Staffing for multi-skill call centers

Thanos Avramidis
DIRO, Université de Montréal, Canada

- A method for multi-skill staffing
- Random arrival rates
- Extensions

Single-period multi-skill staffing problem

$\mathcal{N} = \{1, \ldots, n\}$ = set of call (contact) classes (types)

$\mathcal{M} = \{1, \ldots, m\}$ = set of agent types

$\mathcal{S}_i \subseteq \mathcal{N}$ = type-i skill set

Decision variables: $\mathbf{y} = (y_1, \ldots, y_m)^\mathsf{T}$, where y_i is the number of agents of type i.

Costs: $\mathbf{c} = (c_1, \ldots, c_m)$ where c_i = cost of an agent of type i.

Virtual queue time: the time a customer with infinite patience (i.e., who never abandons) must wait in queue

Service level (SL) for call type j:

$$g_j(\mathbf{y}) = \frac{\text{expected \# calls arrived whose virtual queue time is} < \tau_j}{\text{expected \# calls arrived}}$$

for some constant τ_j.

$g(\mathbf{y}) =$ aggregate SL, defined analogously for given time limit τ.

The functions $g_\bullet$ depend on the routing policy. For planning purposes, it may be necessary to work with a fixed routing policy; we do so here.

$$\begin{aligned} \min \quad & \mathbf{c}^\mathsf{T}\mathbf{y} = \textstyle\sum_{i=1}^{m} c_i y_i \\ \text{subject to} \quad & g_j(\mathbf{y}) \geq l_j \qquad \text{for all } j, \\ & g(\mathbf{y}) \geq l, \\ & \mathbf{y} \geq 0, \text{ and integer.} \end{aligned} \tag{P2}$$

Past work

Cezik and L'Ecuyer (2005)

Bhulai et. al. (2005)

Overflow routing

Station i: the ensemble of agents of type i

$\mathcal{R}_j = \{R_j(1), R_j(2), \ldots, R_j(m_j)\}$: ordered list of stations that can handle call type j

$r(i,j)$: the rank of i in the list $\mathcal{R}_j$; that is, $R_j(r(i,j)) = i$.

Overflow routing policy:
- upon arrival, a class-j call is assigned to an agent in the highest-rank (lowest-index) station in $\mathcal{R}_j$ with an available agent (say, i^*) or else is placed in queue.

Overflow routing, loss system: Past work

Loss system: Calls type j that overflow from the last station, $R_j(m_j)$, are lost.

Optimality of specialist-first routing for special cases of skill sets (Chevalier et. al. 2005, Ormeci 2004)

Exponential decomposition: analyze each station separately, based on 1-moment approximation of overflow process (Koole and Talim 2000) (KT00).

Equivalent Random Method: 2-moment approximation of overflow process, i.e., rate and peakedness (Hayward 1981, Wolff 1989, Chevalier et. al. 2004).

Hyper-exponential decomposition (Franx et. al. 2005)

