Appendix

There are six appendices. Appendix A gives the transition diagram of a small example of the inverted-V system described in Section 3.2. Appendix B includes analysis related to Section 4 and Appendix C analysis related to Section 5. Appendices D and E cover the analysis of the coefficient of variation and correlation coefficient of the inter-overflow time in the pooled overflow system. These results are used in Section 6. Finally, Appendix F generalizes the results of Appendices B and D for systems in which μ_H may be different from μ_L.

A Transition Diagram of a Small Inverted-V System in Section 3.2

Figure 8 shows the transition diagram for a small inverted-V system, with $m_L = 1$ and $m_O = 2$ CSRs, in which calls are preferentially routed to the client's in-house CSR.

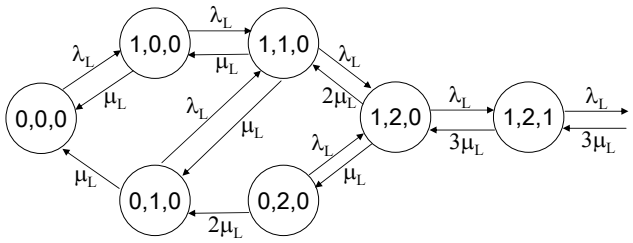

Figure 8: Transition Diagram for Inverted-V with $m_L = 1$ and $m_O = 2$ CSRs

B Analysis of Pooled-Overflow Systems in Section 4

B.1 Optimality of Randomized Threshold-Reservation Policies in Section 4.1

We defer to Appendix F.1 the proof that, among type-H priority policies, there exist stationary, type-H work-conserving policies that maximize the processing throughput of type-L calls in house. (Appendix F covers the general case in which μ_H may be different from μ_L, of which $\mu_H = \mu_L$ – studied here – is a special case.) Here, we make use of Appendix F.1's general result, and we consider only stationary, type-H work-conserving, type-H priority policies. We then prove Theorem 1's result: among these policies, there exist so-called "randomized threshold reservation" policies that are optimal.

Before we begin, we note that, in Section 4.1, we use Little's Law to state the service-level constraint in the pooled-overflow system in terms of the average number of calls in queue, rather than average delay in queue. In fact, the NLP formulation (3)–(6) can actually accommodate a more general class of occupancy-based constraint. In particular, in the service-level constraint (5), the delay-cost function associated with queue-length q, $d(q)$, need only satisfy the following assumptions:

i) $d(0) = 0$ and $d(\bar{q})$ is nondecreasing in $\bar{q}$; ii) $\sup_{\bar{q}} d(\bar{q}) > D^*$; and iii) $\tilde{d}(\alpha) \stackrel{\text{def}}{=} \sum_{\bar{q}=0}^{\infty} \alpha^{\bar{q}}\, d(\bar{q}) < \infty$ for all $\alpha \in (0,1)$. For more on these conditions, please see the statement of Assumption 1 in Appendix F.1.

We now proceed to prove Theorem 1. We begin by noting that, in principal, we could use substitution to eliminate ξ_{m_I} and constraint (4), which defines it, from the NLP formulation (3)–(6). We choose to keep it because the extra variable facilitates our analysis of the problem.

In particular, the inclusion of ξ_{m_I} in the service-level constraint (5) highlights the fact that, for fixed ρ, it is the value of ξ_{m_I} that uniquely determines the constraint's left-hand-side. The following lemma shows that there may be many sets of p_ss from which a given ξ_{m_I} may be obtained:

Lemma 1

Suppose there is a $p \in [0,1]^{m_I}$ for which $0 < p_i, p_{i+1} < 1$ and $0 \le i \le m_I - 2$. Then there also exists a $p' \in [0,1]^{m_I} \neq p$ for which $p'_j = p_j$ for all $j \notin \{i, i+1\}$ and $\xi_{m_I}(p') = \xi_{m_I}(p)$. Furthermore, either $p'_i > p_i$ and $p'_{i+1} < p_{i+1}$ or $p'_i < p_i$ and $p'_{i+1} > p_{i+1}$.

Proof

Let $h_i = \frac{(i+1)\mu}{\lambda_H + \lambda_L p_i}$ and $l_{j,k} = \prod_{i=j}^{k} h_i$. Whenever the range is empty, we define the product to be 1 and the summation to be 0. Then (4) can be written as

$$\xi_{m_I} \;=\; \left[\sum_{s=0}^{m_I-1} l_{s,m_I-1} \;+ \frac{1}{1-\rho} \right]^{-1} ,$$

Note that if $X \stackrel{\text{def}}{=} \sum_{s=0}^{m_I-1} l_{s,m_I-1}$ remains constant, so will ξ_{m_I}. Substituting, in turn, for h_i, we have

$$X \;=\; \sum_{s=0}^{i} l_{s,i-1} h_i h_{i+1} l_{i+2,m_I-1} \;+\; h_{i+1} l_{i+2,m_I-1} \;+\; \sum_{s=i+2}^{m_I-1} l_{s,m_I-1} \,.$$

