s_2. Finally,

equation (5) ensures that if S_1 is 1, then so is S_2. Therefore, if one of the variables $X_{i1}, \quad i = 1 \cdots 10$ is equal to 1, then one of the variables $X_{i2}, \quad i = 1 \cdots 10$ must also be equal to 1.

Other rules are similarly articulated using a similar set of linear inequalities.

6 Applying the scheduling methodology

The main goal of applying an analytical scheduling solution is to reduce the manpower costs associated with providing the IT support. To this end, the above methodology is applied in two ways, corresponding to two different planning horizons, which are as follows:

- *Strategic (capacity) planning*: Calculating the long-term minimum team size required to provide the support while maintaining the service-level constraints.
- *Operational planning*: Generating the optimal operational schedule to ensure that the service-level constraint is met using the existing team size. The use of the above methodology for this purpose is a straightforward implementation of the process depicted in Figure 1.

To determine the minimum team size for the case study team, we first analysed the demand load for any long-term trends. The analysis showed that there was no increasing trend in the load. This indicates that, given no significant changes in the trends of the future load, the minimum team size that was able to provide the support in the past should also be sufficient to provide support in the future.

To calculate the minimum team size required to provide the support in the past, we created optimal operational schedules for all past months. The schedules took into account factors such as the need for vacations and training. Basing the team size on actual operational schedules, while taking the required staffing levels into account, is much more reliable than basing this decision on staffing models alone. This is true because the need to satisfy scheduling rules and practices, such as breaks, vacations and tardiness, may increase the number of people required to provide the SLAs.

When we carried out the above process for strategic planning with the case study team, our results indicated that the current team size could be significantly reduced and still maintain the service-level constraints. Obviously, care needs to be taken in implementing such a reduction. This is due to reasons as follows:

- People management issues associated with removing a person from a team. It is important to note that for the case study team, removing a person from the team did not mean that this person's employment was terminated; the people were transferred to other teams in which their skills were needed. However, even in such a case, complex human management issues are involved.
- Removing and/or adding people to a team cannot be carried out instantaneously. Therefore, if a person is removed from a team due to the recommendations of the analysis, and, contrary to the analysis results, the service level is negatively impacted; it is not possible to quickly 'undo' the removal by returning that person to the team.

Due to the above reasons, it was necessary to validate the results of the analysis before actual reductions in team size could be implemented. To this end, we used the following validation methodology:

- While creating a schedule for a future month, we created two schedules. The first (the actual schedule) had the current number of people in the team (i.e. before the proposed reduction) and the second (optimised schedule) had the reduced number of people recommended by the strategic planning analysis. When we created the optimised schedule, factors such as vacations and training were also taken into account.
- The scheduling of the team was carried out using the actual schedule.
- At the end of the month for which the optimised schedule was created, the actual load for that month was run against the optimised schedule. The goal of this was to check whether the optimised schedule would still maintain the service-level constraint.

The above-mentioned validation methodology was run for several months. For each of these months, the above-mentioned methodology indicated that the optimised schedule (with the reduced team size) would have satisfied the service-level constraint. This validated the potential for a significant decrease in team size and corresponding reduction in personnel costs.

7 Summary and future work

The unique characteristics involved in providing third-level support work, as well as the increasing need to schedule shifts for such work, require new and specific forecasting, staffing and scheduling models and methodologies. This paper presents one such specific methodology, and describes the special models and techniques that we developed. This work also describes the application of this methodology to a specific team and demonstrates significant potential benefits in terms of manpower cost savings.

The current staffing model used in this methodology is based on simulation. Future work includes enhancing analytical queuing models, perhaps based on models from other domains such as healthcare (Green et al., 2005) that will enable us to model and analyse this type of work. We are in the midst of such work, which includes, among other things, addressing the routing of the problem ticket to various parties and taking into account the subexponential nature of the service times. We are also working on improved forecasting models that would enable such analytical analysis. For example, recent analysis carried out by us on the load data has shown that some form of a non-homogeneous Poisson model may indeed be appropriate for describing ticket arrivals. Additional future work includes estimating the exact ticket handling workflow based on available data and taking into account the heterogeneity of agents, which may be especially relevant for small teams.

References

Avramidis, A.N., Deslauriers, A. and L'Ecuyer, P. (2004) 'Modeling daily arrivals to a telephone call center', *Management Science*, Vol. 50, pp.896–908.

Avramidis, A.N., Gendreau, M., L'Ecuyer, P. and Pisacane, O. (2007) 'Simulation-based optimization of agent scheduling in multiskill call centers', *5th Annual International Industrial Simulation Conference (ISC-2007)*,11–13 Jun 2007, Delft, The Netherlands.

Brockwell, P.J. and Davis, R.A. (2002) *Introduction to Time Series and Forecasting*, Springer.

Brusco, M.J. and Jacobs, L.W. (2000) 'Optimal models for meal-break and start-time flexibility in continuous tour scheduling', *Management Science*, Vol. 46, No. 12, pp.1630–1641.

Gans, N., Koole, G. and Mandelbaum, A. (2003) 'Telephone call centers: tutorial, review and research prospects', *Manufacturing and Service Operations Management (M&SOM)*, Vol. 5, No. 2, pp.79–141.

Goldie, C. and Kluppelberg, C., (1997) *Subexponential Distributions, A Practical Guide to Heavy Tails: Statistical Techniques for Analysing Heavy Tails*, Birkhauser, Basel.

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 (POMS)*, Vol. 16, No. 1, pp.13–39.

Green, L.V., Soares, J., Giglio, J. and Green, R. (2005) *Using Queuing Theory to Increase the Effectiveness of Physician Staffing in the Emergency Department*, Working Paper.

Gross, D. and Harris, C.M. (1998) *Fundamentals of Queuing Theory*, Wiley-Interscience.

Mason, A.J., Ryan, D.M. and Panton, D.M. (1998) 'Integrated simulation, heuristic and optimization approaches to staff scheduling', *Operations Research*, Vol. 46, No. 2, pp.161–175.

Mandelbaum, A. and Zeltyn, S. (2007) 'Service engineering in action: the Palm/Erlang-A queue, with applications to call centers', *Advances in Service Innovation*, Springer-Verlag, pp.17–48.

Russel, S. and Norvig, P. (2003) *Artificial Intelligence: A Modern Approach*, Prentice Hall.