

Applications of Operations Research to Customer Service Representative Assignment: A Literature-Based Analytical Study of Twenty-Five Published Models

Author Details:

Author Names- **Karanbir Singh Gill, Jash Shah, Kabir Bahl, Ishaan Mathran, Jia Rathod.**

SVKM's Narsee Monjee Institute of Management Studies Anil Surendra Modi School of Commerce, Mumbai, India



<https://doi.org/10.55041/ijssmt.v2i10.012>

Cite this Article: Gill, K. S., Shah, J., Bahl, K., Mathran, I. & Rathod, J. (2026). Applications of Operations Research to Customer Service Representative Assignment: A Literature-Based Analytical Study of Twenty-Five Published Models. *International Journal of Science, Strategic Management and Technology*, 02(10), 1-9. <https://doi.org/10.55041/ijssmt.v2i10.012>

License:  This article is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting use, distribution, and reproduction in any medium, provided the original author(s) and source are properly credited.

Abstract—

Customer contact centres have a heavy reliance on customer service representatives (CSRs) and labour is often the largest single cost in the operation. How many CSRs to employ, how to schedule their working hours, the skills they should have, and how to route interactions between customers and CSRs, are a series of assignment problems that Operations Research (OR) has been studying for five decades. This review analyses twenty-five peer-reviewed papers, published between 1974 and 2017, that apply OR to CSR assignment. The papers were identified in major scholarly databases, screened against clearly defined inclusion criteria and grouped into five streams: queueing-based staffing, shift scheduling and multi-skill staffing, simulation-based optimisation, planning under forecast uncertainty, and dynamic call-to-agent routing. For each, the real-world problem, the role of OR, the model, the solution method and software, and validation are extracted, and strengths, limitations and extensions are identified through critical review.

A worked example applying the Hungarian method and Erlang-C staffing to a representative case shows a reduction in total handling time of about 27% relative to a naïve assignment and identifies the minimum staffing needed to meet an 80/20 service level. Synthesis shows that the trend is moving from deterministic assignment and covering models to stochastic, simulation-based, and robust formulations that include staffing, scheduling, and routing. Various research gaps are explored, including omnichannel service, human-AI collaboration, learning-based routing, and CSR fairness and well-being.

Keywords— Assignment problem; Call centre staffing; Customer service representatives; Shift scheduling; Skill-based routing; Operations research.

1. INTRODUCTION

Operations Research is the study of applying mathematical models, statistics and algorithms to solve problems in decision-making. The traditional set of tools

in operations research (linear programming (LP), transportation and assignment models, integer programming (IP), queueing theory, network models, simulation and heuristics) has been widely used in manufacturing, logistics, medicine, and public services, and one of the most commercially valuable domains is in the customer contact centre (CC). Customers contact a CC by either phone, email or chat with an organisation and are supported by Customer Service Representatives (CSR) [1], [2].

One frequently asked question to a contact centre is simple: **"what should each CSR do and when?"** This simple question hides several linked decisions. Managers are asked to (a) forecast demand (b) determine *staffing* levels (how many CSRs in each period to achieve a specific level of service), (c) develop shifts to satisfy staffing levels (*scheduling*), (d) develop skill sets for each CSR (*skill assignment*), and (e) in real time, determine which waiting customer is routed to which available CSR (*routing*). Each of these decisions is a (discrete) assignment problem: the optimal matching of limited human resources to tasks that minimises cost or maximises service.

The textbook assignment problem (as solved by the Hungarian method [3]) sequentially assigns n workers to n jobs one-to-one at minimum cost. The assignment of CSRs in reality, however, differs greatly: work flows in randomly, requiring near-immediate attention; demand varies from hour to hour and can be met only through shifts; CSR skill, work-rate, and quality vary; and forecasts are inherently uncertain. Most surveys treat call-centres operations [1], [2] or personnel scheduling in general [4], [5], [6], but few examine the development of, model by model, the evolution of the OR formulations for the assignment problem and their relative merits in terms of goals, constraints, assumptions, and solution procedures.

This article addresses that gap. Its objectives are to: (1) identify published papers applying OR to CSR assignment; (2) describe the real-life context of each; (3) present the mathematical formulation; (4) describe the solution

approach, tools and validation; (5) critically appraise the findings, strengths, limitations and extensions of each model. Section II reviews the OR background, Section III explains the methodology, Section IV presents and discusses the results, including a worked numerical example, and Section V concludes.

II. LITERATURE REVIEW

A. The Classical Assignment Problem

Let $i = 1, \dots, n$ index CSRs and $j = 1, \dots, n$ index tasks such as queues or customer segments. Let c_{ij} be the cost or time when CSR i performs task j and $x_{ij} = 1$ if CSR i is assigned to task j . The model is:

$$\begin{aligned} \text{Min } Z &= \sum_i \sum_j c_{ij} x_{ij} \\ \text{s.t. } \sum_j x_{ij} &= 1 \quad \forall i \quad (\text{one task per CSR}) \\ \sum_i x_{ij} &= 1 \quad \forall j \quad (\text{one CSR per task}) \\ x_{ij} &\in \{0, 1\} \end{aligned}$$

The constraint matrix is totally unimodular, so the LP relaxation is integral and the Hungarian method solves the problem in polynomial time by reducing rows and columns until n independent zeros appear [3]. Unbalanced problems use dummy rows or columns, and maximisation problems are converted by subtracting each entry from the largest value.

B. Queueing View of CSR Assignment

In a contact centre a task is a random stream of contacts, so the number of CSRs needed depends nonlinearly on volume and handling time. With arrival rate λ , mean handling time $1/\mu$ and offered load $R = \lambda/\mu$, the Erlang-C model gives the delay probability and the service level for target time τ [1]:

$$\begin{aligned} C(N, R) &= [R^N N! / N(N-R)] / [\sum_{k=0}^{N-1} R^k / k! + R^N N! / N(N-R)] \\ SL(N) &= 1 - C(N, R) \cdot \exp(-(N-R)\mu\tau) \end{aligned}$$

Heavy-traffic analysis yields the square-root staffing rule $N \approx R + \beta\sqrt{R}$ [7], [8]. Once requirements N_t per interval are known, shift scheduling is a set-covering IP [9]: choose x_s CSRs on each shift s to minimise cost subject to $\sum_s a_{ts} x_s \geq N_t$. With multi-skilled CSRs, routing becomes a dynamic assignment of calls to agents.

C. Taxonomy of Decisions

Table I summarises the decision hierarchy that structures this review.

Table I: Hierarchy of CSR assignment decisions

Level	Decision	Typical OR model
Strategic	Hiring, cross-training, outsourcing	Queueing design, MDP, IP

Level	Decision	Typical OR model
Tactical	Staffing per interval	Erlang C/A, square-root rule, simulation
Tactical	Shift/tour scheduling	Set-covering IP, network flow, stochastic IP
Operational	CSR-to-queue assignment	Assignment problem, LP
Real time	Contact-to-CSR routing	Priority/index policies, MDP, simulation

D. Prior Reviews and Research Gap

Gans et al. [1] and Aksin et al. [2] surveyed call-centre operations broadly; Ernst et al. [4] and Van den Bergh et al. [5] reviewed personnel scheduling across industries, and Defraeye and Van Nieuwenhuyse [6] focused on staffing under nonstationary demand. Although these surveys classify the field, they do not compare the objectives, constraints and assumptions of individual models side by side or illustrate them numerically. The present study addresses this gap by comparing twenty-five articles model by model and adding a worked example.

III. METHODOLOGY

A. Search Strategy

Articles were identified in ScienceDirect, INFORMS PubsOnline, SpringerLink, Wiley Online Library and IDEAS/RePEc using combinations of *call centre*, *contact centre*, *customer service representative*, *agent*, *staffing*, *scheduling*, *assignment*, *skill-based routing* and *multiskill*. Backward citation tracking from the surveys [1], [2], [6] located further studies.

B. Inclusion Criteria

Articles were included if they (1) appeared in a peer-reviewed journal; (2) formulated an explicit OR model for assigning, staffing, scheduling or routing CSRs; (3) were motivated by real contact-centre operations; and (4) described the solution method and its evaluation. Twenty-five articles were selected to cover every level of Table I, a wide range of OR techniques and the development of the field over time (Table II).

Table II: Articles analysed in this study

ID	Study	Str.	Core technique
P1	Segal (1974) [10]	A	Network-flow formulation solved by a network-flow algorithm
P2	Thompson (1997) [11]	A	Piecewise-linear approximation of queueing service curves inside an IP
P3	Gans, Koole & Mandelbaum (2003) [1]	A	Erlang C, Erlang A, square-root staffing, set-covering scheduling, routing models

ID	Study	Str.	Core technique
P4	Borst, Mandelbaum & Reiman (2004) [8]	A	Asymptotic
P5	Green, Kolesar & Whitt (2007) [12]	A	Comparison of SIPP, lagged SIPP, modified-offered-load
P6	Mandelbaum & Zeltyn (2009) [13]	A	Asymptotic analysis in efficiency-driven, QED and quality-driven regimes
P7	Fukunaga et al. (2002) [14]	B	Constraint-based local search with fast simulation-based schedule evaluation
P8	Shumsky (2004) [15]	B	Markov-chain modelling with tractable approximations
P9	Wallace & Whitt (2005) [16]	B	Erlang-based initial staffing + skill assignment heuristic + iterative simulation refinement
P10	Bhulai, Koole & Pot (2008) [17]	B	Approximate staffing per interval followed by an LP/IP shift-covering model with skill substitution
P11	Dietz (2011) [18]	B	MIP formulation with break placement and work-rule constraints
P12	Atlason, Epelman & Henderson (2004) [19]	C	Kelley-type cutting-plane method with subgradients estimated by simulation
P13	Atlason, Epelman & Henderson (2008) [20]	C	Analytic center cutting-plane method
P14	Cezik & L'Ecuyer (2008) [21]	C	IP + simulation cutting planes, rounding heuristics and local search
P15	Avramidis, Chan, Gendreau, L'Ecuyer & Pisacane (2010) [22]	C	Simulation-based cutting planes in an IP, randomised rounding and local search
P16	Ingolfsson, Campello, Wu & Cabral (2010) [23]	C	Integer programming iterated with the randomisation method
P17	Gurvich, Luedtke & Tezcan (2010) [24]	D	Chance-constrained MIP via sample-average approximation
P18	Robbins & Harrison (2010) [25]	D	Stochastic programming with sample-average approximation
P19	Excoffier, Gicquel & Jouini (2016) [26]	D	Joint chance constraints reformulated/approximated as a deterministic MILP
P20	Mattia, Rossi, Servilio & Smriglio (2017) [27]	D	Two-stage robust optimisation with budgeted uncertainty, solved by decomposition
P21	Gans & Zhou (2007) [28]	E	Queueing/Markov decision models of threshold routing policies
P22	Sisselman & Whitt (2007) [29]	E	Assignment/LP-style optimisation of agent-call-type priorities evaluated by simulation
P23	Armony & Mandelbaum (2011) [30]	E	Asymptotic analysis
P24	Mehrotra, Ross, Ryder & Zhou (2012) [31]	E	Queueing model with callbacks
P25	Chan, Koole & L'Ecuyer (2014) [32]	E	Weight-based routing

Streams: A = queueing-based staffing; B = shift scheduling and multi-skill staffing; C = simulation-based optimisation; D = forecast uncertainty; E = dynamic routing.

