OmniaScience

Journal of Industrial Engineering and Management

JIEM, 2012 – 5(2): 285-296 – Online ISSN: 2013-0953 – Print ISSN: 2013-8423

http://dx.doi.org/10.3926/jiem.219

LETRIS: staffing service systems by means of simulation

Albert Corominas， Amaia Lusa

Universitat politècnica de catalunya (Spain)

albert.corominas@upc.edu， amaia.lusa@upc.edu

Received: June 2010

Accepted: November 2012

Abstract

Purpose: This paper introduces a procedure for solving the staffing problem in a service system (i.e., determining the number of servers for each staffing period).

Design/methodology: The proposed algorithm combines the use of queueing theory to find an initial solution with the use of simulation to adjust the number of servers to meet previously specified target non-delay probabilities. The basic idea of the simulation phase of the procedure is to successively fix the number of servers from the first staffing period to the last, without backtracking.

Findings: Under the assumptions that the number of servers is not upper-bounded and there are no abandonments and, therefore, no retrials, the procedure converges in a finite number of iterations, regardless of the distributions of arrivals and services, and requires a reasonable amount of computing time.

Originality / value: The new procedure proposed in this paper is a systematic, robust way to find a good solution to a relevant problem in the field of service management and it is very easy to implement using no more than commonly accessible tools.

Keywords: staffing, time-varying demand, queueuing systems, simulation

1. Introduction

One important problem that arises in the management of a service system with time-varying demand is how to schedule the staff to reach a particular level of service quality at a minimum cost. This problem has been widely addressed in the literature since 1954, when Edie published a pioneering paper about toll-booths.

Quality of service is usually assessed in terms of service level (i.e., the probability that the service to a unit will begin no later than a given number of time units after the arrival of that unit to the system) or probability of delay (i.e., the probability of a unit's waiting time being greater than 0). Probability of delay is the measure used in this paper.

Since this problem is very difficult, it is usually addressed by means of a hierarchical procedure consisting of two phases: staffing and scheduling.

The staffing problem consists of determining, for each daily forecasted demand profile, the number of servers that need to be working during each staffing interval over the course of the day in order to reach a given service quality level. Staffing intervals are defined as the moments at which changes in staffing are allowed (e.g., every half-hour).

Therefore, assuming that the day is divided into T staffing intervals, the staffing problem consists in determining the values of s_t (t=1,...,T), where s_t is the number of people on duty (servers) at the staffing interval t, with the objective of minimizing the total number of working hours required while ensuring a specified service quality level.

Once the staffing problem has been solved, scheduling decisions (that is, how many workers are assigned to each pattern of working time) are made, often by means of a mixed-integer linear program (MILP) (Dantzig, 1954), which considers as input the values of s_t obtained during the staffing phase. One example of this is the Kleen City problem, an exercise in linear programming modelling presented in Wagner (1975).

This hierarchical approach does not guarantee an optimal solution; nevertheless, it is usually the only reasonable way to deal with the problem.

The evaluation of a given solution of the staffing problem may be done by means of queue theory (Green, Kolesar & Whitt, 2007; Ingolfsson et al., 2007; Stolletz, 2008) or simulation. However, the approximation errors given by methods based on queue theory may be fairly large (Stolletz, 2008) unless rather restrictive assumptions are fulfilled.

Anyway, even if an exact evaluating algorithm is available, a procedure is needed to find a solution that achieves the required performance with minimum requirements of staff. This is the purpose of the simulation-based approach, named LETRIS, that is introduced in this paper.

2. State of the art

The paper by Green et al. (2007) exhaustively describes the state of the art of the staffing problem. Therefore, we refer the reader to that paper and limit ourselves to reviewing some essential ideas and additional contributions.

When demand is stationary and the scheduling horizon is long enough for the transient phase to be considered negligible, queueing theory can be used to determine the minimum number of people (servers) needed to guarantee the required service quality level, provided that there is a suitable queueing theory model for the particular distribution of arrival and service times.

However, when the distribution of the demand varies over time, queueing theory is not applicable straightforwardly, since there is no stationary state for the system.