Overflow routing in a delay system

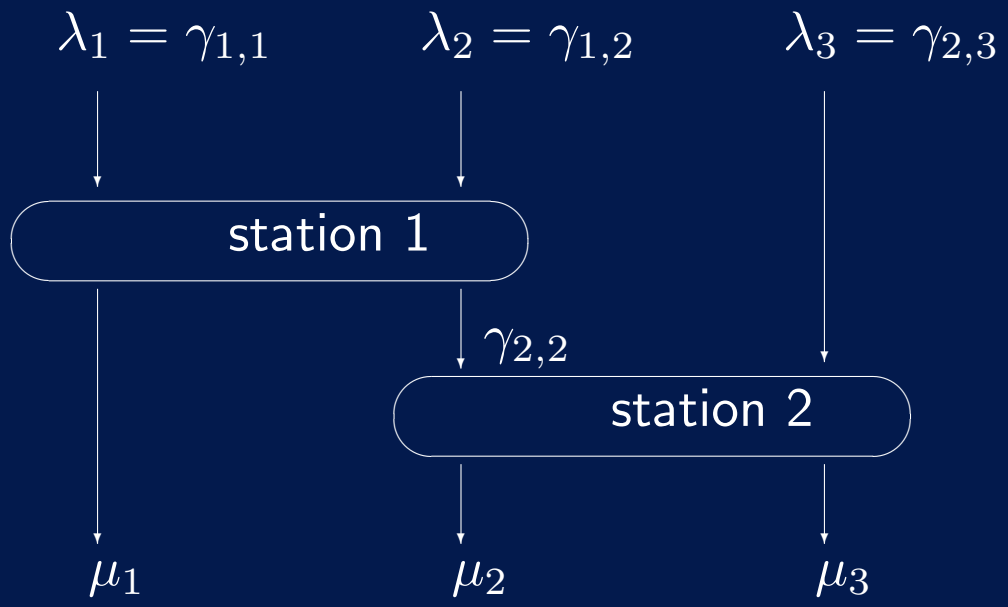
Overflow-or-wait-at-last-station policy:
- each queued call of type j must be served at the last station on its list: $L_j = R_j(m_j)$
- FIFO across multiple call classes

Captures overflow routing and is *conceptually* simple to analyze.
Conceptually not very attractive, because it allows agent idleness.

For each station i, define:
$\mathcal{L}_i = \{j : i = R_j(\ell)$ for some $\ell \neq m_j\}$ =set of classes that overflow at i
$\mathcal{D}_i = \{j : i = R_j(m_j) =$ set of classes that wait at i

Partition stations into three types:
loss-delay: $\mathcal{M}_{\mathrm{LD}} = \{i : \mathcal{L}_i \neq \emptyset, \mathcal{D}_i \neq \emptyset\}$
pure-loss: $\mathcal{M}_{\mathrm{L}} = \{i : \mathcal{L}_i \neq \emptyset, \mathcal{D}_i = \emptyset\}$
pure-delay: $\mathcal{M}_{\mathrm{D}} = \{i : \mathcal{L}_i = \emptyset, \mathcal{D}_i \neq \emptyset\}$

Illustration



$\mathcal{L}_1 = \{2\}$, $\mathcal{D}_1 = \{1\}$ $\Rightarrow$ station 1 is of loss-delay type.
$\mathcal{L}_2 = \emptyset$, $\mathcal{D}_2 = \{2, 3\}$ $\Rightarrow$ station 2 is of pure-delay type.

Analysis of a loss-delay station with abandonment

1. Work with two input streams:
loss (call types in $\mathcal{L}_i$):

delay (call types in $\mathcal{D}_i$):
Assume they are independent Poisson processes.

2. Assume:
- service and time-to-abandonment are exponential r.v.'s - two streams have the same mean service time and mean time to abandonment

3. One-dimensional Birth and death (B-D) model of the loss-delay station.

4. Conditional on encountering s customers in queue, distribution of virtual time in queue is known analytically (Riordan 1962, Koole 2003).

5. PASTA yields the unconditional distribution of virtual time in queue

Approximate performance measures for a loss-delay station:
$B_{\mathrm{A}}(s, \lambda_{\mathrm{L}}, \lambda_{\mathrm{D}}, \mu_{\mathrm{L}}, \mu_{\mathrm{D}}, \eta, c) =$ blocking probability
$P(W > \tau) = D_{\mathrm{A}}(\tau; s, \lambda_{\mathrm{L}}, \lambda_{\mathrm{D}}, \mu_{\mathrm{L}}, \mu_{\mathrm{D}}, \eta, c)$,

where

W = virtual waiting time

λ_{L} = loss stream arrival rate

μ_{L} = loss stream service rate

λ_{D} = delay stream arrival rate

μ_{D} = delay stream service rate

η = time-to-abandonment rate

c = queue capacity

Loss-delay approximation

λ_j = class-j arrival rate
$\mu_{i,j}$ = service rate for class j at station i
η_j = class-j time-to-abandonment rate

$p(i,j)$ = station immediately preceeding station i in the routing of type-j calls (exists whenever $r(i,j) > 1$)

Generalized arrival to station i: encompasses both:
- an exogenous arrival (i.e., $r(i,j) = 1$)
- an overflow to i.