C. Analytical Framework

Each article was analysed with a common template: (i) real-life problem and domain; (ii) suitability of OR; (iii) mathematical model – decision variables, objective function, constraints and problem type; (iv) assumptions;

(v) solution technique, software and validation; (vi) key findings; and (vii) critical appraisal of strengths, limitations and extensions. Notation was harmonised and equations simplified to expose model structure. The numerical example in Section IV-F was solved by hand with the Hungarian method and verified in Python using *scipy.optimize.linear_sum_assignment* and complete enumeration.

IV. RESULTS AND DISCUSSION

A. Stream A – Queueing-Based Staffing Foundations

These studies established how many CSRs are needed for a given workload and how operator schedules can be optimised; they remain the basis of most workforce-management systems.

1) P1 – Segal (1974) [10]: “The operator-scheduling problem: A network-flow approach”

Domain	Telecommunications – telephone operators (the earliest form of CSR)
Problem type	Personnel scheduling; minimum-cost network flow (special LP)
Objective	Minimise weighted under- and over-staffing across the day
Technique	Network-flow formulation solved by a network (out-of-kilter type) algorithm
Tools	Mainframe network-flow code (pre-commercial-solver era)

Real-life problem. Telephone companies employed large pools of operators to assist callers. Call volume varies sharply within a day, so the number of operators on the floor in each short interval (typically 15–30 minutes) has to track the forecast requirement. Each operator works a tour with relief and meal breaks, and the placement of those breaks determines how many operators are available at each point. Planners built these schedules by hand, which was slow and produced persistent mismatches between staff and volume.

Why OR was suitable. The decision is combinatorial, with many possible break positions for each operator, linked across all periods of the day. Segal showed that treating operators as flow units travelling through time turns the problem into a network-flow problem. This guarantees integral solutions without branching, allowing exact optimisation on 1970s hardware.

Mathematical model. Let $t = 1, \dots, T$ index the periods of the day and let b_t be the required number of operators in period t (derived from the call forecast). Arcs in the network carry operators from one period to the next (working), into and out of break nodes (relief or lunch),

and in and out of the system (tour start and end). Let w_t be the flow on the working arc of period t , and s_t^+ , s_t^- the surplus and shortage relative to the requirement.

Minimise $Z = \sum_t (c^+ s_t^+ + c^- s_t^-)$

subject to $w_t - s_t^+ + s_t^- = b_t$ for all t

flow in = flow out at every node

(break and period nodes)

break-window arcs: lower $\leq$ flow $\leq$ upper (each tour takes its breaks in an allowed window)

$w_t, s_t^+, s_t^- \geq 0$ (integrality follows from total unimodularity)

Assumptions. The requirement profile b_t is known and deterministic; operators are identical (single skill); tour start times are given or modelled as fixed supply at source arcs; costs of surplus and shortage are linear.

Solution method and validation. Because the constraint matrix of a network-flow problem is totally unimodular, every basic optimal solution is integral, so a network algorithm returns a usable schedule directly. Segal used a standard minimum-cost flow procedure and showed how break rules (earliest and latest positions, minimum spacing) translate into arc bounds. Validation compared generated schedules with manual ones on operator data.

Findings. The network model produced schedules that matched requirements more closely than manual practice, and at lower cost. It showed that an apparently hard integer problem can be solved exactly by exploiting its structure, and it is one of the first published optimisation models for contact-centre workforce management.

Strengths. Exact, fast and integral by construction; transparent to planners because flows map directly to people on duty.

Limitations. The service target enters only through a fixed staff requirement, so the queueing relationship between staffing and waiting time is ignored. Single-skill, deterministic demand and linear penalties are all strong simplifications for a modern multi-skill centre.

Extensions. Replace the fixed b_t with Erlang-based requirements (see P3, P4), add multiple agent types as a multi-commodity flow, and treat b_t as uncertain (see Stream D).

2) P2 – Thompson (1997) [11]: “Labor staffing and scheduling models for controlling service levels”

Domain	Service operations with queues, including telephone call centres
Problem type	Integer programming (shift scheduling) with service-level constraints
Objective	Minimise labour cost while meeting a service level measured over the whole planning horizon
Technique	Piecewise-linear approximation of queueing service curves inside an IP; heuristic solution
Tools	Integer-programming solver; queueing (Erlang) calculations

Real-life problem. Most scheduling systems first convert a service objective (for example, answering 80% of calls within 20 seconds) into the minimum number of CSRs required in each period, and then build shifts to meet those requirements. Thompson observed that this two-step conversion forces the service target on every period, whereas managers and customers judge service over the day as a whole, which can lead to overstaffing.

Why OR was suitable. The trade-off between staffing and service is nonlinear (queueing), while shift selection is integer. Optimisation can balance cheap periods against expensive ones, which a rule of thumb cannot do.

Mathematical model. Let x_s be the number of CSRs assigned to shift s with cost c_s ; $a_{ts} = 1$ if shift s works in period t . Staff in period t is $n_t = \sum_s a_{ts} x_s$. Let λ_t be the call volume and $g_t(n)$ the fraction of calls answered within the target time when n CSRs are present (an increasing, concave function from Erlang C). The aggregate service constraint requires the call-weighted service level over the day to reach α .

Minimise $\sum_s c_s x_s$

subject to $n_t = \sum_s a_{ts} x_s$ for all t

$\sum_t \lambda_t g_t(n_t) \geq \alpha \sum_t \lambda_t$ (horizon-wide service level)

$g_t(n)$ replaced by piecewise-linear segments using binary or incremental variables

$x_s \in \mathbb{Z}_h$

Assumptions. Each period is analysed as a stationary queue (no carry-over of backlog between periods); service curves are concave so the piecewise approximation is valid; one homogeneous group of CSRs.

Solution method and validation. The paper builds several model variants (per-period, aggregate, and hybrid with a floor on per-period service) and solves them as integer programs, using heuristics where problem size makes

exact solution expensive. Performance is compared on a set of test problems that vary demand profiles and shift structures.

Findings. Controlling service at the horizon level gives similar overall service at noticeably lower labour cost than a period-by-period approach, because staff are directed to the periods where their marginal contribution is greatest. A minimum service level per period can also be imposed so that no period is served very badly, at a small extra cost.

Strengths. Directly links the scheduling objective to what customers experience; conceptually simple to adopt in existing IP-based systems.

Limitations. Ignores the dynamics between periods (queues carried over) and abandonment; single skill. Aggregate targets can hide unfairness to customers who call at unpopular times.

Extensions. Combine aggregate service constraints with simulation-based evaluation (P12–P15) and with uncertainty in λ_t (P18).

3) P3 – Gans, Koole & Mandelbaum (2003) [1]: “Telephone call centers: Tutorial, review, and research prospects”

Domain	Telephone call centres across banking, telecom, retail and public services
Problem type	Tutorial and survey: queueing, staffing, scheduling and skill-based routing
Objective	Organise the OR knowledge base for CSR planning and set a research agenda
Technique	Erlang C, Erlang A, square-root staffing, set-covering scheduling, routing models; statistical analysis of call data
Tools	Analytical models; empirical call-centre data

Real-life problem. By the year 2000, call centres employed millions of people and labour made up the largest share of their cost. Decisions about how many CSRs to hire, when they should work and which calls they should take were often based on spreadsheet versions of the Erlang C formula. The authors set out to bridge the gap between queueing theory and practice.

Why OR was suitable. Every layer of CSR planning: forecasting, staffing, scheduling, rostering and real-time routing is a recognised OR problem, and the authors show how they fit into a single hierarchy.

Mathematical model. The central building block is the M/M/N queue (Erlang C). With arrival rate λ , mean handling time $1/\mu$ and offered load $R = \lambda/\mu$, the probability that a caller waits and the service level for

target time τ are given below. The paper then extends this to Erlang A (M/M/N+M), which adds caller abandonment at rate θ , and to square-root safety staffing.

$$P(\text{Wait} > 0) = C(N, R) = \left[\frac{R^N / N! \cdot N / (N - R)}{\sum_{k=0}^{N-1} \frac{R^k}{k!} + \frac{R^N}{N!} \cdot \frac{N}{N - R}} \right] \text{ Service level: } SL(N) = 1 - C(N, R) \cdot e^{-(N - R)\mu\tau}$$

Square-root staffing: $N = R + \beta \sqrt{R}$ (β = service-grade parameter)

Shift scheduling (set covering): $\min \sum_s c_s x_s$ s.t. $\sum_s a_{ts} x_s \geq N_t, x_s \in \mathbb{Z}$

Assumptions. Poisson arrivals over a given period of time and the use of exponential or general service times (depending on the model) assume the independence of callers; this paper investigates situations where this assumption may be violated (over-dispersed arrivals and caller impatience).

Solution method and validation. A structured tutorial and literature review. The authors illustrate the models with real call-centre data, discuss forecasting of arrivals, staffing in the Quality-and-Efficiency-Driven (QED) regime, shift scheduling, and skill-based routing designs such as the 'N', 'X' and 'W' models.

Findings. Erlang C can mislead because it ignores abandonment; the square-root principle explains why large centres are efficient (economies of scale); multi-skill routing and uncertainty in arrival rates are the main open problems. The paper became the standard reference that most later studies in this review cite.

Strengths. Unifies queueing, scheduling and routing; grounded in data; provides formulas practitioners can apply.

Limitations. As a survey it does not solve a specific operational case; models for multi-skill staffing were still immature at the time.

Extensions. The open problems it identified – multi-skill staffing, uncertain arrival rates and routing with heterogeneous agents – are exactly the topics of Streams B–E.

4) P4 – Borst, Mandelbaum & Reiman (2004) [8]: “Dimensioning large call centers”

Domain	Large inbound call centres
Problem type	Queueing-based staffing optimisation (cost minimisation and constraint satisfaction)
Objective	Minimise staffing cost plus waiting (delay) cost
Technique	Asymptotic (heavy-traffic) analysis of the M/M/N queue
Tools	Analytical derivations; numerical verification

Real-life problem. Managers must decide how many CSRs to put on the phones for a given workload. Too few cause long waits and lost goodwill; too many waste wages. Exact Erlang calculations exist, but they offer little intuition, and practitioners wanted a rule that scales from tiny to enormous centres.

Why OR was suitable. This is a classic economic optimisation problem: one integer decision (N) with a convex cost structure. OR asymptotics turn it into a closed-form rule.

Mathematical model. Let c be the cost per CSR per unit time and $D(N)$ the delay cost incurred per unit time when N agents serve offered load $R = \lambda/\mu$ in an $M/M/N$ queue. The problem is to choose N to minimise total cost; alternatively, minimise N subject to a delay constraint.

Minimise $TC(N) = c \cdot N + D(N; \lambda, \mu)$ subject to $N > R, N \in \mathbb{Z}$

Asymptotically optimal staffing: $N^* = R + y^* \sqrt{R} + o(\sqrt{R})$ $y^* = \operatorname{argmin}_y \{ c \cdot y + d \cdot \alpha(y)/y \}$, where $\alpha(y)$ is the Halfin–Whitt delay probability

Regimes: quality-driven ($N - R \gg \sqrt{R}$), QED ($N - R \approx \beta \sqrt{R}$), efficiency-driven ($N \approx R$)

Assumptions. Stationary Poisson arrivals, exponential service, no abandonment, one homogeneous pool of CSRs, linear staffing cost.

Solution method and validation. The authors analyse the problem as R grows large and show that the optimal staffing level falls into one of three operating regimes, depending on the ratio of delay cost to staffing cost. The optimal safety coefficient y^* comes from a one-dimensional optimisation, and the result is checked numerically against exact Erlang optimisation.

Findings. The square-root rule with an optimised coefficient is remarkably accurate – often exactly optimal – even for centres with few agents. The analysis also shows that safety staffing grows more slowly than the offered load as centre size increases, which quantifies economies of scale.