This notwithstanding, the use of "stationary models in a nonstationary manner" (Green et al., 2007) makes it possible to use queueing theory in many circumstances. To do this, we must assume that the model $M_t/G/s_t+G/$ is suitable (i.e., arrivals follow a non-homogeneous Poisson process, M_t, with a time-varying arrival rate; time services are independently and identically distributed (IID) according to a general distribution, $G/$; the number of servers, s_t, is time-varying; and customers leave the system after waiting for IID times that follow a general distribution, $G/$). Under these assumptions, pointwise stationary approximation (PSA) (Green & Kolesar, 1991) yields good results when the standard of quality is high and service times and staffing intervals are short. (This approach applies the stationary model at each moment of time, using the values of the parameters corresponding to that time.) PSA provides the number of servers for each moment of time, thus ignoring staffing intervals; segmented PSA assigns to each staffing interval the maximum number of servers computed for the time moments belonging to that interval.

The stationary independent period-by-period (SIPP) method (Green, Kolesar & Soares, 2001) has a name that is very descriptive of its main idea, which is the same idea that lies behind PSA, i.e., the consideration of independent periods. A stationary model is applied to each period (i.e., staffing interval) using the average arrival rate for each interval. (SIPP Max, a refinement of SIPP proposed in Green et al., 2001, instead uses the maximum arrival rate within each interval.)

Each customer remains in the system for the length of the service time. When the service time is long, there is a significant lag between the arrival-rate peak and the customer-delay peak, with the former preceding the latter. Therefore, methods like PSA and SIPP can be improved by shifting the arrival rate to the right by an amount corresponding to the mean service time before applying the stationary queueing models (see Green et al., 2001; Green, Kolesar & Soares, 2003, for the lagged version of SIPP).

Green et al. (2007) also discuss other assumptions under which queueing theory provides good solutions.

As indicated in Green et al. (2007), SIPP may be applied even when there is no suitable queueing theory model for the system by using simulation as a replacement for the queueing model. However, the two main assumptions behind PSA and SIPP (i.e., that the intervals are independent and the behaviour of the system in each interval is that of the stationary state) obviously do not hold. Therefore, these methods only provide estimates and simulation is needed in order to validate and refine the solution when the estimate is not good enough.

Stolletz (2008), to approximate the behavior of non-stationary *M(t)/M(t)/c(t)*-queues, proposes SBC (stationary backlog-carryover), which is able to deal with temporarily overloaded systems, and MAR (modified arrival rate), which performs better than lagged SIPP approach.

Ingolfsson et al. (2007) compare, in terms of accuracy and computing time, the performance of seven methods in computing or approximating service levels for non-stationary *M(t)/M/s(t)* queueing systems with exhaustive discipline.

Simulation, regardless of the characteristics of the studied system, may evaluate solutions with any required accuracy. However, simulation by itself is not a suitable tool for solving the staffing problem, since the number of solutions to be compared is usually huge (Green et al., 2007). It is essential, however, for evaluating a given solution and determining how to modify it in order to reduce the cost or reach the prescribed quality level. One approach that is generally appropriate is to use a queueing theory model to obtain an initial solution and to modify it afterwards by means of simulation.

The use of simulation in setting staffing levels has been described in several papers. Kwan, Davis and Greenwood (1988) use a simulation model to check the solution given by a *M*/*M*/*s* model; when the simulated server utilization for a particular period is greater than a specified threshold value, the number of servers for that period is increased by one and the simulation is repeated; the authors point out that queueing theory underestimates the utilization of the servers during a light traffic period when it is preceded by a heavy traffic period. Brigandi, Dargon, Sheenan and Spencer (1994) describe a simulation tool conceived for designing and evaluating inbound call centers and synthesize some case studies.

The above-mentioned papers describe the simulation model and how it can be used to evaluate a solution, but they give little or no indication about a systematic procedure for modifying the initial solution in order to obtain a satisfactory one. This is probably because the initial and final solutions are assumed to be very similar. Of course, increases in calculation

speed have made it possible to conceive and apply methods that are very demanding in terms of computer power and which just two decades ago would have been impractical.

Corominas, Lusa and Muñoz (2005) and Muñoz (2007) describe an iterative method for solving the staffing problem. This method starts from an initial solution given by a queueing theory model. At each iteration, the performance of the system is simulated and the results of the simulation are used to revise the values of s_t (t=1,...,T). A computational experience shows that the method requires a number of iterations comparable to T in order to converge. Nevertheless, the convergence of the method cannot be demonstrated in all cases.