We will approximate:
$\gamma_{i,j}$ = generalized arrival rate for all i and $j \in \mathcal{S}_i$
B_i = blocking probability, whenever i is a pure-loss or loss-delay station

Loss-Delay approximation with Abandonment (LDA):

$$\gamma_{i,j} = \begin{cases} \lambda_j & \text{for all } i \in \mathcal{M},\ j \in \mathcal{S}_i, \text{ and } r(i,j) = 1 \\ \gamma_{p(i,j),j} B_{p(i,j)} & \text{for all } i \in \mathcal{M},\ j \in \mathcal{S}_i, \text{ and } r(i,j) > 1 \end{cases} \tag{1}$$

$$\gamma_{i,\mathrm{L}} = \sum_{j \in \mathcal{L}_i} \gamma_{i,j}, \quad \frac{1}{\mu_{i,\mathrm{L}}} = \sum_{j \in \mathcal{L}_i} \frac{\gamma_{i,j}}{\gamma_{i,\mathrm{L}}} \frac{1}{\mu_{i,j}}, \quad i \in \mathcal{M}_{\mathrm{LD}} \cup \mathcal{M}_{\mathrm{L}}, \tag{2}$$

$$\gamma_{i,\mathrm{D}} = \sum_{j \in \mathcal{D}_i} \gamma_{i,j}, \quad \frac{1}{\mu_{i,\mathrm{D}}} = \sum_{j \in \mathcal{D}_i} \frac{\gamma_{i,j}}{\gamma_{i,\mathrm{D}}} \frac{1}{\mu_{i,j}}, \quad \frac{1}{\eta_i} = \sum_{j \in \mathcal{D}_i} \frac{\gamma_{i,j}}{\gamma_{i,\mathrm{D}}} \frac{1}{\eta_j}, \quad i \in \mathcal{M}_{\mathrm{LD}} \cup \mathcal{M}_{\mathrm{D}} \tag{3}$$

$$B_i = B_{\mathrm{A}}\left(y_i, \gamma_{i,\mathrm{L}}, \gamma_{i,\mathrm{D}}, \mu_{i,\mathrm{L}}, \mu_{i,\mathrm{D}}, \eta_i, c_i\right), \quad i \in \mathcal{M}_{\mathrm{LD}} \cup \mathcal{M}_{\mathrm{L}}, \tag{4}$$

where: $c_i = \max(\lceil \psi \sqrt{y_i} \rceil, 10)$ and ψ is a queue-size control parameter.

Similar to KT00, but blocking probability function B_{A} differs from theirs in stations having a delay stream.

Assume solution to (1)–(4) is available.

$L_j = \mathcal{R}_j(m_j) =$ last station in class-j routing

Class-j service level:

$$g_j(\mathbf{y};\tau) = 1 - \frac{\gamma_{L_j,j}}{\lambda_j} D_{L_j}(\tau), \quad \tau > 0, \quad j \in \mathcal{N}, \tag{5}$$

where

$$D_i(\tau) = P(W_i > \tau) = D_{\mathrm{A}}\left(\tau; y_i, \gamma_{i,\mathrm{L}}, \gamma_{i,\mathrm{D}}, \mu_{i,\mathrm{L}}, \mu_{i,\mathrm{D}}, \eta_i, c_i\right), \quad \tau > 0,\ i \in \mathcal{M}_{\mathrm{LD}} \cup \mathcal{M}_{\mathrm{D}}.$$

A view of solution approaches to (P2)

Stage 1 combines:

Optimizer (over the space of staffing vectors)

Evaluator of performance (service level, expected waiting time, loss rate due to abandonment)
A solution is *E-(in)feasible* whenever it is declared (in)feasible for (P2) by the evaluator.