Strengths. Elegant closed form; robust; gives managers intuition about how service costs scale.

Limitations. No abandonment, single skill and stationary demand; delay costs are hard to estimate in practice.

Extensions. Add abandonment (P6), time-varying demand (P5), heterogeneous servers (P23) and uncertainty in R (P17).

5) P5 – Green, Kolesar & Whitt (2007) [12]: “Coping with time-varying demand when setting staffing requirements for a service system”

Domain	Call centres and other service systems with intraday demand variation
Problem type	Staffing requirement setting for non-stationary queues
Objective	Find the minimum staff in each period that keeps the delay probability below a target
Technique	Comparison of SIPP, lagged SIPP, modified-offered-load (MOL) and infinite-server approximations
Tools	Numerical solution of time-varying Markov chains; simulation

Real-life problem. Call arrivals can vary throughout the day. The current approach, Stationary Independent Period-by-Period (SIPP) analysis, describes each half hour period as a separate steady state Erlang queue. For fast load changes, or long handling times, this may not capture the lag in the system's response to changes in demand and so may under-staff for rising load and over-staff for falling load.

Why OR was suitable. The question is how to translate a time-varying forecast into staff numbers that actually deliver the target; a poor approximation leads to costly over- or under-staffing, and the requirements feed directly into the scheduling model.

Mathematical model. For a time-varying $M_t/G/s_t$ system with arrival rate $\lambda(t)$ and service-time distribution G , the offered load $m(t)$ is the mean number of busy servers in an infinite-server system. Staffing is then set with a square-root rule around $m(t)$.

$$m(t) = E \left[\int_{t-S}^t \lambda(u) du \right] = \int_{-\infty}^0 \lambda(t-u) (1 - G(u)) du$$

$$\text{Staffing (MOL / infinite-server): } s(t) = \lceil m(t) + \beta \sqrt{m(t)} \rceil$$

Target: $P(\text{Delay at } t) \leq \alpha$ for every t in the day

Lagged SIPP: apply Erlang C to $\lambda(t - E[S])$ instead of $\lambda(t)$

Assumptions. Poisson arrivals are assumed with a known rate function, and staffing can be changed at period boundaries. Abandonment is not considered in the base comparison.

Solution method and validation. The authors perform a broad number of numerical tests that involve finding the precise time-dependent performance at each point in

time by solving the Kolmogorov forward equations. Then they compare the relative effectiveness of each heuristic in achieving the desired targets. Their work identifies the cases in which SIPP is likely to fail.

Findings. Simple SIPP works for short calls and mild demand variation, where it lands close to the target, but it can miss badly otherwise. Lagged SIPP fixes most of its problems with little extra effort, and the infinite-server/MOL approach is the best all-round method.

Strengths. The study is pragmatic, numeric, and easily implemented into existing tools for managing workforces.

Limitations. The study only considers requirements determination, and not shift constraints or multi-skill pools. Arrival rates are also assumed to be known.

Extensions. Incorporate the MOL-based constraints into the scheduling IPs of P10–P11, and the arrival rate uncertainty as in P18–P19.

6) P6 – Mandelbaum & Zeltyn (2009) [13]:
“Staffing many-server queues with impatient customers: Constraint satisfaction in call centers”

Domain	Inbound call centres where callers hang up when kept waiting
Problem type	Staffing with service constraints in queues with abandonment (M/M/n+G)
Objective	Minimise the number of CSRs subject to limits on abandonment and waiting
Technique	Asymptotic analysis in efficiency-driven, QED and quality-driven regimes
Tools	Analytical approximations; validation against exact calculations and call-centre data

Real-life problem. Real callers have finite patience, which the Erlang C model (it assumes infinite patience) ignores. This can cause large under- or over-staffing when abandonment is not negligible. Managers often specify constraints such as abandonment below 3% or an average speed of answer below 30 seconds, rather than cost trade-offs.

Why OR was suitable. This is a constrained optimisation problem in the queueing model setting. Asymptotics also leads to simple, accurate staffing formulas for each of the constraints.

Mathematical model. Callers arrive at rate λ , service times are exponential with rate μ , and each caller abandons after a random patience time with distribution G. With offered load $R = \lambda/\mu$, the problem is to choose n .

Minimise n

subject to $P_n(\text{Abandon}) \leq \alpha$ and/or $P_n(W > T) \leq \delta$
and/or $E_n[W] \leq \bar{w}$
Efficiency-driven: $n \approx R(1 - \gamma)$ (when constraints are loose, e.g. $\alpha \geq 10\%$)
QED: $n \approx R + \beta \sqrt{R}$ (β depends on α and the patience density at 0)
Quality-driven: $n \approx R + c \sqrt{R}$ with large c

Assumptions. Stationary Poisson arrivals; exponential service; patience independent of all other variables; a single skill set; and a first-come-first-served queue discipline.

Solution method and validation. The authors calculate asymptotic expressions of the performance measures in each regime. They invert these expressions to obtain staffing rules. The authors also determine how managers implicitly choose one regime by certain restrictions and check their staffing rules by numerical calculations and empirical observations using the call-centre data.

Findings. With abandonment, it can be feasible – and economical – to staff below the offered load when moderate abandonment is tolerable. The type of constraint (abandonment, waiting probability or mean wait) determines which regime applies, so managers should keep these regimes in mind when setting targets.

Strengths. Directly addresses how managers actually set targets and gives explicit, accurate formulas.

Limitations. Single-skill and stationary; callers who redial after abandoning are not modelled without further assumptions; patience distributions must be estimated.

Extensions. Incorporate retries and callbacks, multiple skills (P9, P14) and uncertain arrival rates (P17).

B. Stream B – Shift Scheduling and Multi-Skill Staffing

These studies move from “how many” to “who, with which skills, on which shift”, covering multi-skilled CSRs, cross-training and practical work rules.

1) P7 – Fukunaga et al. (2002) [14]: “Staff scheduling for inbound call and customer contact centers”

Domain	Inbound call centres and multi-channel contact centres (phone, e-mail)
Problem type	Multi-skill, multi-channel CSR scheduling with labour rules and preferences
Objective	Weighted combination of service level, staffing cost and agent preference satisfaction
Technique	Constraint-based local search with fast simulation-based schedule evaluation
Tools	Commercial workforce-management software (Blue Pumpkin Director)

Real-life problem. Contact centres are no longer traditional single queue telephony environments. CSRs need to have multiple skills, handle e-mail as well as phone calls, and expect the shift preferences to be respected. Conventional Erlang spreadsheets are unable to evaluate the efficiency of a schedule in which agents handle multiple queues. This paper describes the scheduling engine of a commercial product that has been implemented in several centres.

Why OR was suitable. The problem exhibits integer scheduling decisions coupled with a performance measure, the service level in multiple interacting queues, which does not have a closed form. The suggested operations research/artificial intelligence (OR/AI) hybrid approach – search over schedules coupled with simulation to evaluate them – is well suited to this mixture.

Mathematical model. Decision: for every CSR k , a daily or weekly schedule σ_k (shift start, length, breaks, activities). The schedule set must satisfy hard constraints (labour law, contracts, break rules) and is scored by a weighted objective. Service levels per queue q and period t , $SL_{qt}(\sigma)$, are estimated by simulation of the multi-skill system.

$$\begin{aligned} \text{Minimise } F(\sigma) &= w_1 \sum_{\{q,t\}} \max(0, SL^*_{qt} - SL_{qt}(\sigma)) + w_2 \\ &\text{Cost}(\sigma) + w_3 \sum_k \text{Penalty}_k(\sigma_k) \\ \text{subject to } \sigma_k &\in \text{Feasible}_k \text{ (contract hours, break windows,} \\ &\text{days off, skill eligibility)} \\ \sigma &= (\sigma_1, \dots, \sigma_K) \end{aligned}$$

Assumptions. Call and e-mail forecasts per interval are given; the simulation model represents the routing logic accurately; and the weights reflect management priorities.

Solution method and validation. The solution uses a constraint-based representation to guarantee that each candidate schedule is feasible, and a local search using move and swap neighborhoods to look for improvements. A fast discrete-event simulation is used to evaluate each candidate's service level since no exact queueing formulas exist for multi-skill routing. The paper shares some lessons learned from using the system in real production contact centres.

Findings. Simulation-in-the-loop search was able to produce schedules that achieved the service goals with significantly less staff than the single-skill Erlang methods, while allowing the application of practical rules and preferences, which are normally omitted in classical

integer programming models. This is one of the first large-scale industrial examples of the use of optimisation for multi-skill contact-centre scheduling.

Strengths. Realistic: handles multi-skill, multi-channel work, preferences and labour rules, and has been proven in deployment.

Limitations. The heuristic search gives no optimality guarantee; results depend on the weight settings; proprietary details limit reproducibility.

Extensions. Embed the simulation evaluation in exact cutting-plane methods (P14, P15) or add stochastic demand scenarios (Stream D).

2) P8 – Shumsky (2004) [15]: “Approximation and analysis of a call center with flexible and specialized servers”

Domain	Call centres with two call types and cross-trained CSRs
Problem type	Queueing performance analysis and staffing mix (specialists vs. flexible agents)
Objective	Accurately predict service levels to choose how many specialised and flexible CSRs to employ
Technique	Markov-chain modelling with tractable approximations
Tools	Analytical approximations; numerical comparison with exact/simulated results

Real-life problem. Cross-training CSRs in order to handle more than one type of call is likely to improve pooling, but adds cost in training, paid hours and maybe slower call handling times. A call centre that takes two different types of call may have specialists on each type of call and a pool of flexible agents who can handle overflow from each type of call. It is known that managers need information on how service levels are affected by the number of agents of different types employed.

Why OR was suitable. The exact Markov chain model derived for this configuration grows quickly with increasing numbers of agents. Approximations are therefore needed before the staffing mix can be optimised.

Mathematical model. Two call types $j = 1, 2$ with arrival rates λ_j . Specialist pools of size N_1, N_2 serve only their type; a flexible pool of size N_f serves both. Calls are routed first to a free specialist and then to a free flexible agent. Performance measures (delay probability, service level) are functions of (N_1, N_2, N_f) . The design problem is to choose the mix.

$$\begin{aligned} \text{Minimise } &c_1 N_1 + c_2 N_2 + c_f N_f \\ \text{subject to } &SL_j(N_1, N_2, N_f) \geq SL^*_j \quad j = 1, 2 \end{aligned}$$

$N_1, N_2, N_f \in \mathbb{Z}_h$ (with $c_f > c_j$ reflecting cross-training cost)

Assumptions. The model is based on the assumptions of Poisson arrivals, exponential service, fixed priority routing with specialists assigned to the higher priority calls and no abandonment.

Solution method and validation. Shumsky constructs approximations that separate the system but still account for the overflow interactions between service pools. These approximations are then used to compare to the exact numerical solutions for various parameters. The approximations allow the performance changes that occur by adding flexible agents to be explored.

Findings. The approximations are accurate enough for planning purposes, indicating that cross-training a modest proportion of flexible customer service representatives can realise most of the potential benefits of full pooling. This gives a quantitative basis for choosing how many representatives to cross-train.

Strengths. This study is quick to conduct and provides valuable insights into a common real-world design problem, which supports an immediate decision on cross-training.

Limitations. The analysis focuses only on two types of calls and a single routing rule; abandonment and time-varying load are not considered.

Extensions. A model may be generalised to handle multiple skills and flexible routing rules (P9, P14, P25).