Feldman, Mandelbaum, Massey and Whitt (2008) introduce the iterative-staffing algorithm (ISA). The general idea of ISA is to start with a number of servers that is large enough to assure that the probability of delay, $\forall t$, is negligible. At each iteration, a large number of simulations (e.g.: 5,000) is performed in order to estimate Q_{nt} (the distribution of the number of customers in the system in the period t, when the solution is that which corresponds to the iteration n of ISA). The number of servers for the next iteration is determined such that $P\left(Q_{nt} \geq s_t^{n+1}\right) \leq \pi < P\left(Q_{nt} \geq s_t^{n+1} - 1\right)$, where π is the target delay probability. The algorithm stops when, between two consecutive iterations, $\left|s_t^n - s_t^{n+1}\right| \leq 1 \ \forall t$. The convergence of ISA is reported for the model $M_t/M/s_t+M$.

3. LETRIS: a simulation-based approach to solving the staffing problem

Left-to-right simulation (LETRIS) is a simulation-based procedure for solving the staffing problem, assuming that there are no abandonments and, therefore, no retrials, that converges to a good solution (which is quite likely to be optimum in many cases) regardless of the specific assumptions about arrivals and service. Given a scheduling horizon consisting of T staffing intervals and the profile of the stochastic demand within the horizon, the method is devised to determine the staffing levels s_t (t=1,...,T), where s_t are the positive integers $\forall t$, perhaps bounded from above, in such a way that the probability of delay corresponding to any staffing period, p_t, $\forall t$, is no greater than a previously specified threshold, π_t, $\forall t$, and $\sum_{t=1}^{T} \tau_t \cdot s_t$ is minimized, where τ_t is the length of the staffing interval t.

Thus, probability of delay is the quality measure adopted in LETRIS. Among other possible measures (e.g., average waiting time or probability of abandonment), the two measures that appear in the literature on the staffing problem, as indicated in Section 1, are as follows: 1) service level (Green et al., 2007), that is, the probability of not having to wait for more than a given time, y (Testik, Cochran & Runger, 2004); and 2) probability of delay. In fact, all reasonable measures of performance are positively correlated: any increase in the number of

servers has a positive impact on all of them. Green et al. (2007) point out that the probability of delay can be seen as a particular case of the service level, considering y=0, which is generally easier to compute and tends to be a relatively robust performance measure.

However, the most important characteristic of probability of delay as a quality criterion in LETRIS is the fact that it can be computed in any period t regardless of the behaviour of the system during the periods from t+1 to T. Instead, to compute service level for the units arriving during period t, we have to know the behaviour of the system during some subsequent periods

The general idea of LETRIS is to successively fix the values of s_t from t+1 to T, without backtracking. That is, firstly, the value of s_1 (the minimum number of servers such that the estimation of p_1, $\hat{p}_1$, is no greater than π_1), is calculated (note that the value of s_1 depends only on the behaviour of the system within the period t=1.) Next, s_2 is calculated, and so forth until s_T is found.

$SIMUL\left(t, s, N, \hat{p}_t\right)$ process:

Input :

- $t$staffing period
- s number of servers in the staffing period
- N number of runs in the simulation model
- State of the system, for each of the N simulation runs, corresponding to the end of the staffing period t-1 when the number of servers during this period is s_{t-1} (previously determined). These N final states will be the initial states for the N simulation runs corresponding to staffing period t. For t=1, according to the initial conditions of the real system, the N initial states may be the same for the N runs (for instance, the system may always be empty at the beginning of the scheduling horizon) or they may have to be drawn from a given distribution probability. To define the state of the system, at most s_{t-1}+1 values are needed: one for each server (an indication that the server is idle—for instance, a negative value—or the time elapsed since the server began serving the customer present at the end of the staffing interval t-1, although this time is not necessary when the distribution of service time is memoryless, i.e., exponential); plus the number of customers in the queue (assumed to be unique).

Process:

Simulate N times the behaviour of the system during the staffing period t, using as the initial state at each run, n, the final state given by the simulation run n of the previous period (t -1).

Output:

- $\hat{p}_t$, estimation of the probability of delay.
- State of the system, at the end of staffing period t, for each of the N simulation runs.