Solving for h_{i+1} yields

$$h_{i+1} \;= \frac{X \;-\; \sum_{s=i+2}^{m_I-1} l_{s,m_I-1}}{h_i \sum_{s=0}^{i} l_{s,i-1} l_{i+2,m_I-1} \;+\; l_{i+2,m_I-1}} \,.$$

For a given X, we can implicitly differentiate h_{i+1} with respect to h_i to yield

$$\frac{\partial\, h_{i+1}}{\partial\, h_i} \;= -\, \frac{\left(X \;-\; \sum_{s=i+2}^{m_I-1} l_{s,m_I-1}\right) \left(\sum_{s=0}^{i} l_{s,i-1} l_{i+2,m_I-1}\right)}{\left(h_i \sum_{s=0}^{i} l_{s,i-1} l_{i+2,m_I-1} \;+\; l_{i+2,m_I-1}\right)^2} \le 0 \,.$$

Thus, as we increase (decrease) h_i by an infinitesimal amount, it is always possible for us to simultaneously decrease (increases) h_{i+1} to maintain a constant X.

Since both $\partial h_i / \partial p_i < 0$ and $\partial h_{i+1} / \partial p_{i+1} < 0$ we conclude that, equivalently, for a given $0 < p_i, p_{i+1} < 1$ and X, a decrease (increase) in p_i can be compensated for with a simultaneous increase (decrease) in p_{i+1}. □

Therefore, given a set of routing probabilities p, with elements p_i and p_{i+1} that are interior, we can construct an alternative set p', with different p'_i and p'_{i+1}, that obtains the same ξ_{m_I}. Furthermore, p'_i and p'_{i+1} move away from p_i and p_{i+1} in opposite directions.

Now suppose two solutions, p and p', yield the same ξ_{m_I}. Which has the higher objective function value? To answer the question, we again consider two solutions, p and p', as defined in Lemma 1.

Lemma 2

Suppose there are p and p' defined as in Lemma 1. If $p'_i > p_i$, or equivalently $p'_{i+1} < p_{i+1}$, then system idleness (3) *under p' is strictly less then that under p .*

Proof

Again, let $h_i = \frac{(i+1)\mu}{\lambda_H + \lambda_L p_i}$ and $l_{j,k} = \prod_{i=j}^{k} h_i$. Whenever the range is empty, we define the product to be 1 and the summation to be 0.

In the proof of Lemma 1, we showed that a decrease in h_i could be compensated by a simultaneous increase in h_{i+1} to maintain a constant $X = \sum_{s=0}^{m_I-1} l_{s,m_I-1}$.

Now consider such a change and its effect on each of the terms l_{s,m_I-1} within the summation. In particular, note

i) $\sum_{s=i+2}^{m_I-1} l_{s,m_I-1} = \sum_{s=i+2}^{m_I-1} \prod_{j=s}^{m_I-1} h_j$ does not change.

ii) From (i), we see that $\sum_{s=0}^{i+1} l_{s,m_I-1}$ remains constant, since $X - \sum_{s=i+2}^{m_I-1} l_{s,m_I-1}$ does not change.

iii) The single term, $l_{i+1,m_I-1} = \prod_{j=i+1}^{m_I-1} h_j$, increases.

iv) From (ii) and (iii) we see that

- l_{i+1,m_I-1} increases and $\sum_{s=0}^{i+1} l_{s,m_I-1}$ remains constant $\Longrightarrow \sum_{s=0}^{i} l_{s,m_I-1}$ decreases;
- $l_{0,m_I-1}, \ldots, l_{i,m_I-1}$ all move in the same direction (as $h_i h_{i+1}$) $\Longrightarrow l_{0,m_I-1}, \ldots, l_{i,m_I-1}$ all decrease.

Similarly, consider the change in the objective function. Recall that ξ_{m_I} remains constant. Using the definition of $l_{j,k}$ and grouping terms, as above, we see that (3) can be rewritten as

$$\xi_{m_I} \left[\sum_{s=0}^{i} (m_I - s)\, l_{s,m_I-1} \;+\; (m_I - (i+1))\, l_{i+1,m_I-1} \;+\; \sum_{s=i+2}^{m_I-1} (m_I - s)\, l_{s,m_I-1} \right].$$

Furthermore, noting that $m_I - s > m_I - (i+1)$ for $s < i+1$ we can rearrange terms to restate (3) as

$$\xi_{m_I} \left[\sum_{s=0}^{i} ((i+1) - s)\, l_{s,m_I-1} \;+\; \sum_{s=0}^{i+1} (m_I - (i+1))\, l_{s,m_I-1} \;+\; \sum_{s=i+2}^{m_I-1} (m_I - s)\, l_{s,m_I-1} \right].$$

Because the first term strictly decreases, and the second and third terms do not change, an appropriate, simultaneous increase in p_i and decrease in p_{i+1} strictly decreases system idleness. The same argument can be used to show that an opposite change strictly increases (3). □

Thus, by appropriately increasing the probability of putting a type-L call into service in state i and decreasing the probability of putting a type-L call into service in state $i+1$, the in-house pool can maintain the same type-H service level *and* strictly improve its type-L throughput.