Stage 2: Adjustor Adjusts the optimizer's incumbent for:
infeasibility
or
cost reduction
Require more accurate evaluator than Stage 1; local search requiring few evaluations.

Cezik and L'Ecuyer (2005): Cutting Plane / Simulation.

Neighborhood search: Elements

$\mathbf{y}^{(k)} =$ the incumbent solution, where k is a counter of incumbents

$q =$ a positive integer *move size*

Remove neighborhood: $\mathcal{Y}_1(\mathbf{y}^{(k)}, q) = \{\mathbf{y} : \mathbf{y} = \mathbf{y}^{(k)} - q\mathbf{e}_i, i \in \mathcal{C}\}$, where $\mathcal{C} = \{i : y_i^{(k)} \geq q\}$.

Pivot: some $i \in \mathcal{T} = \{i : y_i^{(k)} \geq q\}$.

Switch neighborhood:
$\mathcal{Y}_2 = \mathcal{Y}_2(\mathbf{y}^{(k)}, q, i) = \{\mathbf{y} : \mathbf{y} = \mathbf{y}^{(k)} - q\mathbf{e}_i + q\mathbf{e}_j,\ y_i^{(k)} \geq q, j \in \mathcal{M}\}$,
where $\mathbf{e}_i$ is the vector with 1 on i-the coordinate, 0 elsewhere.

Cost-reducing switch neighborhood:
$\mathcal{Y}_2^-(\mathbf{y}^{(k)}, q, i) = \{\mathbf{y} : \mathbf{y} = \mathbf{y}^{(k)} - q\mathbf{e}_i + q\mathbf{e}_j,\ y_i^{(k)} \geq q,\ c_j < c_i\} \subseteq \mathcal{Y}_2$.

Search overview: A search integrator selects a neighborhood to consider. Given a neighborhood, either we determine it contains no cost-reducing and E-feasible members and record its identity, or else we select the next incumbent. Then return control to the integrator.

Agent removal

$q^*(k)$ = smallest q such that $\mathcal{Y}_1(\mathbf{y}^{(k)}, q)$ contains no E-feasible solutions.

Procedure `Remove`$(\mathbf{y}^{(k)}, q, q^*(k))$ {

Evaluate all candidates in $\mathcal{Y}_1(\mathbf{y}^{(k)}, q)$
If (at least one candidate is E-feasible)
    Incumbent selection via best-candidate criterion:
    Minimize the ratio:

$$\frac{\hat{g}(\mathbf{y}^{(k)}) - \hat{g}(\mathbf{y})}{\mathbf{c}^\mathsf{T}\mathbf{y}^{(k)} - \mathbf{c}^\mathsf{T}\mathbf{y}}$$

    over $\mathbf{y} \in \mathcal{Y}_1$ and E-feasible, where $\hat{g}$ = approximate service-level function
Else
    $\mathcal{Y}_1$ contains no E-feasible solutions
    Update: $q^*(k) \leftarrow \min(q^*(k), q)$.
End if
}

Agent switching

A pivot i is $(\mathbf{y}, q)$-infeasible whenever $\mathcal{Y}_2^-(\mathbf{y}, q, i)$ is either empty or all its members are E-infeasible ($q \geq 1$)

$q_i^*(k)$ = the smallest q such that i is known to be a $(\mathbf{y}^{(k)}, q)$-infeasible pivot

Procedure `Switch`$(\mathbf{y}^{(k)}, q, (q_i^*(k))_{i \in \mathcal{M}})$ {

Select a pivot P randomly, uniformly over $\mathcal{P} = \mathcal{T} \cap \{i : q_i^*(k) > q\}$
Evaluate all candidates in $\mathcal{Y}_2^-(\mathbf{y}^{(k)}, q, P)$.
If (at least one candidate is E-feasible)
    Select the next incumbent by best-candidate criterion
Else
    Pivot P has been proven to be $(\mathbf{y}^{(k)}, q)$-infeasible
    Update: $q_P^*(k) \leftarrow \min(q_P^*(k), q)$
End if
}