Figure 1. SIMUL process

A key feature of the procedure with regard to computational burden is that, at each period t, the configuration and the simulated behaviour of the system in past periods are given, and we need only simulate the behaviour of the system within the period t, regardless of future periods, in order to set the value of s_t.

Figure 1 describes the *SIMUL* process, which is a crucial component of LETRIS. Figure 2 outlines the LETRIS procedure.

- Define:
  - The scheduling horizon.
  - The stochastic process of arrivals.
  - The distribution of service times.
  - The *T* staffing intervals.
  - The upper bounds, S_t, on the number of servers $\left(0 < s_t \leq S_t; t = 1,...,T\right)$.
  - The admissible probabilities of delay, π_t, $\forall t$.
  - The initial state of the system (or the set of feasible initial states, with their respective probabilities).
  - *N*.
- Choose and apply the most suitable queueing theory model to calculate the initial values, s_t^0, of s_t $\forall t$.
- Simulate *N* times the arrivals throughout the scheduling horizon and associate the corresponding service time with each customer.
- For *t*=1 to *T*:
  - Perform $SIMUL\left(t, s_t, N, \hat{p}_t\right)$
  - If $\hat{p}_t < \pi_t$, until ($\hat{p}_t \geq \pi_t$ or s_t=1), repeat:

    $s_t = s_t - 1$; $SIMUL\left(t, s_t, N, \hat{p}_t\right)$

    If $\hat{p}_t > \pi_t$, $s_t = s_t + 1$
  - If $\hat{p}_t > \pi_t$, until ($\hat{p}_t \leq \pi_t$ or s_t=*S*), repeat:

    $s_t = s_t + 1$; $SIMUL\left(t, s_t, N, \hat{p}_t\right)$

Figure 2. LETRIS procedure

The procedure tends to minimize the total number of required working hours. However, it does not guarantee an optimal solution for the broader scheduling problem if servers cannot be scheduled on an interval-by-interval basis (which is the operationally normal situation). Moreover, although, given s_τ $\left(\tau = 1,...,t-1\right)$, s_t is optimal for the period *t*, we cannot preclude the possibility that an increase η in s_τ may allow a reduction greater than η in some values of s_t ($t > \tau$), or, if the durations of the staffing intervals are different, the possibility that the reduction multiplied by the duration of the corresponding staffing interval may be greater than η multiplied by the duration of the staffing interval τ.

The confidence interval associated with the estimator $\hat{p}_t$ at level $1-\alpha$ is equal to $\hat{p}_t \pm t_\alpha \cdot \sqrt{\frac{\hat{p}_t \cdot \left(1-\hat{p}_t\right)}{N}}$, where, for high enough values of *N*, t_α is the corresponding value of the standard normal distribution. Given that the worst case is $\hat{p}_t = 0.5$, it follows, if we impose

$t_\alpha \cdot \sqrt{\frac{\hat{p}_t \cdot (1-\hat{p}_t)}{N}} \leq \varepsilon$, that $N \geq \frac{t_\alpha^2}{4 \cdot \varepsilon^2}$ (of course, in most applications the required value of N will be smaller, since the prescribed value of π_t will be much smaller than 0.5).

The convergence of the procedure is guaranteed, provided that the upper bounds, S_t, are high enough to achieve the required delay probability at every iteration (i.e., for every value of t), since $\hat{p}_t$ monotonically decreases as s_t increases, given that the arrival and service times do not depend on s_t, due to the way in which the simulations are performed (i.e., N streams of arrivals and, for every stream, the service time corresponding to each customer are generated and stored at the beginning of the procedure and then used for all the simulation runs). Let κ_t be an integer equal to the absolute value of the difference between s_t^0 and s_t (i.e., the difference between the initial values provided by the queueing model and the final values that LETRIS yields). If the estimation of p_t were to coincide with the true value, the number of iterations required to converge at each staffing period would be equal to $\kappa_t + 1$ (therefore, the volume of computation would be proportional to $N \cdot \sum_{t=1}^{T} (\kappa_t + 1)$).

4. A computational experiment

Of course, the computing time required to apply LETRIS, for given hardware and software, depends on the characteristics of the system to be staffed. Therefore, we devised an experiment intended only to illustrate the behaviour of the procedure for a specific set of 36 scenarios.