Together, Lemmas 1 and 2 show that, whenever there exists a feasible p with $p_i < 1$ and $p_{i+1} > 0$, we can improve the NLP's objective function by simultaneously increasing p_i and decreasing p_{i+1} until one of them hits a boundary (either $p_i = 1$ or $p_{i+1} = 0$). In turn, this allows us to demonstrate that a randomized threshold reservation policy is optimal:

Proof of Theorem 1

Note that a stationary, type-H priority, type-H work-conserving policy is an (L, p_L) policy if and only if $p_i = 0 \implies p_j = 0$ for all $j > i$ and $p_i = 1 \implies p_j = 1$ for all $j < i$.

Then we consider each of two contradictory cases. First, suppose that there exists an optimal policy for which $p_i = 0$ and $p_j > 0$ for some $j > i$. Then there exists a k such that $p_k = 0$ and $p_{k+1} > 0$, and we can simultaneously maintain the feasibility of the service-level constraint (5) and decrease the objective function value (3) by increasing p_k and decreasing p_{k+1} as in Lemmas 1 and 2. Thus, the original policy could not have been optimal. Similarly, suppose there exists an optimal policy for which $p_i = 1$ and $p_j < 1$ for some $j < i$. Then there exists $p_{k-1} < 1$ and $p_k = 1$, and the argument used in the previous case leads to a contradiction here as well. □

Proof of Proposition 1

As in Lemma 2, define $X = \sum_{s=0}^{m_I-1} \prod_{i=s}^{m_I-1} \frac{(i+1)\mu}{\lambda_H+\lambda_L p_i}$. Then differentiation shows that $\frac{\partial X}{\partial p_i} < 0$ for all p_i, so $\xi_{m_I} = [X + 1/(1-\rho)]^{-1}$ is increasing in each p_i.

Now consider two sets of routing probabilities, $p, p' \in [0,1]^{m_I}$, with $p'_k > p_k$ and $p'_i = p_i$ for all $i \neq k$, and define $\Delta_i = \xi_i(p') - \xi_i(p)$. Then we have the following.

i) As demonstrated above, $\xi_{m_I}(p') > \xi_{m_I}(p)$, or $\Delta_{m_I} > 0$. The fact that $\xi_i = \xi_{m_I}\rho^{i-m_I}$ for all $i > m_I$ implies that $\xi_i(p') > \xi_i(p)$, or $\Delta_i > 0$ for all $i \geq m_I$.

ii) Together (i) and the fact that $\sum_{i=0}^{\infty} \xi(p') = \sum_{i=0}^{\infty} \xi(p) = 1$ imply that $\sum_{i=0}^{m_I-1} \xi(p') < \sum_{i=0}^{m_I-1} \xi(p)$, or $\sum_{i=0}^{m_I-1} \Delta_i < 0$.

iii) Together (i) and the fact that $\xi_i = \xi_{m_I} \prod_{j=i}^{m_I-1} \frac{(j+1)\mu}{\lambda_H+\lambda_L p_j}$ imply that, for all $i \in \{k+1, \ldots, m_I-1\}$, we have $\xi_i(p') > \xi_i(p)$, or $\Delta_i > 0$.

iv) Together (ii) and (iii) imply that $\sum_{i=0}^{k} \xi(p') < \sum_{i=0}^{k} \xi(p)$, or $\sum_{i=0}^{k} \Delta_i < 0$.

v) Together (iv) and the fact that $\xi_i = \xi_k \prod_{j=i}^{k} \frac{(j+1)\mu}{\lambda_H+\lambda_L p_j}$, or equivalently $\frac{\xi_i(p)}{\xi_k(p)} = \frac{\xi_i(p')}{\xi_k(p')}$, imply that, for all $i \leq p$, we have $\xi_i(p') < \xi_i(p)$, or $\Delta_i < 0$.

Finally, recall that the NLP's objective (3) is to minimize system idleness, $\sum_{i=0}^{m_I-1}(m_I - i)\xi_i$.