3) P9 – Wallace & Whitt (2005) [16]: “A staffing algorithm for call centers with skill-based routing”

Domain	Multi-skill call centres with skill-based routing (SBR)
Problem type	Joint staffing and skill-assignment (cross-training design) under SBR
Objective	Minimise total CSRs needed to meet per-call-type service targets
Technique	Erlang-based initial staffing + skill assignment heuristic + iterative simulation refinement
Tools	Discrete-event simulation

Real-life problem. In a skill-based routing centre each customer service representative (CSR) will have a particular skill set (for example languages and products) and each call type can only be handled by those with that skill set. How many CSRs should we employ and how should their skill sets be distributed? This is not straightforward as it will also depend on the routing policy and the extent of the skill overlap.

Why OR was suitable. This is a combined design problem (in terms of skill allocation to agents), and a sizing problem (as performance measures are not available in closed form), providing an interesting case for the application of heuristics in combination with simulation.

Mathematical model. Call types $j = 1, \dots, J$ (six in the main experiments). Each CSR k receives a skill set $S_k \subseteq \{1, \dots, J\}$. The staffing problem is to choose the total number of agents N and the skill sets so that every call type meets its service target.

Minimise $N = \sum_k 1$
subject to $SL_j(S_1, \dots, S_N; \text{routing}) \geq SL^*_j \quad j = 1, \dots, J$
 $|S_k| \leq m$ (number of skills per CSR, e.g. $m = 2$)
Step 1: $N_0 = \text{Erlang-C staffing for the pooled load } R = \sum_j \lambda_j / \mu$

Assumptions. In the base case, all call types and all agents require the same handling time; calls are routed according to priority and according to each CSR's strongest skill; two-skill CSRs follow a balanced 'chaining' pattern.

Solution method and validation. In step 1, a single-pool Erlang computation is used to compute the workload total, which yields a first headcount estimate; then, in step 2, primary and secondary skills are distributed to give adequate coverage; in the third step, a simulation is performed, and staffing levels are tuned to meet the goals for each call type.

Findings. The results show that two skills per CSR allow performance close to full cross-training, demonstrating that "a little flexibility goes a long way." As a result, pooled single-skill Erlang staffing is a good proxy for total headcount in well-designed SBR centres.

Strengths. The technique provides a simple, practical algorithm along with the important managerial insight on cross-training that is confirmed by simulation.

Limitations. It assumes a number of symmetry assumptions, such as comparable handling time and balanced demand, while using a fixed routing rule and stationary demand.

Extensions. Possible extensions to our results include asymmetric loads and heterogeneous agents (P24) and with shift scheduling (P10, P15).

4) P10 – Bhulai, Koole & Pot (2008) [17]: “Simple methods for shift scheduling in multiskill call centers”

Domain	Multi-skill call centres
Problem type	Two-step staffing and shift scheduling; integer/linear programming
Objective	Minimise the number (cost) of scheduled CSRs while meeting service levels per skill
Technique	Approximate staffing per interval followed by an LP/IP shift-covering model with skill substitution
Tools	LP/IP solver; simulation for validation

Real-life problem. Optimal scheduling of multi-skilled personnel is hard because the service levels of all skill groups, over all time intervals, are interconnected. The authors sought a method that is as practical as possible while still exploiting the flexibility of multi-skilled CSRs.

Why OR was suitable. Shift scheduling is well modelled as a covering IP; the skill dimension provides for substitution, which can be handled by an LP. In addition, the separation of the problem into a staffing component and a scheduling component makes the problem more manageable.

Mathematical model. Let g index agent groups (skill sets) and t the intervals. Step 1 produces staffing requirements y_{tg} . Step 2 chooses x_{sg} , the number of agents of group g on shift s . Because an agent with more skills can stand in for one with fewer, coverage constraints are written over sets of groups rather than individually.

$$\begin{aligned} &\text{Minimise } \sum_s \sum_g c_{sg} x_{sg} \\ &\text{subject to } \sum_s \sum_{\{g \in G(A)\}} a_{ts} x_{sg} \geq \sum_{\{g \in G(A)\}} y_{tg} \text{ for all } t \text{ and relevant skill subsets } A \\ &\quad x_{sg} \in \mathbb{Z}_+ \quad (G(A): \text{groups whose skill set includes subset } A) \end{aligned}$$

Assumptions. Staffing requirement per interval can be calculated independently. It is assumed that substitution within groups is sufficiently modelled by coverage constraints on a group level, and load is stationary within the interval.

Solution method and validation. A quick approximation of the multi-skill staffing need is computed in Step 1 for each time interval. Then, in Step 2, the shift covering model LP/IP is solved and schedules are validated through simulation and benchmarking to lower bounds and advanced techniques.

Findings. The two-step method provides schedules within a few percent of the best-known solutions. Being simpler and quicker to compute, the two-step method should appeal to practitioners.

Strengths. Simple, efficient and transparent; compatible with standard solvers.

Limitations. Two-step decomposition can lose savings compared with one-step optimisation (P15), and approximation errors in step 1 can carry through.

Extensions. Feed simulation results back into step 2 (as in P15) and include uncertain arrival rates.

5) P11 – Dietz (2011) [18]: “Practical scheduling for call center operations”

Domain	Operating call centres (CSR tour scheduling)
Problem type	Mixed-integer programming for tour/shift scheduling with practical rules
Objective	Match scheduled CSRs to forecast requirements at minimum cost while honouring practical constraints
Technique	MIP formulation with break placement and work-rule constraints
Tools	Commercial MIP solver

Real-life problem. Many academic scheduling models neglect constraints that must be respected in practice: break placement, start-time bands, part-time contracts and consecutive days off. As a result, such models are often considered unusable. Dietz stresses the need for schedules that a call-centre planner can actually use.

Why OR was suitable. Scheduling tours with many work rules is a big but well-structured integer programming problem. Modern mixed-integer programming (MIP) solvers can handle it directly if it is formulated carefully.

Mathematical model. Let x_{τ} be the number of CSRs assigned to tour τ (which fixes working days, shift start, length and break positions). Requirements r_t per interval come from the forecast. Under- and over-staffing variables u_t, o_t absorb mismatches.

$$\begin{aligned} &\text{Minimise } \sum_{\tau} c_{\tau} x_{\tau} + \sum_t (p_u u_t + p_o o_t) \\ &\text{subject to } \sum_{\tau} a_{t\tau} x_{\tau} + u_t - o_t = r_t \quad \text{for all intervals } t \\ &\quad \sum_{\tau \in T_{PT}} x_{\tau} \leq P \quad (\text{part-time limits}), \text{ break-window and start-band rules} \\ &\quad x_{\tau} \in \mathbb{Z}_+, u_t, o_t \geq 0 \end{aligned}$$

Assumptions. Requirements per interval are given (typically from Erlang calculations); costs of under- and over-staffing are linear; and the base model has a single skill group.

Solution method and validation. Implicit or compact representations of the breaks are used to keep the model manageable, and a commercially available mixed integer programming (MIP) solver is used to solve the model. Results are subsequently validated against schedules based on current practices.

Findings. Including practical rules does not make the model intractable. The resulting schedules were more

realistic than manual or spreadsheet schedules and can directly save paid hours. Dietz notes that implementation issues matter as much as the model itself.

Strengths. The work is practitioner-oriented, complies with real work rules, and can be reduced to a problem solvable in standard software.

Limitations. Requirements are assumed to be predetermined, no queueing feedback is assumed, there is only one skill, and demand is assumed to be deterministic.

Extensions. Future work may combine the model with aggregate service-level constraints (P2), multi-skill coverage (P10) and stochastic requirements (P18–P20).

C. Stream C – Simulation-Based Optimisation

When routing and inter-period effects make formulas unreliable, simulation becomes the evaluation engine inside integer programming.

1) P12 – Atlason, Epelman & Henderson (2004) [19]: “Call center staffing with simulation and cutting plane methods”

Domain	Inbound call centres with time-varying demand
Problem type	Integer shift scheduling with simulation-estimated service-level constraints
Objective	Minimise staffing cost subject to a service level in every period
Technique	Kelley-type cutting-plane method with subgradients estimated by simulation (sample-average approximation)
Tools	Discrete-event simulation + integer-programming solver

Real-life problem. The standard analytical staffing formula does not consider the interaction between periods. In practice, calls still waiting at the end of one period spill over into the next, so performance in one period depends on staffing in earlier periods. The authors' goal was to find the best shift schedules using a simulation that would model these dynamics.

Why OR was suitable. The service-level function has no closed form but can be estimated by simulation, and if it is concave in the number of staff, linear cuts can approximate it inside an IP model.

Mathematical model. Let x be the vector of CSRs per shift and $N = Ax$ the staffing in each period. $f_t(N)$ is the expected fraction of calls in period t answered within the target, estimated by simulation over n replications.

$$\begin{aligned} &\text{Minimise } c^T x \\ &\text{subject to } \hat{f}_t(Ax) \geq 1_t \quad t = 1, \dots, T \text{ (simulated service level)} \end{aligned}$$

$$\begin{aligned} &\text{Cut added at trial point } \bar{N}: \hat{f}_t(\bar{N}) + \bar{q}_t^T(N - \bar{N}) \geq 1_t \\ &(\bar{q}_t = \text{finite-difference subgradient}) \\ &x \in \mathbb{Z}^h \end{aligned}$$

Assumptions. Service-level functions are concave over the relevant region (so cuts remove no optimal solutions); simulation with common random numbers gives consistent estimates; and the arrival process is known.

Solution method and validation. The method alternates between solving a relaxed IP and simulating its solution. Where a period's service constraint is violated, a linear cut based on estimated subgradients is added and the IP is re-solved. Convergence for the sample-average problem is shown by increasing the simulation length, and the approach is tested numerically.

Findings. The cutting-plane method finds schedules satisfying all period targets more economically than period-by-period analytical techniques, by capturing cross-period effects. Concavity can break down at low staffing levels, which motivates the extension in P13.

Strengths. Combines simulation accuracy with IP optimisation; rigorous methodology.

Limitations. Computationally expensive; assumes concavity and a single skill.

Extensions. The results motivate relaxing the concavity assumption (P13) and extending the implementation to multi-skill centres (P14, P15).

2) P13 – Atlason, Epelman & Henderson (2008) [20]: “Optimizing call center staffing using simulation and analytic center cutting-plane methods”

Domain	Inbound call centres
Problem type	Simulation-based integer staffing/scheduling
Objective	Minimise cost subject to period service levels
Technique	Analytic center cutting-plane method (ACCPM) with simulation; handles pseudoconcave service functions
Tools	Discrete-event simulation; ACCPM implementation; IP solver

Real-life problem. This paper extends P12. Empirical data show that the service-level curve is S-shaped: flat when staffing is very low, steep during the transition, and flat again when staff are plentiful. Such functions are not concave, so the cutting-plane approach used earlier can remove optimal solutions.

Why OR was suitable. Cutting-plane algorithms based on the analytic centre of the current feasible region suit pseudoconcave constraints and may need fewer simulations.

Mathematical model. Same scheduling model as P12. The key change is the assumption that f_t is pseudoconcave rather than concave, and that cuts are generated only at points where the function behaves well.

Minimise $c^T x$ subject to $f_t(Ax) \geq l_t$ for all t , $x \in Z_h$
ACCPM: next trial point = analytic centre of $\{x : c^T x \leq z_{UB}, \text{ all cuts so far}\}$
Feasibility cut (if f_t violated): $\bar{q}_t^T (Ax - \bar{N}) \geq 0$
Optimality cut (if feasible): $c^T x \leq c^T \bar{x}$

Assumptions. Service functions in a single-skill centre are pseudoconcave, and their (pseudo)gradients can be estimated by simulation.