The following assumptions define the scenarios:

- Scheduling horizon: 8 hours.
- Staffing intervals with equal length of 15 minutes each. Therefore, T=32.
- Target non-delay probability: $\pi_t = 0.1 \ \forall t$.
- Nonstationary Poisson arrivals, with the arrival rate following a uniform distribution $U\left[(1-r)\cdot\lambda'(\tau), (1+r)\cdot\lambda'(\tau)\right]$, where:
- $\lambda'(\tau) = \bar{\lambda}\cdot(1 + A\cdot\sin(2\pi\tau/8))$, $\bar{\lambda} = 30\,h^{-1}$, A takes three values $(0.1, 0.5, 1.0)$ and r follows a uniform distribution $U[0, R]$, where R takes the three values 0.05, 0.15 and 0.25. Combining the three values of A and the three values of r yields nine different patterns of arrivals.

- Service times such that $\mu = 12\,h^{-1}$, following four different distributions: exponential, $U\left[0, 2\cdot\mu^{-1}\right]$, $U\left[\left(2-\sqrt{3}\right)\mu^{-1}, \sqrt{3}\mu^{-1}\right]$ and *D* (therefore, with standard deviations equal to $\mu^{-1}, \frac{1}{\sqrt{3}}\cdot\mu^{-1}, \left(1-\frac{1}{\sqrt{3}}\right)\cdot\mu^{-1}, 0$, respectively). Combining the nine patterns of arrivals with the four service time distributions yields 36 scenarios.
- There is no customer abandonment.
- There is no upper bound on the number of servers.

When $s_t < s_{t-1}$, it is assumed that, before leaving the system, the extra servers finish the services that are active at the final instant of the *t*-1 staffing interval.

The values s_t^0 were computed by applying lagged SIPP to the M/M/s model.

The calculations were performed on a 3.2 GHz Pentium IV PC with 1.5 GB of RAM using a program coded in Java 2 SDK v. 1.4.2.

The parameter *N* (i.e., the number of run simulations) was set to 10,000.

Table 1 shows the average computing times (excluding those of the queuing model) for the nine possible values of the pair (*A*, *r*). These times include the time needed to generate the arrivals and service times and to calculate the s_t values.

$A\downarrow$ $r\rightarrow$	0.05	0.15	0.25
0.1	827.29	827.9	857.2
0.5	1016.1	1037.9	1045.4
1.0	1358.4	1357	1713.3

Table 1. Computing times (in seconds) for the nine combinations of the *A* and *r* values

The computing times needed to generate and store the arrival times range from 7 seconds $(A = 0.1,\ r = 0.05)$ to 54 seconds $(A = 1.0,\ r = 0.25)$. The times needed to generate the service times are obviously very similar for the various combinations of *A* and *r*. However, the greater the variance of the arrival rate, which depends mainly on the value of *A*, the greater the time required to calculate the s_t values for a specific distribution of service time (from about 800 seconds for *A*=0.1 and *r*=0.05 to about 1,350 seconds for *A*=1.0 and *r*=0.25). These times do not depend very greatly on the service time distribution; however, they tend to be smaller than average for the exponential distribution and larger than average for the *D* distribution.

The discrepancies, κ_t, between the initial and final values of s_t are never negative or greater than 1, except in the first staffing period, when they can be as high as 3 or 4 (SIPP tends to overstaff the first period when the system is supposed to be initially empty).

Table 2 shows the average values for the total relative discrepancy, in percentage terms (i.e., $100 \cdot \sum_t \kappa_t / \sum_t s_t^0$), for the nine possible values of the pair (*A*, *r*). These values tend to be smaller than average for the exponential distribution and larger than average for the *D* distribution.

$A \downarrow$ $r \rightarrow$	0.05	0.15	0.25
0.1	3.31%	3.21%	3.21%
0.5	2.23%	2.59%	2.38%
1.0	2.33%	2.39%	2.18%

Table 2. Relative discrepancies for the nine combinations of the *A* and *r* values

5. Conclusions

This paper introduces a procedure, called LETRIS, for solving the staffing problem, when there are no abandonments and, therefore, no retrials. LETRIS combines the use of queuing theory to find an initial solution with the use of simulation to adjust the number of servers for every staffing interval in order to meet previously specified target non-delay probabilities.