Then we have $\xi_i(p') = \xi_i(p) + \Delta_i$, and

$$\begin{aligned}\sum_{i=0}^{m_I-1}(m_I-i)\xi_i(p') &= \sum_{i=0}^{m_I-1}(m_I-i)\xi_i(p) + \sum_{i=0}^{k}(m_I-i)\Delta_i + \sum_{i=k+1}^{m_I-1}(m_I-i)\Delta_i \\ &\leq \sum_{i=0}^{m_I-1}(m_I-i)\xi_i(p) + \sum_{i=0}^{k}(m_I-(k+1))\Delta_i + \sum_{i=k+1}^{m_I-1}(m_I-(k+1))\Delta_i\end{aligned}$$

since $\Delta_i < 0$ for $i \leq k$ and $\Delta_i > 0$ for $i > k$. But

$$\sum_{i=0}^{k}(m_I-(k+1))\Delta_i + \sum_{i=k+1}^{m_I-1}(m_I-(k+1))\Delta_i = \sum_{i=0}^{m_I-1}(m_I-(k+1))\Delta_i < 0$$

since $\sum_{i=0}^{m_I-1}\Delta_i < 0$. Therefore, $\sum_{i=0}^{m_I-1}(m_I-i)\xi_i(p') < \sum_{i=0}^{m_I-1}(m_I-i)\xi_i(p)$, so the objective function (3) is decreasing in p_k. □

B.2 Efficiency of Type-H Priority Policies

In this section we prove Proposition 2, which states that the expected throughput of the best type-H priority policy is within $\frac{\lambda_L}{\lambda_L+\lambda_H+m_I\mu}$ of an upper bound on the globally optimal throughput for the pooled-overflow system.

Proof of Proposition 2

Part (i).

We use a sample-path argument to prove that the optimal throughput when $\lambda_L = \infty$ is an upper bound on the globally optimal throughput when $\lambda_L < \infty$. Let system 1 have "$\lambda_L = \infty$" – so that there is always a type-L call waiting to be served – and let system 2 have $\lambda_L < \infty$. Then a simple sample-path argument shows that any policy used in system 2 – including the optimal policy – can be matched exactly in system 1, so that the type-L throughput and type-H delay performance of the two systems are identical. Since this policy is feasible, though not necessarily optimal, for system 1, optimal type-L throughput in system 1 must be at least as large as that in system 2.

The sample-paths of the two systems are constructed to have the same type-H inter-arrival times and the same service times, once calls are put into service. The only difference between the two systems is that, rather than having explicit type-L inter-arrival times, system 1 always has a type-L call waiting to be served.

The sample-path argument, itself, is then trivial. At every instant that a call of either type arrives to system 2, there also exists a call of the same type in system 1 that is waiting to be served, and every call that is put into service in system 2 can be feasibly put into service in system 1. Thus system 1 can feasibly match the performance of system 2.

Part (ii)

Next, we use a coupling argument to show that, for a fixed $\lambda_L < \infty$, there exists a type-H priority

policy whose type-L throughput is at least as great as $\frac{\lambda_L}{\lambda_L+\lambda_H+m_I\mu}$ of the globally optimal throughput when $\lambda_L = \infty$.

Let system 1 have "$\lambda_L = \infty$" – so that there always exists a type-L call waiting to be served – and let system 2 have $\lambda_L < \infty$. We couple the two systems using a single set of i.i.d., exponentially distributed inter-event times of rate $\lambda_H + \lambda_L + m_I\mu$. Without loss of generality, we assume that the time scale is set so that $\lambda_H + \lambda_L + m_I\mu = 1$.

The event generator is driven off of system 1 and works as follows. Given inter-event times that are exponentially distributed with mean $1/(\lambda_H+\lambda_L+m_I\mu) = 1$, the conditional probability that the next event is a service completion by CSR i equals μ. If CSR i is busy, then the service is real, and if it is idle, then the service is a "dummy" completion. Similarly, the conditional probability the event is a type-H arrival equals λ_H, and the probability that it is a "dummy" type-L arrival equals λ_L. Note that, since system 1 always has a type-L call waiting to be served, the arrival of this additional call does not change the system state.

System 2 is coupled to system 1 in a straightforward fashion. Type-H arrivals are the same as in system 1. Type-L arrivals in this system are at the same time as in system 1; in this case they are real, however. Service completions occur at the same epochs as in system 1. Furthermore, the CSRs in the two systems are matched and labelled $i = 1, \ldots, m_I$.

Two related points concerning this coupling mechanism are important to observe. First, we need only consider policies which put calls into service at event epochs, and in this proof we only consider policies within this broad class. In addition, for system 1 we only consider policies which put type-L calls into service at event epochs that correspond to type-H arrivals or at service completions, since these are the event epochs associated with system 1. That is, in system 1 there always exists a type-L call waiting to be served, type-L "arrivals" are dummy, and there exist optimal policies which ignore these dummy type-L arrival epochs.

Then given an arbitrary call-routing policy (within the class described above) that is used in system 1, we construct a policy for system 2 as follows. Whenever system 1 puts a type-H job into service with CSR i, system 2 does so as well. Whenever system 1 puts a type-L job into service with CSR i, system 2 waits for the next event: if it is a type-L arrival, then system 2 puts the type-L call into service with CSR i as well; if it is a service completion by CSR i, then system 2 does nothing; in all the other cases, system 2 puts a "dummy" type-L job into service with CSR i so that, for accounting purposes, the occupancies of the two systems remains the same.