Search integration

Given: $\mathbf{y}^{(k)}$, $q^*(k)$, $q_i^*(k)$ (all i)

Procedure `Search`

Select a move size $q \in \{1, 2, \ldots, q_{\max}\}$ where $q_{\max} = \max_i x_i^{(k)}$
If $(q < q^*(k))$
    Call `Remove` with move size q
Else
    Call `Switch` with move size q
End if

Normal termination: $\Leftrightarrow$ All possible pivots are $(\mathbf{y}^{(k)}, 1)$-infeasible.

Early termination: `Search` is terminated before having met the above condition (e.g., via a time limit).

Search Properties

Problem (P2*): Replace all functions $g_\bullet$ in (P2) by the evaluator's estimates $\tilde{g}_\bullet$.

N_{E}: (random) number of evaluations until normal termination.

Assumption: if at incumbent ℓ some approximate service level j decreases with a switch of size q_1, then it decreases by at least as much for all larger move sizes:

$$\tilde{g}_j(\mathbf{x}^{(\ell)}-q_1\mathbf{e}_i+q_1\mathbf{e}_k) < \tilde{g}_j(\mathbf{x}^{(\ell)}),\ q_2 > q_1 \ \Rightarrow \tilde{g}_j(\mathbf{x}^{(\ell)}-q_2\mathbf{e}_i+q_2\mathbf{e}_k) \leq \tilde{g}_j(\mathbf{x}^{(\ell)}-q_1\mathbf{e}_i+q_1\mathbf{e}_k) \tag{6}$$

for any class j and agent types i, k with $c_k < c_i$.

Proposition 3.

1. N_{E} is finite.

2. Let $\mathbf{y}^{(k)}$ be the output of `Search` with normal termination. Then

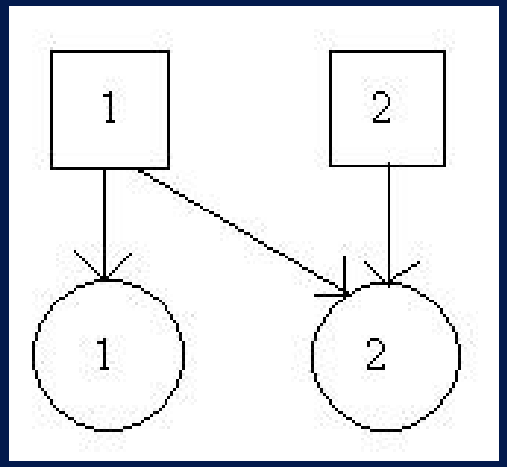
   (a) There exist no (P2*)-feasible vectors in $\mathcal{Y}_1(\mathbf{y}^{(k)}, 1)$.

(b) There exist no (P2*)-feasible cost-reducing vectors (relative to $\mathbf{y}^{(k)}$) in the neighborhood $\cup_{i:y_i^{(k)}\geq 1}\mathcal{Y}_2(\mathbf{y}^{(k)},1,i)$.

(c) If (6) holds for incumbent k, then there exist no (P2*)-feasible cost-reducing vectors in $\cup_{q\geq 1}\cup_{i:y_i^{(k)}\geq q}\mathcal{Y}_2(\mathbf{y}^{(k)},q,i)$.

3. Suppose (6) holds for all incumbents (i.e, all ℓ). Then, in each call to `Switch` with incumbent $\mathbf{y}^{(k)}$ and move size q, the members of $\{i : q_i^*(k)\leq q\}$ are $(\mathbf{y}^{(k)},q)$-infeasible pivots.