Solution method and validation. The authors choose the trial points based on the Analytic Centre Cutting Plane Method (ACCPM), since the points are pulled towards the centre of the available feasible region to maintain simulated points within the well-behaved portion of the service curve. They describe the conditions for validity of the method and show their numerical experiments comparing the method to the previously described Kelley method and analytical heuristics.

Findings. The analytic-centre method is more robust to non-concavity and, in many settings, needs fewer simulation calls to find a cost-effective schedule, which suggests that simulation-optimisation is a practical tool for real staffing decisions.

Strengths. More robust than P12; good theoretical properties; general enough for other service systems.

Limitations. Still computationally demanding; pseudoconcavity can also fail; tested only in single-skill settings.

Extensions. Use the same idea for multi-skill and uncertain-arrival settings, with parallel simulation to cut run time.

3) P14 – Cezik & L'Ecuyer (2008) [21]: “Staffing multiskill call centers via linear programming and simulation”

Domain	Large multi-skill call centres
Problem type	Multi-skill staffing integer programme with simulated service constraints
Objective	Minimise cost of agents across skill groups subject to per-type and global service levels
Technique	IP + simulation cutting planes, rounding heuristics and local search
Tools	CPLEX for (I)LPs; ContactCenters simulation library (Java/SSJ)

Real-life problem. For multi-skill centres, a supervisor must decide how many CSRs from each skill group to

employ for a planned period. Calls of one type may be handled by several groups so service levels depend on the complete staffing vector plus the routing rules. There are no analytical formulas for realistic routing configurations.

Why OR was suitable. The number of staff is an integer vector subject to the constraints of a well-structured complex stochastic system. The article shows how a simulation-based cutting-plane integer programming method can solve real-world sized problems.

Mathematical model. Call types $k = 1, \dots, K$; agent groups $i = 1, \dots, I$ with costs c_i ; y_i is the number of agents in group i . $g_k(y)$ is the long-run fraction of type- k calls answered within the target, $g(y)$ the aggregate service level, both estimated by simulation.

Minimise $\sum_i c_i y_i$
subject to $g_k(y) \geq l_k \quad k = 1, \dots, K$
 $g(y) \geq l$
 $y \in Z_h^I$
Linear cut at $\bar{y}$: $\hat{g}_k(\bar{y}) + \hat{q}_k^T (y - \bar{y}) \geq l_k$

Assumptions. The model is a single period model with constant arrival rates, assumes that the service functions are concave over the relevant region, and assumes a static routing policy.

Solution method and validation. The algorithm solves the integer programming model or the linear programming relaxation of it, processes the suggested solution and add cuts for violated constraints, uses rounding heuristics and local search to remove unnecessary agents. Testing has been done on call centres with up to about 20 call types and 35 agent groups.

Findings. The method is shown to identify near-optimal staffing for realistic multi-skill centres in a reasonable amount of time, and to provide substantial improvement over simple heuristics. It is expected to be a benchmark multi-skill staffing procedure.

Strengths. The model handles the real world sizes and routing efficiently, combines carefully derived methodology with heuristics, and is built using open simulation.

Limitations. Single period (no shifts); concavity may fail; computing time grows with problem size.

Extensions. Combine with shift scheduling in one model (P15) and incorporate demand uncertainty (P17).

4) P15 – Avramidis, Chan, Gendreau, L'Ecuyer & Pisacane (2010) [22]: “Optimizing daily agent scheduling in a multiskill call center”

Domain	Multi-skill call centres – daily CSR scheduling
Problem type	Simultaneous staffing and shift scheduling (one-step) integer programme
Objective	Minimise cost of shifts assigned to each skill group subject to service levels per type and period
Technique	Simulation-based cutting planes in an IP, randomised rounding and local search
Tools	CPLEX; ContactCenters simulation library

Real-life problem. Industry practice and most earlier models (e.g. P10) first compute the staffing required in each period and then choose shifts to cover it. This can be suboptimal, because requirements rarely coincide exactly with the available shifts. The authors propose an integrated optimisation.

Why OR was suitable. One-step optimisation lets the model exploit flexibility across periods and skill groups; service constraints still require simulation, so the same IP-plus-simulation framework as P14 is used, now with shifts.

Mathematical model. x_{si} = number of agents of skill group i assigned to shift s ; staffing in period t is $y_t = \sum_s a_{ts} x_{si}$. Service levels $g_{tk}(y)$ per call type k and period t , and aggregate ones, are estimated by simulation.

$$\begin{aligned} &\text{Minimise } \sum_s \sum_i c_{si} x_{si} \\ &\text{subject to } y_t = \sum_s a_{ts} x_{si} \quad \text{for all } t, i \\ &\quad g_{tk}(y_t) \geq l_{tk}, \quad g_t(y_t) \geq l_t, \quad g(y) \geq l \\ &\quad \text{(simulated)} \\ &\quad x_{si} \in \mathbb{Z}^+ \end{aligned}$$

Assumptions. For the purpose of simulation analysis, each period is assumed to be stationary within itself with a fixed routing policy and deterministic arrival rates specified in each period.

Solution method and validation. Cutting planes derived from simulated subgradients approximate the service constraints. Because exact integer solutions are computationally expensive, the authors solve LP relaxations, apply randomised rounding and then improve the schedules by local search. The one-step procedure is compared with the two-step procedure on a number of test centres.

Findings. The one-step method yields significant cost savings relative to the two-step method, especially when shifts are rigid, showing that decoupling staffing and scheduling is costly in multi-skill settings.

Strengths. Integrated model of realistic multi-skill size; quantifies the value of integration.

Limitations. Computationally intensive; arrival rates are assumed known; routing is not optimised.

Extensions. Add arrival-rate uncertainty (P18–P20) and joint optimisation of routing weights (P25).

5) P16 – Ingolfsson, Campello, Wu & Cabral (2010) [23]: “Combining integer programming and the randomization method to schedule employees”

Domain	Call centres and similar service systems with time-varying demand
Problem type	Shift scheduling with exact transient queueing evaluation
Objective	Minimise staffing cost while meeting service-level targets
Technique	Integer programming iterated with the randomisation (uniformisation) method for transient Markov queues
Tools	IP solver; numerical transient-analysis code

Real-life problem. SIPP-style techniques, which treat each period as a steady-state queue, are inaccurate when arrival rates vary (see P5). Simulation can capture this but is slow and noisy; the randomisation method computes the transient distribution of a Markov queue quickly and exactly.

Why OR was suitable. Exact transient analysis combined with integer programming gives accurate service evaluation without simulation noise, so the optimisation converges more reliably.

Mathematical model. x_s = CSRs on shift s ; staffing function $s(t) = \sum_s a_{ts} x_s$. For a given staffing function, the randomisation method computes the expected service level $SL_t(x)$ in each period of an $M_t/M/s_t$ queue. Constraints can be per period or aggregated over the day.

$$\begin{aligned} &\text{Minimise } \sum_s c_s x_s \\ &\text{subject to } SL_t(x) \geq l_t \quad (\text{per period}) \quad \text{or} \quad \sum_t \lambda_t SL_t(x) \geq \alpha \sum_t \lambda_t \quad (\text{daily}) \\ &\quad \text{Iteration: solve IP with linearised constraints} \rightarrow \text{evaluate } SL_t \text{ exactly by randomisation} \rightarrow \text{add constraints/cuts} \rightarrow \text{repeat} \\ &\quad x_s \in \mathbb{Z}^+ \end{aligned}$$

Assumptions. Markovian (exponential) service; Poisson arrivals at known piecewise-constant rates; single skill group.

Solution method and validation. The algorithm alternates between an IP that proposes a schedule and

the randomisation method that evaluates it, adding constraints where targets are missed until the schedule passes. Results are compared with SIPP-based scheduling on realistic demand profiles.

Findings. Schedules found this way meet targets that SIPP-based schedules miss, often at similar or lower cost, because the evaluation carries congestion forward into later periods.

Strengths. Precise and noise-free assessment; more efficient than simulation; accommodates aggregate goals.

Limitations. Restricted to Markovian single-skill models; the state space grows with the number of agents.

Extensions. Add abandonment and multi-skill pools; combine with forecast uncertainty.

D. Stream D – Planning under Forecast Uncertainty

These studies use chance-constrained, stochastic and robust optimisation so that schedules remain good when demand deviates from forecast.

1) P17 – Gurvich, Luedtke & Tezcan (2010) [24]: “Staffing call centers with uncertain demand forecasts: A chance-constrained optimization approach”

Domain	Multi-class, multi-pool call centres
Problem type	Chance-constrained staffing and routing under arrival-rate uncertainty
Objective	Minimise staffing cost so that service targets are met with a chosen probability
Technique	Chance-constrained MIP via sample-average approximation; many-server queueing approximations
Tools	MIP solver; simulation for validation

Real-life problem. Call-volume forecasts are uncertain even a day ahead because the arrival rate itself is random. Relying on a single point forecast for staffing can lead to regular target breaches, so managers need staffing decisions that meet targets with a specified probability.

Why OR was suitable. Chance-constrained programming is the OR tool for decisions that must satisfy constraints with a stated probability, and it combines with queueing approximations without computational difficulty.

Mathematical model. Customer classes k and agent pools i with costs c_i ; N_i agents in pool i . The arrival-rate vector Λ is random. For each realisation, the set $F(\Lambda)$ contains staffing vectors for which some routing achieves every class's service target. The manager chooses N so that the probability of being in $F(\Lambda)$ is at least $1 - \epsilon$.

Minimise $\sum_i c_i N_i$

subject to $P(N \in F(\Lambda)) \geq 1 - \epsilon$

SAA: for scenarios $\omega = 1, \dots, M$: $N \in F(\Lambda^\omega)$ or $z_\omega = 1$;

$\sum_\omega z_\omega \leq \lfloor \epsilon M \rfloor$, $z_\omega \in \{0, 1\}$

$F(\Lambda^\omega)$: linear routing-feasibility constraints $\sum_i \rho_{ki\omega} \geq \text{load}_{k\omega}$, $\sum_k \rho_{ki\omega} \leq N_i$ – safety terms

Assumptions. The distribution of arrival rates can be sampled (e.g. from historical forecast errors); each realisation operates in the QED/many-server regime, and routing can be adjusted after the rates are observed.

Solution method and validation. We use sample-average approximation to approximate the chance constraint as a mixed-integer program with binary scenario indicators. Queueing asymptotics are used to convert each service requirement into a linear capacity constraint. All solutions are validated by simulation of the full queueing network.

Findings. The approach produces staffing levels that achieve targets at the desired reliability and lets the manager measure the incremental cost of increased reliability by one percentage point. Also, by ignoring forecast uncertainty, service levels are systematically under-delivered.

Strengths. Explicit risk control; handles multi-skill systems; theoretically sound.

Limitations. Requires a credible distribution for arrival rates; SAA problems can be large; limited to a single period.

Extensions. Extend to multi-period scheduling with joint chance constraints (P19) and to robust optimisation (P20).

2) P18 – Robbins & Harrison (2010) [25]: “A stochastic programming model for scheduling call centers with global service level agreements”

Domain	Call centres operating under contractual service-level agreements (SLAs)
Problem type	Two-stage stochastic integer programme for CSR scheduling
Objective	Minimise staffing cost plus expected penalty for missing a global (horizon-wide) SLA
Technique	Stochastic programming with sample-average approximation; linearised service-level curves
Tools	Commercial MIP solver; simulation for evaluation

Real-life problem. Many outsourced call centres have Service Level Agreements (SLAs) that are evaluated over a week or month, in terms of a 'global' SLA, rather than every half hour. Schedules must be fixed before volumes are known, and volumes are uncertain; the authors model this situation explicitly.