One last important element of the coupling mechanism is worth noting here. When system 2 puts a type-L call into service – either dummy or real – the processing time of the call equals the residual service time of the type-L call that had previously been put into service in system 1. Because type-L service times are exponentially distributed with rate μ, however, the conditional distribution of the residual service-time in system 1 is exponential with rate μ as well. Therefore, by assigning system 1's residual service times to type-L calls in system 2, we generate a sequence of service times for

type-L calls in system 2 that are exponentially distributed with rate μ as well.

Thus, the only times at which the evolutions of the two systems differ are those moments in which system 1 has put a type-L call into service and system 2 is still waiting for the next event to occur. At these times, the number of busy servers in system 2 is one less than in system 1; and after the next event, the two "accounting" occupancies are, again, the same. Most importantly, there are always at least as many idle servers in system 2 as in system 1.

We claim that system 2 feasibly handles a fraction $\frac{\lambda_L}{\lambda_H+\lambda_L+m_I\mu}$ of the type-L throughput of system 1. For feasibility, note that system 2 can and does take type-H calls at exactly the same epochs as system 1. Therefore the length of the type-H queue is always equal in the two systems, and system 2 is feasible whenever system 1 is. For throughput, note that every time system 1 takes a type-L call, there is a probability of $\frac{\lambda_L}{\lambda_H+\lambda_L+m_I\mu}$ that the next event will be a type-L arrival and that system 2 will take a real, rather than dummy, type-L job as well. So the type-L throughput of system 2 is $\frac{\lambda_L}{\lambda_H+\lambda_L+m_I\mu}$ times that of system 1.

We also note that if system 1 uses a type-H priority policy, then the policy induced in system 2 will be of type-H priority as well. Furthermore, from [21] we know that there exist type-H priority policies that are optimal in system 1. Thus, there exists a type-H priority policy in system 2 that achieves $\frac{\lambda_L}{\lambda_H+\lambda_L+m_I\mu}$ of the type-L throughput obtained using the optimal policy in system 1. □

Proposition 2 states that, for λ_L that is "large" with respect to λ_H, μ, and m_I, the performance of type-H priority policies should be excellent. A natural next question is "how large is 'large'?" The results of numerical tests, shown below, indicate that the numbers are quite reasonable.

We set the time scale so that average service times equal $\mu^{-1} = 3.33$ minutes, and we impose a service-level constraint that the ASA of type-H calls must be 0.5 minutes or less. We then systematically vary λ_H and λ_L. The three panels of Figure 9 display the results for three sizes of in-house pools: a small system, driven by $\lambda_H = 6$, or equivalently $R_H = \lambda_H/\mu = 20$; a medium system, with $\lambda_H = 30$ and $R_H = 100$; and a larger system, with $\lambda_H = 150$ and $R_H = 500$.

In each panel the horizontal axis shows the number of CSRs used in the pool, as it climbs above the minimum needed to meet the type-H service-level constraint. The vertical axis marks the load of type-L calls that is processed in house. (We plot load, in CSRs, rather than throughput rates, so that the scales of the two axes are comparable.) Each curve within a panel shows, for a given $R_L = \lambda_L/\mu$, the load of type-L calls processed in-house by the optimal type-H priority policy.

Figure 9 shows that, in fact, type-L throughput is nearly optimal for relatively large R_L. In each panel, the curve for $R_L = \infty$ represents an upper bound on optimal performance, and the curves for the finite R_L's systematically approach the upper bound. For small in-house systems, with $R_H = 20$, the performance when $R_L = 100$ is nearly optimal. This represents a rate of low-value calls that is five times the rate of the high-value calls. Given the "80–20" maxim, that 20% of the customers provide 80% of the value to a company, this appears to be a quite reasonable balance. Furthermore, as the