Simple multi-skill case: N Design



$\mathbf{\Lambda} = (\Lambda_1, \Lambda_2)$ = Random arrival rate with mean (100,50), coefficient of variation 1/4.
Each marginal distribution discretized to 3 values (low, medium, high) $\Rightarrow$ 9 scenarios

Patience: exponential with mean 3 minutes

$\tau = 20$ seconds

Target SL: 80% for each class

Overflow routing: Class 1 prefers group 1, overflows to group 2 if necessary

Staffing: (82 specialists, 65 generalists (best solution found, for "priority" policy and mean arrival rate)

For this staffing, varied the routing policy (call selection by agents).

Performance with "Priority"

Priority: group 2 gives priority to class-2 queue

ℓ_i = class-i abandonment rate (rate of call loss)
ℓ = aggregate abandonment rate (call loss, both classes)

Probab.	α_1	α_2	SL_1	SL_2	ℓ_1	ℓ_2	ℓ
0.11	76.6	38.3	1.00	1.00	0.0	0.0	0.0
0.11	76.6	49.3	1.00	0.98	0.1	0.3	0.4
0.11	76.6	62.4	0.97	0.78	0.7	3.7	4.4
0.11	98.7	38.3	0.95	0.95	1.4	0.6	2.0
0.11	98.7	49.3	0.83	0.83	4.5	2.4	6.9
0.11	98.7	62.4	0.64	0.57	9.5	7.0	16.5
0.11	124.8	38.3	0.53	0.82	15.2	2.2	17.4
0.11	124.8	49.3	0.32	0.68	23.0	4.4	27.4
0.11	124.8	62.4	0.15	0.47	31.7	8.5	40.2
average	100	50	0.66	0.76	9.5	3.2	12.7

Performance with "longest queue first"

Longest queue first: group 2 serves calls in FIFO order, giving priority to the longest of the two queues

Probab.	α_1	α_2	SL_1	SL_2	ℓ_1	ℓ_2	ℓ
0.11	76.6	38.3	1.00	1.00	0.0	0.0	0.0
0.11	76.6	49.4	1.00	0.97	0.1	0.4	0.5
0.11	76.6	62.4	0.98	0.78	0.4	3.7	4.1
0.11	98.7	38.3	0.98	0.91	0.9	1.0	1.9
0.11	98.7	49.4	0.91	0.71	2.8	4.0	6.8
0.11	98.7	62.4	0.77	0.41	5.9	10.3	16.2
0.11	124.8	38.3	0.69	0.39	9.6	7.9	17.5
0.11	124.8	49.4	0.50	0.20	14.0	13.6	27.6
0.11	124.8	62.4	0.31	0.08	19.1	20.8	39.9
average	100.0	50.0	0.75	0.58	5.9	6.8	12.7

Performance with "single FIFO"

Single FIFO queue: FIFO call selection across classes, by both groups

Probab.	α_1	α_2	SL_1	SL_2	ℓ_1	ℓ_2	ℓ
0.11	76.5	38.3	1.00	1.00	0.0	0.0	0.0
0.11	76.5	49.4	1.00	0.97	0.1	0.4	0.5
0.11	76.5	62.4	0.98	0.76	0.5	4.1	4.6
0.11	98.6	38.3	0.96	0.92	1.2	0.9	2.1
0.11	98.6	49.4	0.88	0.72	3.4	3.6	7.0
0.11	98.6	62.4	0.72	0.41	7.0	9.8	16.8
0.11	124.7	38.3	0.59	0.45	12.6	5.2	17.8
0.11	124.7	49.4	0.40	0.24	18.2	9.8	28.0
0.11	124.7	62.4	0.22	0.09	24.2	16.6	40.8
average	100.0	50.0	0.70	0.59	7.5	5.6	13.1