Why OR was suitable. Decisions are taken in two stages: schedule first, then observe demand. This structure fits two-stage stochastic programming, in which the trade-off against the global SLA is exploited across periods.

Mathematical model. First-stage variables x_s : CSRs on shift s . Scenarios ω represent possible arrival-rate profiles $\lambda_{t\omega}$ with probability p_ω . Second-stage variables describe the service achieved in each period and scenario through a piecewise-linear approximation of the service-level curve. A shortfall variable u_ω measures how far the global SLA is missed.

$$\begin{aligned} & \text{Minimise } \sum_s c_s x_s + \sum_\omega p_\omega \cdot \pi \cdot u_\omega \\ & \text{subject to } n_t = \sum_s a_{ts} x_s \quad \text{for all } t \\ & v_{t\omega} \leq \alpha_{jt\omega} + \beta_{jt\omega} n_t \quad (\text{piecewise-linear service curve, segments } j) \\ & \sum_t v_{t\omega} + u_\omega \geq SL^* \cdot \sum_t \lambda_{t\omega} \quad \text{for all } \omega \quad (\text{global SLA}) \\ & x_s \in \mathbb{Z}^+, u_\omega, v_{t\omega} \geq 0 \end{aligned}$$

Assumptions. Service in each period can be approximated by a concave function of staffing in each scenario; the scenarios capture forecast uncertainty adequately; and the linear penalty for SLA shortfall is appropriate.

Solution method and validation. The deterministic equivalent is solved by sample-average approximation using a MIP solver. The value of the stochastic solution is evaluated versus a model with mean forecast. The data presented for the evaluation of the stochastic solution reflect the demand levels and uncertainty of an operational call centre.

Findings. The results indicate that, when compared to the deterministic schedule, the stochastic schedule is able to meet the SLA more consistently and, on average, at a lower total cost. This demonstrates the considerable value of the stochastic solution in the presence of forecast errors.

Strengths. Matches real contractual structures; exploits the flexibility of global SLAs; gives a clear economic interpretation.

Limitations. The study is confined to one skill, and piecewise approximations don't treat the carry-over of queues. The generation of scenarios also influences the results.

Extensions. Possible further directions might be multi-skill stochastic scheduling, extending the problem with

recourse actions (e.g. Overtime, subcontracting), or considering risk-averse objectives.

3) P19 – Excoffier, Gicquel & Jouini (2016) [26]: “A joint chance-constrained programming approach for call center workforce scheduling under uncertain call arrival forecasts”

Domain	Inbound call centres – weekly or daily workforce scheduling
Problem type	Joint chance-constrained integer programme for shift scheduling
Objective	Minimise staffing cost so that service targets in all periods hold jointly with high probability
Technique	Joint chance constraints reformulated/approximated as a deterministic MILP
Tools	MILP solver; simulation for out-of-sample evaluation

Real-life problem. When call volume forecasts are subject to uncertainty, "on average" plans that meet targets may fail for a large number of individual periods. Individual chance constraints protect each period separately but do not guarantee the entire day is met. The authors require all periods to satisfy their requirements together with a given probability.

Why OR was suitable. A joint chance constraint describes the manager's true overall daily success condition. The OR reformulation methods could even convert it into a tractable MILP.

Mathematical model. x_s = CSRs on shift s . In each period t the random arrival rate Λ_t implies a required staffing level $R_t(\Lambda_t)$ obtained from an Erlang-type service model. The joint chance constraint requires coverage in every period at once.

$$\begin{aligned} & \text{Minimise } \sum_s c_s x_s \\ & \text{subject to } P(\sum_s a_{ts} x_s \geq R_t(\Lambda_t) \text{ for all } t) \geq 1 - \epsilon \\ & \text{With independent periods: } \prod_t P(\sum_s a_{ts} x_s \geq R_t(\Lambda_t)) \geq 1 - \epsilon \\ & \Rightarrow \sum_t \ln P(\sum_s a_{ts} x_s \geq R_t(\Lambda_t)) \geq \ln(1 - \epsilon) \quad (\text{linearised}) \\ & x_s \in \mathbb{Z}^+ \end{aligned}$$

Assumptions. The distribution of arrival rates per period is assumed to be known (for example, based on historical forecast-error data). Staffing requirements are assumed to be a monotone function of the arrival rate, and periods are assumed to be independent in the base formulation.

Solution method and validation. Piecewise-linear approximations are used to convert the joint constraint into a deterministic MILP model, exploiting the structure of the distribution. The resulting model is then solved using a commercial solver. These results are then validated using simulation to compare with those

obtained by the deterministic and individual-chance-constrained models.

Findings. The results indicate that, compared to the individual chance constraints, the joint formulation is more robust in terms of daily service, but with only a small price increase. In addition, the joint formulation outperforms the deterministic model, in the face of forecast uncertainty.

Strengths. The model captures very well the daily reliability, is tractable, and provides a transparent parameter risk.

Limitations. Periods are assumed to be independent. It considers one skill only, and the mapping of requirements does not support queue carry-over.

Extensions. Possible extensions include correlated forecast errors, multi-skill pools, and intraday re-scheduling.

4) P20 – Mattia, Rossi, Servilio & Smriglio (2017) [27]: “Staffing and scheduling flexible call centers by two-stage robust optimization”

Domain	Flexible call centres where CSRs can be re-allocated after demand is observed
Problem type	Two-stage robust integer programme for staffing and scheduling
Objective	Minimise cost in the worst case over an uncertainty set of demand
Technique	Two-stage robust optimisation with budgeted uncertainty, solved by decomposition (cutting planes)
Tools	MIP solver; tests on real-life call-centre data

Real-life problem. Probability distributions for call volume are often unreliable. Most centres have some operational flexibility to either reallocate CSR to alternate activities or to modify schedules as real volumes become more apparent. The authors' approach casts planning as a robust decision problem with "adaptive" second-stage adjustments.

Why OR was suitable. Robust optimisation guarantees feasibility for all demand scenarios within a defined set without introducing probabilistic assumptions, and the two-stage version captures the value of flexibility neatly.

Mathematical model. First stage: x = shift assignments made before demand is known. Uncertainty set $U(\Gamma)$ contains demand vectors d where at most Γ periods deviate from their nominal values. Second stage: $y(d)$ = adjustments (re-allocation of agents, overtime, or under-coverage) chosen after d is revealed.

$$\text{Minimise}_x \quad c^T x + \max_{d \in U(\Gamma)} \min_{y \in Y(x,d)} q^T y$$

$$U(\Gamma) = \{ d : d_t = \bar{d}_t + \hat{d}_t z_t, 0 \leq z_t \leq 1, \sum_t z_t \leq \Gamma \}$$

$$Y(x,d): \text{coverage } \sum_s a_{ts} x_s + \text{flex}_t(y) \geq d_t, \text{ flexibility limits on } y$$

Assumptions. Nominal forecasts and maximum deviations are known; the budget Γ limits how many periods can deviate together; and recourse actions have linear costs.

Solution method and validation. The min-max-min problem is solved by a decomposition approach where the worst-case demand scenarios are added sequentially as a cutting plane or a column-and-constraint method. The authors validate the model with empirical data and compare robust solutions against deterministic solutions under simulated demands.

Findings. Robust schedules protect service against demand surges at a moderate “price of robustness”, and second-stage flexibility can substantially reduce that price.

Strengths. Probability distribution is not necessary; good representation of flexibility and control over the degree of conservatism.

Limitations. A worst-case focus can be conservative; computation grows with Γ and the horizon; queueing effects are simplified into requirements.

Extensions. There is potential to analyse distributionally robust models and multi-skill flexible pools in subsequent research.

E. Stream E – Dynamic Call-to-CSR Routing

These studies bring the assignment problem into real time: when a call arrives or a CSR becomes free, who should be matched with whom?

1) P21 – Gans & Zhou (2007) [28]: “Call-routing schemes for call-center outsourcing”

Domain	Firms that split call volume between an in-house centre and an outsourcing vendor
Problem type	Routing (dynamic call-to-agent assignment) and capacity design
Objective	Minimise in-house staffing cost plus outsourcing fees and service penalties
Technique	Queueing/Markov decision models of threshold routing policies
Tools	Analytical queueing models; numerical experiments

Real-life problem. Many organisations have decided to keep high-value customer calls in house and outsource calls from customers with less value. Deciding which calls to send to the vendor, and when, is a key consideration,

since in-house CSRs usually cost more but may deliver better service. Vendors usually charge per call or per agent.

Why OR was suitable. This was a dynamic assignment problem where each call is to be assigned to either a particular in-house CSR, a queue, or an external vendor, depending on the conditions. Markov decision processes and queueing models are conventional approaches to such problems.

Mathematical model. Two call types: high-value (H) kept in-house and lower-value (L) that may be handled in-house or routed to the vendor. In-house staff N ; state = number of calls of each type in the system. A threshold policy routes L calls to the vendor when in-house congestion exceeds a level K .

Minimise $c_N \cdot N + c_O \cdot \lambda_L \cdot P(L \text{ routed out} | N, K) + h_H \cdot E[W_H] + h_L \cdot E[W_L]$
subject to service constraint for H calls, e.g. $P(W_H > \tau) \leq \delta$
Policy: route L to vendor $\Leftrightarrow$ (number of busy in-house agents + queue) $\geq K$

Assumptions. Poisson arrivals; Exponential service; H calls have priority-in-house; the vendor's capacity can be regarded as unlimited as far as the firm is concerned.

Solution method and validation. The authors analyse families of routing policies, derive performance measures and compare policies numerically across cost parameters to determine when each is best.

Findings. Simple threshold routing captures most of the benefit of optimal dynamic control; the best scheme and the in-house staffing level are highly sensitive to the structure of the outsourcing contract.

Strengths. Addresses a common strategic–operational decision; insightful policies.

Limitations. Two call types only; vendor quality and behaviour are simplified; demand is stationary.

Extensions. Include vendor service quality, multiple vendors and time-varying demand.

2) P22 – Sisselman & Whitt (2007) [29]: “Value-based routing and preference-based routing in customer contact centers”

Domain	Customer contact centres with heterogeneous CSRs
Problem type	Assignment-based routing: matching call types to CSRs by value or preference
Objective	Maximise total value (revenue, quality, agent satisfaction) while maintaining service levels

Technique	Assignment/LP-style optimisation of agent–call-type priorities evaluated by simulation
Tools	Optimisation model + discrete-event simulation

Real-life problem. CSRs differ in skill and in how much they enjoy different kinds of call. Routing calls purely to minimise waiting ignores the opportunity to direct particular calls to particular CSRs where sales conversion, customer satisfaction or agent preference matter, whether for morale or for turnover.

Why OR was suitable. Choosing which CSRs should handle which call types is a classical assignment problem with a value matrix; the twist is that the assignment becomes routing priorities in a dynamic queueing system.

Mathematical model. Let v_{ij} be the value created when CSR i handles a call of type j (or the preference score of CSR i for type j). Decision variables $x_{ij} \in [0, 1]$ indicate how strongly type j is assigned to agent i , subject to agents' capacity and the need to keep every call type adequately covered. The solution is converted into a priority ordering used by the router.