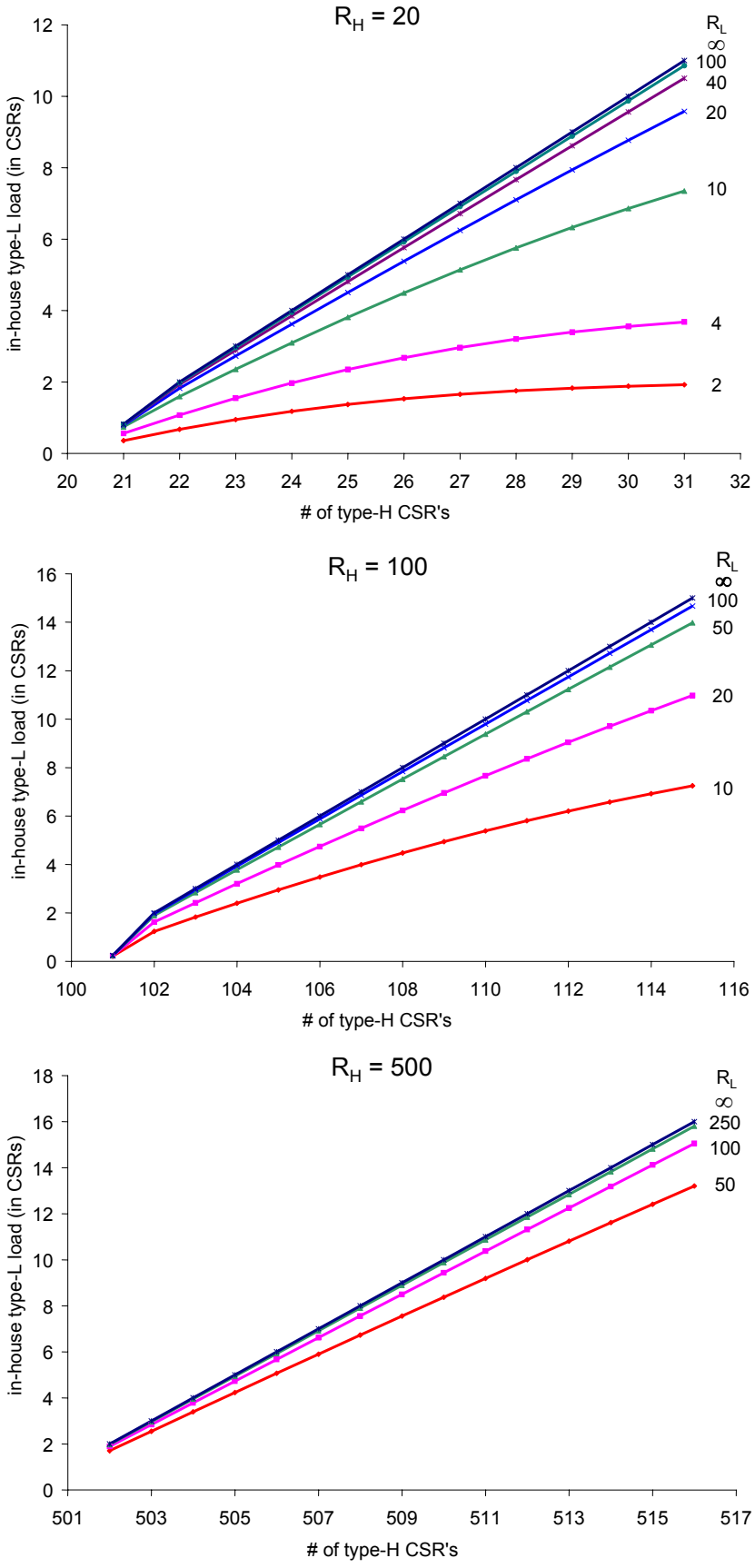

Figure 9: In-House Type-L Load for Type-H Priority Policies

scale of type-H traffic grows, the relative level of type-L traffic required to obtain nearly-optimal performance systematically declines. With $R_H = 500$, low-value calls need only have $R_L = 250$ – *half* the rate of high-value traffic – in order to provide excellent performance. Thus, when $\mu_H = \mu_L$, type-H priority policies should have excellent, if not globally optimal, performance for relatively large systems.

C Performance Bounds for the N-Network System in Section 5

In this appendix we prove Proposition 3's performance bounds for the N-network system.

Proof of Proposition 3

Part (i)

Step 1. Given an N-network with m_I in-house and m_O outsourcer CSRs and routing policy π, consider a pooled system with $m = m_I + m_O$ CSRs, all capable of handling both types of calls. Label the first m_I of the pool "in-house" and the last m_O of them "outsourcer" and use routing policy π. The performance of the two systems will be exactly the same.

Step 2. Consider the pooled system in which routing policy π is used. Let ASA_H^π and ASA_L^π be the ASA achieved by policy π in the N-network for type-H and type-L calls respectively. Then because $ASA_H^\pi \leq ASA^*$ and $ASA_L^\pi \leq ASA^*$,

$$ASA^\pi \;=\; \frac{\lambda_H}{\lambda_H+\lambda_L}\,ASA_H^\pi \;+\; \frac{\lambda_L}{\lambda_H+\lambda_L}\,ASA_L^\pi \;\leq\; ASA^*.$$

From Little's law, we know that there exists a long-run average number in queue, L^π, such that $L^\pi = (\lambda_H + \lambda_L)ASA^\pi$.

Step 3. We use a coupling argument to show we can construct a not necessarily work-conserving first-come-first-served service discipline, π', that maintains the same number-in-queue process as there is under π. More specifically, let $\{T_1, T_2, \ldots\}$ be an i.i.d. sequence of exponentially-distributed inter-arrival times of mean $(\lambda_H + \lambda_L)^{-1}$. Let $\{p_1, p_2, \ldots\}$ be an i.i.d. sequence of Bernoulli random variables that equal one – corresponding to a type-H, rather than type-L, arrival – with probability $\frac{\lambda_H}{\lambda_H+\lambda_L}$. Finally, let $\{S_1, S_2, \ldots\}$ be an i.i.d. sequence of exponentially-distributed service times of mean μ^{-1}. These service-time realizations are sampled in the order in which calls are put into service, no matter whether the call is of type-H or type-L.