Performance with "Priority" and positive correlation on Λ

Same 9 arrival-rate scenarios

Preference routing

Simply change the scenario probabilities

Probab.	α_1	α_2	SL_1	SL_2	ℓ_1	ℓ_2	ℓ
0.17	76.6	38.3	1.00	1.00	0.0	0.0	0.0
0.11	76.6	49.4	0.99	0.98	0.1	0.3	0.4
0.06	76.6	62.4	0.97	0.78	0.7	3.7	4.4
0.11	98.7	38.3	0.95	0.96	1.3	0.6	1.9
0.12	98.7	49.4	0.83	0.83	4.4	2.4	6.8
0.11	98.7	62.4	0.64	0.56	9.5	7.1	16.6
0.06	124.8	38.3	0.51	0.81	15.5	2.2	17.7
0.11	124.8	49.4	0.31	0.68	23.3	4.5	27.8
0.17	124.8	62.4	0.15	0.47	31.9	8.5	40.4
average	100.0	50.0	0.63	0.75	10.5	3.4	13.9

2-stage model formulation with re-staffing option

Stage 1. Intermediate planning horizon (a few weeks), high forecast uncertainty
$\mathbf{x} = (x_1 \dots x_m)$ = stage-1 decision = planned staffing.

$\mathbf{\Lambda} = (\Lambda_1 \dots \Lambda_m)$= random arrival rate vector; $F_{\mathbf{\Lambda}}$=known distribution of $\mathbf{\Lambda}$.

Stage 2. Day-of-operation; low forecast uncertainty
- Have $\mathbf{x}$ from stage 1
- λ = observed (constant) arrival rate
- Have Option to increase or decrease staffing

$\mathbf{y} = (y_1 \dots y_m)$ = stage-2 (adjusted) staffing

$g_j(y; \lambda)$ = class-j service level (or other performance measure) when arrival rate vector is λ.

c_i^+ = cost per additional type-i agent

c_i^- = benefit per type-i agent released from duty or sent to other activities

$\mathbf{u}$ = upper bound on new staffing vector

p_j = penalty per unit violation of constraint j (meeting fixed target service-levels may be impossible for some λ (with positive probability).

Stage-2 cost (re-staffing cost + infeasibility penalty):

$f_2(\mathbf{y}; \mathbf{x}, \lambda) = \sum_i c_i^+ (y_i - x_i)^+ - \sum_i c_i^- (x_i - y_i)^+ + \sum_j p_j (l_j - g_j(\mathbf{y}; \mathbf{x}, \lambda))$,
where $x^+ = \max(x, 0)$

Stage-2 problem: For given $\mathbf{x}$ and λ,

$$\begin{array}{rl} \min_{\mathbf{y}} & f_2(\mathbf{y}; \mathbf{x}, \lambda) \\ \text{subject to} & 0 \le \mathbf{y} \le \mathbf{u}, \text{and integer.} \end{array}$$

$f_2^*(\mathbf{x}, \lambda)$ = optimal cost of (P2')

$f_1(\mathbf{x}) = \mathrm{E}_{\mathbf{\Lambda}}[f_2^*(\mathbf{x}, \mathbf{\Lambda})]$ = expected stage-2 cost, ($\mathrm{E}_{\mathbf{\Lambda}}$ is expectation under $F_{\mathbf{\Lambda}}$).

Stage-1 problem:

$$\min \sum_i c_i x_i + f_1(\mathbf{x})$$

Modeling issues in staff planning

1. Poor forecasts of arrival rates $\Rightarrow$ Early planning decisions will often be "off" (service level too high some days, too low on others).

2. How can we build better models of daily operations?

   (a) What are the re-planning options available to managers after an initial staff plan is made, all the way up to the day of operation ?
   What are the costs of reducing and/or increasing staffing? after good forecasts of the arrival rate are available?
   (b) Routing policy constraints versus routing flexibility.

   $\Rightarrow$ Early staffing decisions linked to the re-planning options and costs.

3. Alternative service level definitions?
   Work with "expected low-service costs" instead of service-level constraints ?