The basic idea of the simulation phase of the procedure is to successively fix the number of servers from the first staffing period to the last, without backtracking.

The procedure finishes in a finite number of iterations, provided that the upper bounds on the number of servers are high enough to achieve the required delay probability at every iteration (i.e., for every value of *t*), and its application requires a reasonable amount of computing time.

Future research may deal with the extending LETRIS to consider abandonments and retrials and the case when the upper bounds on the number of servers are not high enough.

Acknowledgments

Supported by the Spanish Ministry of Economy and Competitivity (project DPI2010-15614). The authors are grateful to Alberto García-Villoria for his help in performing the computational experiment.

References

Brigandi, A., Dargon, D., Sheenan, M., Spencer III, T. (1994). AT&T's call processing simulator (CAPS): operational design for inbound call centres. *Interfaces,* 24, 6-28. http://dx.doi.org/10.1287/inte.24.1.6

Corominas, A., Lusa, A., Muñoz, N. (2005). Cálculo de la capacidad necesaria para obtener un nivel de servicio predeterminado. *Proceedings of the IX Congreso de Ingeniería de Organización*, Gijón.

Dantzig, G.B. (1954). A comment on Edie's "Traffic delays at toll booths. *J. Opl. Res. Soc. of America.* 2, 339-341.

Edie, L.C (1954). Traffic delays at toll booths. *J. Opl. Res. Soc. of America.* 2, 107-138.

Feldman, Z., Mandelbaum, A., Massey, W.A., Whitt, W. (2008). Staffing of time-varying queues to achieve time-stable performance. *Mngt. Sci.,* 54, 324-338. http://dx.doi.org/10.1287/mnsc.1070.0821

Green, L.V., Kolesar, P.J. (1991). The pointwise stationary approximation for queues with nonstationary arrivals. *Mngt. Sci.,* 37, 84-97. http://dx.doi.org/10.1287/mnsc.37.1.84

Green, L.V., Kolesar, P.J., Soares, J. (2001). Improving the SIPP approach for staffing service systems that have cyclic demands. *Opns. Res.,* 49, 549-564. http://dx.doi.org/10.1287/opre.49.4.549.11228

Green, L.V., Kolesar, P.J., Soares, J. (2003). An improved heuristic for staffing telephone call centres with limited operating hours. *Prod. and Opns. Mngt.,* 12, 46-61. http://dx.doi.org/10.1111/j.1937-5956.2003.tb00197.x

Green, L.V., Kolesar, P.J., Whitt, W. (2007). Coping with time-varying demand when setting staffing requirements for a service system. *Prod. and Opns. Mngt.,* 16, 13-39. http://dx.doi.org/10.1111/j.1937-5956.2007.tb00164.x

Ingolfsson, A., Akhmetshina, E., Budge, S., Li, Y., Wu, X. (2007). A survey and experimental comparison of service level approximation methods for non-stationary *M(t)/M/s(t)* queueing systems with exhaustive discipline. *INFORMS J. of Computing,* 19(2), 201-214. http://dx.doi.org/10.1287/ijoc.1050.0157

Kwan, S.K., Davis, M.M., Greenwood, A.G. (1988). A simulation model for determining variable worker requirements in a service operation with time-dependent customer demand. *Queueing Systems,* 3, 265-276. http://dx.doi.org/10.1007/BF01161218

Muñoz, N. (2007). *Consideración de aspectos aleatorios en la planificación del tiempo de trabajo con jornada anualizada*. PhD thesis, Universitat Politècnica de Catalunya.

Stolletz, R. (2008). Approximation of the non-stationary *M(t)/M(t)/c(t)*-queue using stationary queueing models.: The stationary backlog-carryover approach. *European J. of Opl. Res.*, 190(2), 478-493. http://dx.doi.org/10.1016/j.ejor.2007.06.036

Testik, M.C., Cochran, J.K., Runger, G.C. (2004). Adaptive server staffing in the presence of time-varying arrivals: a feed-forward control approach. *J. Opl. Res. Soc.,* 55, 233-239. http://dx.doi.org/10.1057/palgrave.jors.2601677

Wagner, H.M. (1975). *Principles of Operations Research, 2nd ed.* Prentice-Hall.