Then every time π puts a call into service, π' does as well. Since the inter-arrival-time, call-type, and service-time samples are the same in the two systems, the number-in-system process is identical in the two systems. In turn, $L^\pi = L^{\pi'}$.

Step 4. Define FCFS to be a work-conserving first-come-first-served service discipline. Then a FCFS policy will have number-in-system process that is no more than that of π'. Therefore,

$L^{FCFS} \leq L^{\pi}$, and by Little's Law

$$ASA^{FCFS} \;=\; \frac{L^{FCFS}}{\lambda_H + \lambda_L} \;\leq\; \frac{L^{\pi}}{\lambda_H + \lambda_L} \;=\; ASA^{\pi} \;\leq\; ASA^{*}.$$

Finally, by PASTA, we know that expected delay in queue of every type-H and every type-L call is

$$ASA_H^{FCFS} \;=\; ASA_L^{FCFS} \;=\; ASA^{FCFS} \;\leq\; ASA^{*}.$$

Thus, if policy π is feasible for a system with m_I in house and m_O outsourcer CSRs, then a FCFS policy with $m = m_I + m_O$ pooled CSRs is feasible as well. This proves part (i).

Part (ii)

The result follows from the fact that the globally optimal type-L throughput when $\lambda_L = \infty$ is also an upper bound on the throughput achievable in the N-network scheme. The proof is analogous to that of Proposition 2. □

D Inter-Overflow Time CV in the Pooled-Overflow System

The analysis in this appendix characterizes the moments of the inter-overflow times of type-L calls from the client company to the outsourcer in the pooled-overflow scheme. This information is used to numerically evaluate the coefficient of variation (CV) of the inter-overflow time in the examples in Section 6.4.

For notational convenience, we drop the subscript, I, from the number of in-house CSRs, m_I. Thus, the total number of in-house CSRs is referred to as "m" below.

When the client company uses the (L, p_L) policy to for type-L call admission, the overflow of type-L calls to the outsourcer follows a Markov Modulated Poisson Process (MMPP): it is Poisson with rate λ_L when $s > L$, Poisson with rate $(1 - p_L)\lambda_L$ when $s = L$, and rate 0 when $s < L$. For such a policy, $p_0 = \cdots = p_{L-1} = 1$ and $p_{L+1} = \cdots = p_{m-1} = 0$.

We now characterize the inter-overflow time. Let $\tilde{T}$ denote the random inter-overflow time, and let $\tilde{T}_s$, $\forall\ s$, denote the random time to next overflow starting in after-action state s. Because the overall arrival process of type-L calls is Poisson, we can apply PASTA to obtain

$$\mathbf{P}\left\{\tilde{T} \leq t\right\} \;=\; \sum_s \left(\frac{\xi_s(1-p_s)}{\sum_s \xi_s(1-p_s)}\right) \mathbf{P}\left\{\tilde{T}_s \leq t\right\}, \tag{7}$$

where ξ_s can be computed from (2) with $p_0 = \cdots = p_{L-1} = 1$ and $p_{L+1} = \cdots = p_{m-1} = 0$.

Because $\mathbf{P}\left\{\tilde{T} \leq t\right\}$ is a convex combination of $\mathbf{P}\left\{\tilde{T}_s \leq t\right\}$ for all $s \geq L$, the k^{th} moment of $\tilde{T}$ will be the same convex combination of the k^{th} moment of $\tilde{T}_s$. Therefore, it suffices to find all the moments of $\tilde{T}_s$ for any $s \geq L$.

Let $\mu_s = \begin{cases} s\mu, & \forall s \le m \\ m\mu, & \forall s > m \end{cases}$ be the total service rate in state s and $\lambda_s = \lambda_H + \lambda_L$ be the total arrival rate in state s. Then $\omega_s = \lambda_s + \mu_s$ is the total rate at which next event takes place in state s. In particular, $\omega_m = \lambda_H + \lambda_L + m\mu$. Denoting by $\tilde{e}_\omega$ an exponential random variable with rate ω, we obtain the following relationship among all $\tilde{T}_s$s:

$$\tilde{T}_s \;=\; \tilde{e}_{\omega_s} + \begin{cases} \tilde{T}_{s-1} & \text{w.p.} \quad \mu_s/\omega_s \\ 0 & \text{w.p.} \quad (1-p_s)\lambda_L/\omega_s \\ \tilde{T}_{s+1} & \text{w.p.} \quad (\lambda_H + p_s\lambda_L)/\omega_s \end{cases} \qquad \forall\, s \ge 1. \tag{8}$$