Maximise $\sum_i \sum_j v_{ij} x_{ij}$
subject to $\sum_j x_{ij} \leq 1$ for each CSR i (capacity / time shares)
 $\sum_i \mu_{ij} x_{ij} \geq \lambda_j / \kappa_j$ for each call type j (coverage with headroom κ_j)
 $x_{ij} \geq 0$; router gives type j higher priority at agent i when x_{ij} (or v_{ij}) is larger

Assumptions. Values or preferences can be quantified; service levels are maintained through sufficient staffing; and coverage and skill eligibility are known.

Solution method and validation. The optimisation creates a set of priority levels between each agent and call type. A more detailed simulation is then used to measure the value and service levels, and compare this routing with skill-based routing.

Findings. Value-based and preference-based routing can substantially increase the value generated or agent satisfaction, with only small adverse effects on waiting times, provided staffing is sufficient.

Strengths. Brings the classical assignment problem directly into contact-centre routing and humanises routing by considering agents' preferences.

Limitations. Values and preferences are difficult to measure reliably and fixed priorities can sometimes be unresponsive to current congestion.

Extensions. Learn values from data with machine learning; combine with dynamic routing (P25) and fairness considerations.

3) P23 – Armony & Mandelbaum (2011) [30]:
“Routing and staffing in large-scale service systems: The case of homogeneous impatient customers and heterogeneous servers”

Domain	Large call centres with CSR pools that differ in speed (and cost)
Problem type	Joint staffing and routing in an inverted-V queueing system with abandonment
Objective	Minimise staffing cost subject to service/abandonment constraints, with optimal routing
Technique	Asymptotic analysis (QED regime); Fastest-Servers-First routing
Tools	Analytical derivations; numerical experiments

Real-life problem. CSRs differ: experienced agents handle calls faster and are paid more, while new hires are slower and cheaper. Customers are treated as homogeneous. Managers must decide how many CSRs of each type to staff and whether an arriving call should go to an idle experienced or inexperienced CSR.

Why OR was suitable. Routing and staffing interplay; asymptotic queueing theory leads to tractable, near-optimal joint rules.

Mathematical model. Customers arrive at rate λ and abandon at rate θ . Pools $j = 1, \dots, J$ have N_j agents with service rate μ_j and cost $c_j(\mu_j)$. The router chooses which idle pool receives each arrival.

Minimise $\sum_j c_j N_j$
subject to $P(\text{Abandon}) \leq \alpha$ (or a delay constraint) Routing (FSF): assign an arriving call to an idle agent in the fastest available pool
Staffing (QED): $\sum_j \mu_j N_j \approx \lambda + \beta \sqrt{\lambda}$, with the split across pools chosen by cost

Assumptions. Homogeneous customers; exponential service and patience; many-server (large-scale) regime.

Solution method and validation. The authors prove that Fastest-Servers-First (FSF) routing combined with an appropriate staffing rule is asymptotically optimal and analyse its performance in the asymptotic regime. Numerical experiments validate the results for moderate sizes.

Findings. Routing calls to the fastest idle agents improves performance and, with the right staffing rule, is asymptotically optimal. The paper notes that this routing overloads fast agents relative to slow ones.

Strengths. Joint treatment of staffing and routing; simple, implementable rules.

Limitations. FSF raises fairness concerns because fast agents work more; homogeneous customers; exponential assumptions.

Extensions. Fairness-aware routing and heterogeneous customer classes; link speed to quality (P24).

4) P24 – Mehrotra, Ross, Ryder & Zhou (2012) [31]:
“Routing to manage resolution and waiting time in call centers with heterogeneous servers”

Domain	Call centres where CSRs differ in speed and first-call resolution
Problem type	Routing (dynamic agent assignment) with quality and speed trade-offs
Objective	Reduce waiting time and repeat calls (improve resolution) jointly
Technique	Queueing model with callbacks; comparison of routing policies; simulation calibrated on data
Tools	Statistical analysis of call-centre data; discrete-event simulation

Real-life problem. A CSR who is fast but often fails to resolve the problem generates repeat calls, adding to workload and frustrating customers. Most routing systems consider speed only. Using data from an existing call centre, the authors study whether routing can use both speed and problem-solving capability.

Why OR was suitable. Allocation of calls to different agents is an OR routing problem; where the 'cost' of an assignment includes the future work created by unresolved calls.

Mathematical model. Agents (or agent groups) i have mean handling time $1/\mu_i$ and resolution probability r_i for each call type. An unresolved call returns after a delay with probability $1 - r_i$, so the effective arrival rate is $\lambda_{\text{eff}} = \lambda_{\text{new}} / (\text{average resolution})$. Routing policies rank idle agents by speed, by resolution, or by a combined index.

Effective load: $\lambda_{\text{eff}} = \lambda_{\text{new}} \cdot \sum_{k \geq 0} (1 - \bar{r})^k = \lambda_{\text{new}} / \bar{r}$

Index routing: choose idle agent i maximising $I_i = w \cdot r_i + (1 - w) \cdot \mu_i / \mu_{\text{max}}$

Performance: $E[W](\text{policy})$, first-call resolution FCR(policy), repeat-call rate

Assumptions. The traits of agents are assumed to be fixed and estimable from data; callback events are assumed to have a known delay distribution; and routing is assumed to be carried out by groupings of agents.

Solution method and validation. (i) The authors estimate heterogeneity of the agents using call-centre data; (ii) from the call centre data they derive a queueing and simulation model with re-dialling; (iii) they test different rules for routing under different load conditions.

Findings. Routing rules that account for agents' resolution capability can improve call resolution and reduce total workload, possibly without longer waiting times. The best rule depends on the load.

Strengths. Empirical, and builds service quality into the routing objective, both of which matter to management.

Limitations. Agent performance may change with workload and learning, and the policy class is limited to simple indices.

Extensions. Adaptive, learning-based routing and resolution-aware routing combined with staffing decisions.

5) P25 – Chan, Koole & L'Ecuyer (2014) [32]: “Dynamic call center routing policies using call waiting and agent idle times”

Domain	Multi-skill call centres
Problem type	Dynamic routing (real-time call-to-CSR assignment) with parameterised policies
Objective	Minimise staffing-related cost or maximise service levels under a given staffing
Technique	Weight-based routing (WR) using waiting and idle times; weights optimised by simulation-based metaheuristic search
Tools	Discrete-event simulation (ContactCenters); genetic-algorithm style optimiser

Real-life problem. Most centres use static priority routing, in which each agent group has a fixed order of call types. Static rules cannot cope when one queue floods while agents elsewhere sit idle. The authors propose dynamic rules simple enough to implement in commercial routers.

Why OR was suitable. Optimal dynamic routing is an intractable Markov decision problem at realistic scales, so a simulation-optimised parameterised policy offers a feasible OR solution.

Mathematical model. When an agent of group i becomes free, the router selects the waiting call type k that maximises a score based on the longest waiting time w_k ; when a call of type k arrives, it selects the idle group i that maximises a score based on the longest idle time v_i . Weights and offsets (a_{ik} , b_{ik}) are decision variables.

Agent-free rule: $k^* = \arg\max_k \{ a_{ik} \cdot w_k + b_{ik} \}$

Call-arrival rule: $i^* = \arg\max_i \{ a'_{ik} \cdot v_i + b'_{ik} \}$

Optimise weights: $\min_{\{a,b\}} \text{Cost}(\text{staffing})$ or $\max_{\{a,b\}} \sum_k SL_k(a, b)$ (evaluated by simulation)

Assumptions. Fixed or co-optimised staffing levels. Stationary or piecewise-stationary arrival distributions. Simulation model is a sufficiently detailed representation of the centre.

Solution method and validation. Weights are adjusted via a simulation-based heuristic search and the obtained policies are benchmarked against static priorities and a variety of other dynamic rules on different multi-skill configurations.

Findings. The weight-based routing performs better than static priority routing. It can meet target service levels with fewer agents and is resilient to moderate load variations.

Strengths. Practical, implementable dynamic rules; large gains for low complexity.

Limitations. Heavy computation for weight tuning; no guarantee of optimality; agent fairness not explicitly taken into account.

Extensions. Reinforcement learning for routing; joint optimisation with staffing and fairness constraints.

F. Worked Numerical Example

Two core techniques from the review are applied to an illustrative, hypothetical case.

1) Assigning five CSRs to five queues (Hungarian method)

Five CSRs (A–E) are to be assigned to five queues – Billing, Technical, Returns, Account and Complaints. Table III gives the expected handling time (minutes) of each CSR on each queue; total handling time is to be minimised.

Table III: Handling time c_{ij} (minutes)

CSR	Bill.	Tech.	Ret.	Acc.	Comp.
A	9	12	8	10	14
B	11	7	10	12	9
C	8	10	13	7	11
D	12	9	9	11	8
E	10	11	7	9	12

Step 1 – row reduction: subtract the row minima (8, 7, 7, 8, 7). **Step 2 – column reduction:** only the Billing column lacks a zero, so its minimum (1) is subtracted, giving Table IV.

Table IV: Reduced cost matrix

CSR	Bill.	Tech.	Ret.	Acc.	Comp.
A	0	4	0	2	6
B	3	0	3	5	2

CSR	Bill.	Tech.	Ret.	Acc.	Comp.
C	0	3	6	0	4
D	3	1	1	3	0
E	2	4	0	2	5

Step 3 – optimality test: B–Technical, D–Complaints and E–Returns are forced (single zeros in their rows), leaving A–Billing and C–Account. Five independent zeros exist, so the assignment is optimal, and the total reduction ($37 + 1$
 $= 38$) equals the optimal cost (Table V).

Table V: Optimal CSR–queue assignment

CSR	Queue	Time (min)
A	Billing	9
B	Technical	7
C	Account	7
D	Complaints	8
E	Returns	7
Total		38

Python verification and enumeration of all 120 assignments confirm a unique optimum of 38 minutes (next best: 40). The diagonal assignment costs 52 minutes, so optimisation reduces total handling time by about 27%.

2) Staffing one queue (Erlang C)

The Billing queue receives 120 calls per hour with a 5-minute mean handling time ($R = 10$ Erlangs); the target is 80% of calls answered within 20 seconds. Table VI applies the formulas of Section II-B.

Table VI: Erlang-C staffing analysis ($R = 10$)

N	Occup.	P(wait)	SL(20 s)	ASA (s)
11	90.9%	0.682	36.2%	204.6
12	83.3%	0.449	60.7%	67.4
13	76.9%	0.285	76.6%	28.5
14	71.4%	0.174	86.7%	13.1
15	66.7%	0.102	92.7%	6.1
16	62.5%	0.057	96.2%	2.9

Fourteen CSRs is the minimum meeting the target, matching the square-root rule with $\beta \approx 1.26$. Raising staff from 11 to 14 (27%) cuts the average speed of answer from over three minutes to 13 seconds – the nonlinearity that motivates Stream A. Repeating this per queue and interval yields the requirements that Streams B–D cover with shifts.

G. Comparative Synthesis

Objectives and constraints. Three objective families recur: cost minimisation (P1, P2, P4, P10–P20), constraint satisfaction – minimum headcount subject to service targets (P6, P9, P12–P15) – and value or quality

maximisation (P22, P24, P25). Constraints fall into service-level, set-covering, work-rule and skill-eligibility groups. Table VII compares modelling features.

Table VII: Modelling features of the reviewed articles

ID	Model	Demand	Skills	Shifts	Routing
P1	Network flow	Det.	Single	Yes	No
P2	IP	Det.	Single	Yes	No
P3	Survey	Det.	Both	Yes	Yes
P4	Asymptotic	Det.	Single	No	No
P5	Approximation	Det.	Single	No	No
P6	Asymptotic	Det.	Single	No	No
P7	Local search+sim.	Det.	Multi	Yes	Impl.
P8	Markov approx.	Det.	Two	No	Fixed
P9	Heuristic+sim.	Det.	Multi	No	Fixed
P10	LP/IP	Det.	Multi	Yes	Impl.
P11	MIP	Det.	Single	Yes	No
P12	IP+sim. cuts	Det.	Single	Yes	No
P13	ACCPM+sim.	Det.	Single	Yes	No
P14	IP+sim. cuts	Det.	Multi	No	Fixed
P15	IP+sim. cuts	Det.	Multi	Yes	Fixed
P16	IP+randomis.	Det.	Single	Yes	No
P17	Chance-constr.	Stoch.	Multi	No	Yes
P18	2-stage SP	Stoch.	Single	Yes	No
P19	Joint CC	Stoch.	Single	Yes	No
P20	2-stage robust	Robust	Flexible	Yes	Recourse
P21	Queue/MDP	Det.	Two	No	Yes
P22	Assignment+sim.	Det.	Multi	No	Yes
P23	Asymptotic	Det.	Hetero.	No	Yes
P24	Queue+sim.	Det.	Hetero.	No	Yes
P25	Sim. search	Det.	Multi	No	Yes

Det. = known arrival rates; Stoch. = random rates with a distribution; Robust = uncertainty set; Impl. = implicit.

Techniques and software. Early work exploited structure – network flow (P1) and LP relaxations (P2, P10). Queueing asymptotics (P4, P6, P23) produced closed-form rules. From the mid-2000s, fast simulation and commercial MIP solvers such as CPLEX enabled simulation-optimisation (P12–P15, P25), and since 2010 stochastic and robust programming (P17–P20) has addressed forecast uncertainty. Heuristics appear wherever exact methods are too slow (P7, P14, P15, P25). Excel Solver or LINGO suffice for small textbook instances such as the example above.

Validation. Four approaches were observed: comparison with exact solutions or bounds (P4, P8, P10); discrete-event simulation (P7, P9, P12–P17, P22, P24, P25); out-of-sample scenario testing (P17–P20); and calibration on real call-centre data or comparison with practice (P1, P3, P11, P18, P20, P24).

Evolution. Four shifts are evident: from deterministic to stochastic models; from single-skill to multi-skill CSRs;

from sequential to integrated staffing, scheduling and routing (P15, P17, P23); and from pure efficiency to broader effectiveness, including resolution, revenue and agent preferences (P22, P24).

H. Critical Reflection

1) Strengths

- **Theory–practice link:** Erlang staffing, covering-based scheduling and priority routing are embedded in commercial software, and several studies report implementation (P1, P7, P11).
- **Quantified trade-offs:** the cost of service quality (P4, P6), the value of cross-training (P8, P9), the value of integration (P15) and the cost of reliability (P17–P20) are made explicit.
- **Scalability and methodological richness:** asymptotic rules and decomposition methods scale to large centres, while simulation-optimisation and chance-constrained programming add methodological richness.

2) Limitations

However, there are also some drawbacks.

- **Distributional assumptions:** many models assume Poisson arrivals with known rates and exponential handling times, whereas real data often show over-dispersion and log-normal handling times.
- **Homogeneous CSRs:** learning, fatigue and individual differences are largely ignored (partly addressed by P23, P24).
- **Sequential decisions and limited behavioural realism:** abandonment, retrials, absenteeism and CSR well-being are included only occasionally, and most models are voice-only.

3) Future research

Future research directions are many, including:

- **Omnichannel and concurrency:** models in which one CSR serves several chats simultaneously.
- **Human–AI collaboration:** staffing when chatbots resolve simple contacts and transfer complex ones to human agents.
- **Learning-based routing:** reinforcement learning to learn the value of each customer–CSR match under service constraints.
- **Fairness and well-being:** workload balance and preference satisfaction in routing and scheduling objectives.

- **Emerging markets:** multilingual, night-shift and high-attrition settings in outsourced centres in India and elsewhere.

V. CONCLUSION

This review analysed twenty-five peer-reviewed articles that apply OR to CSR assignment, ranging from network-flow scheduling of telephone operators, to robust and simulation-based models that incorporated staffing, scheduling and routing. Three features stand out. First, because of the queueing system, staffing is nonlinear and therefore quantitative rules cannot be proportional and OR models are needed; second, there appears to be huge value in flexibility (for example, cross-trained CSRs, integrated decisions, recourse), and even small degrees of flexibility provide large benefits; and third, explicitly modelling uncertainty is also critically important in contact centre contexts and assuming deterministic environments lead to systematic suboptimal performance.

The example illustrated that using a Hungarian assignment method to allocate CSRs to queues plus an Erlang-C model for staffing can determine which queues should be assigned to which CSRs, and how many CSRs to assign to each queue, respectively.

In the omnichannel and AI-enabled contact centre future, these OR principles will remain relevant: formulation, structural properties, validation, and explicit trade-offs.

DECLARATIONS

Conflict of interest: The author(s) declare no conflict of interest. **Funding:** The author(s) received no external funding for this research. **Data availability:** This manuscript utilises published literature and the data for the numerical example are fully presented in Tables III–VI.

REFERENCES

- [1] N. Gans, G. Koole, and A. Mandelbaum, "Telephone call centers: Tutorial, review, and research prospects," *Manufacturing & Service Operations Management*, vol. 5, no. 2, pp. 79–141, 2003.
- [2] Z. Aksin, M. Armony, and V. Mehrotra, "The modern call center: A multi-disciplinary perspective on operations management research," *Production and Operations Management*, vol. 16, no. 6, pp. 665–688, 2007.
- [3] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1–2, pp. 83–97, 1955.
- [4] A. T. Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier, "Staff scheduling and rostering: A review of applications, methods and models," *European Journal of Operational Research*, vol. 153, no. 1, pp. 3–27, 2004.

- [5] J. Van den Bergh, J. Beliën, P. De Bruecker, E. Demeulemeester, and L. De Boeck, "Personnel scheduling: A literature review," *European Journal of Operational Research*, vol. 226, no. 3, pp. 367–385, 2013.
- [6] M. Defraeye and I. Van Nieuwenhuysse, "Staffing and scheduling under nonstationary demand for service: A literature review," *Omega*, vol. 58, pp. 4–25, 2016.
- [7] S. Halfin and W. Whitt, "Heavy-traffic limits for queues with many exponential servers," *Operations Research*, vol. 29, no. 3, pp. 567–588, 1981.
- [8] S. Borst, A. Mandelbaum, and M. I. Reiman, "Dimensioning large call centers," *Operations Research*, vol. 52, no. 1, pp. 17–34, 2004.
- [9] G. B. Dantzig, "A comment on Edie's 'Traffic delays at toll booths'," *Journal of the Operations Research Society of America*, vol. 2, no. 3, pp. 339–341, 1954.
- [10] M. Segal, "The operator-scheduling problem: A network-flow approach," *Operations Research*, vol. 22, no. 4, pp. 808–823, 1974.
- [11] G. M. Thompson, "Labor staffing and scheduling models for controlling service levels," *Naval Research Logistics*, vol. 44, no. 8, pp. 719–740, 1997.
- [12] L. V. Green, P. J. Kolesar, and W. Whitt, "Coping with time-varying demand when setting staffing requirements for a service system," *Production and Operations Management*, vol. 16, no. 1, pp. 13–39, 2007.
- [13] A. Mandelbaum and S. Zeltyn, "Staffing many-server queues with impatient customers: Constraint satisfaction in call centers," *Operations Research*, vol. 57, no. 5, pp. 1189–1205, 2009.
- [14] A. Fukunaga, E. Hamilton, J. Fama, D. Andre, O. Matan, and I. Nourbakhsh, "Staff scheduling for inbound call and customer contact centers," *AI Magazine*, vol. 23, no. 4, pp. 30–40, 2002.
- [15] R. A. Shumsky, "Approximation and analysis of a call center with flexible and specialized servers," *OR Spectrum*, vol. 26, no. 3, pp. 307–330, 2004.
- [16] R. B. Wallace and W. Whitt, "A staffing algorithm for call centers with skill-based routing," *Manufacturing & Service Operations Management*, vol. 7, no. 4, pp. 276–294, 2005.
- [17] S. Bhulai, G. Koole, and A. Pot, "Simple methods for shift scheduling in multiskill call centers," *Manufacturing & Service Operations Management*, vol. 10, no. 3, pp. 411–420, 2008.
- [18] D. C. Dietz, "Practical scheduling for call center operations," *Omega*, vol. 39, no. 5, pp. 550–557, 2011.
- [19] J. Atlason, M. A. Epelman, and S. G. Henderson, "Call center staffing with simulation and cutting plane methods," *Annals of Operations Research*, vol. 127, pp. 333–358, 2004.
- [20] J. Atlason, M. A. Epelman, and S. G. Henderson, "Optimizing call center staffing using simulation and analytic center cutting-plane methods," *Management Science*, vol. 54, no. 2, pp. 295–309, 2008.
- [21] M. T. Cezik and P. L'Ecuyer, "Staffing multiskill call centers via linear programming and simulation," *Management Science*, vol. 54, no. 2, pp. 310–323, 2008.
- [22] A. N. Avramidis, W. Chan, M. Gendreau, P. L'Ecuyer, and O. Pisacane, "Optimizing daily agent scheduling in a multiskill call center," *European Journal of Operational Research*, vol. 200, no. 3, pp. 822–832, 2010.
- [23] A. Ingolfsson, F. Campello, X. Wu, and E. Cabral, "Combining integer programming and the randomization method to schedule employees," *European Journal of Operational Research*, vol. 202, no. 1, pp. 153–163, 2010.
- [24] I. Gurvich, J. Luedtke, and T. Tezcan, "Staffing call centers with uncertain demand forecasts: A chance-constrained optimization approach," *Management Science*, vol. 56, no. 7, pp. 1093–1115, 2010.
- [25] T. R. Robbins and T. P. Harrison, "A stochastic programming model for scheduling call centers with global service level agreements," *European Journal of Operational Research*, vol. 207, no. 3, pp. 1608–1619, 2010.
- [26] M. Excoffier, C. Gicquel, and O. Jouini, "A joint chance-constrained programming approach for call center workforce scheduling under uncertain call arrival forecasts," *Computers & Industrial Engineering*, vol. 96, pp. 16–30, 2016.
- [27] S. Mattia, F. Rossi, M. Servilio, and S. Smriglio, "Staffing and scheduling flexible call centers by two-stage robust optimization," *Omega*, vol. 72, pp. 25–37, 2017.
- [28] N. Gans and Y.-P. Zhou, "Call-routing schemes for call-center outsourcing," *Manufacturing & Service Operations Management*, vol. 9, no. 1, pp. 33–50, 2007.
- [29] M. E. Sisselman and W. Whitt, "Value-based routing and preference-based routing in customer contact centers," *Production and Operations Management*, vol. 16, no. 3, pp. 277–291, 2007.
- [30] M. Armony and A. Mandelbaum, "Routing and staffing in large-scale service systems: The case of homogeneous impatient customers and heterogeneous servers," *Operations Research*, vol. 59, no. 1, pp. 50–65, 2011.
- [31] V. Mehrotra, K. Ross, G. Ryder, and Y.-P. Zhou, "Routing to manage resolution and waiting time in call centers with heterogeneous servers," *Manufacturing & Service Operations Management*, vol. 14, no. 1, pp. 66–81, 2012.
- [32] W. Chan, G. Koole, and P. L'Ecuyer, "Dynamic call center routing policies using call waiting and agent idle times," *Manufacturing & Service Operations Management*, vol. 16, no. 4, pp. 544–560, 2014.