Let $T_s(\theta)$ denote the Laplace transform of $\tilde{T}_s$. Then (8) implies

$$T_s(\theta) \;=\; \frac{\mu_s}{\omega_s+\theta} T_{s-1}(\theta) + \frac{(1-p_s)\lambda_L}{\omega_s+\theta} + \frac{\lambda_H + p_s\lambda_L}{\omega_s+\theta} T_{s+1}(\theta), \qquad \forall s \ge 1. \tag{9}$$

For all $s \ge m$, $p_s = 0$, $\mu_s = m\mu$, and $\omega_s = \omega_m$. Therefore when $s \ge m$, from (9) we obtain

$$T_{m+q}(\theta) = \frac{m\mu}{\omega_m+\theta} T_{m+q-1}(\theta) + \frac{\lambda_L}{\omega_m+\theta} + \frac{\lambda_H}{\omega_m+\theta} T_{m+q+1}(\theta), \qquad \forall q \ge 1, \tag{10}$$

where $q = s - m$ and q is the type-H queue length.

The solution to (10) has a geometric form, determined completely by the boundary point, $T_m(\theta)$:

Proposition 4
The solution to (10) *is:*

$$T_{m+q}(\theta) = T_m(\theta) z_1^q(\theta) + \frac{\lambda_L}{\lambda_L+\theta}\left[1 - z_1^q(\theta)\right], \tag{11}$$

where $z_1(\theta) = \frac{(\omega_m+\theta) - \sqrt{(\omega_m+\theta)^2 - 4m\mu\lambda_H}}{2\lambda_H}$.

Proof
The proof will follow these three steps:

1. Show that the solution to (10) is of the form of $T_{m+q}(\theta) = J_1(\theta) z_1(\theta)^q + J_2(\theta) z_2(\theta)^q + \frac{\lambda_L}{\lambda_L+\theta}$, for some easily calculated $J_1(\theta), J_2(\theta), z_1(\theta), z_2(\theta)$, where $0 \le z_1(\theta) < 1 < z_2(\theta)$.
2. Show that $\lim_{q\to\infty} T_{m+q}(\theta) = \frac{\lambda_L}{\lambda_L+\theta}$.
3. From the first two steps, we must have $J_2(\theta) = 0$ and $J_1(\theta) = T_m(\theta) - \frac{\lambda_L}{\lambda_L+\theta}$.

Step 1 To simplify notation, we will suppress the θ in all of the Laplace transforms, so that $T_{m+q}(\theta)$ will become T_{m+q}, and so on. We define $\gamma_1 = \frac{m\mu}{\omega_m+\theta}, \gamma_2 = \frac{\lambda_H}{\omega_m+\theta}, \gamma_3 = \frac{\lambda_L}{\omega_m+\theta}$. Then (10) becomes:

$$T_{m+q} = \gamma_1 T_{m+q-1} + \gamma_3 + \gamma_2 T_{m+q+1} \qquad \forall q \ge 1. \tag{12}$$

Multiplying both sides by z^{q+1} and summing from 1 to ∞, we obtain

$$\sum_{q=1}^{\infty} T_{m+q} z^{q+1} = \gamma_1 \sum_{q=1}^{\infty} T_{m+q-1} z^{q+1} + \gamma_3 \sum_{q=1}^{\infty} z^{q+1} + \gamma_2 \sum_{q=1}^{\infty} T_{m+q+1} z^{q+1}. \tag{13}$$

Denoting $f(z) = \sum_{q=0}^{\infty} T_{m+q} z^q$, we have

$$\begin{aligned} z[f(z) - T_m] &= \gamma_1 z^2 f(z) + \gamma_3 z^2/(1-z) + \gamma_2[f(z) - T_m - T_{m+1} z] && (14) \\ \Longrightarrow f(z) &= \frac{\gamma_2 T_{m+1} z - T_m z - \gamma_3 z^2/(1-z) + \gamma_2 T_m}{\gamma_1 z^2 - z + \gamma_2} && (15) \\ &= \frac{(1-z)(\gamma_2 T_{m+1} z - T_m z + \gamma_2 T_m) - \gamma_3 z^2}{\gamma_2 (1 - z_1 z)(1 - z_2 z)(1-z)}, && (16) \end{aligned}$$

where $z_1 \le z_2$ are the two roots of $\gamma_2 z^2 - z + \gamma_1 = 0$ (because $\gamma_2 \neq 0$):

$$z_1 = \frac{1 - \sqrt{1 - 4\gamma_1\gamma_2}}{2\gamma_2} \qquad \text{and} \qquad z_2 = \frac{1 + \sqrt{1 - 4\gamma_1\gamma_2}}{2\gamma_2}. \tag{17}$$

Since $z_1 + z_2 > 0$ and $z_1 z_2 = \gamma_1/\gamma_2 > 1$, we have $0 < z_1 < 1 < z_2$. Carrying out partial-fraction expansion, we obtain

$$f(z) = \left[\frac{J_1}{1 - z_1 z} + \frac{J_2}{1 - z_2 z} + \frac{J_3}{1 - z} \right], \tag{18}$$

where

$$\begin{aligned} J_1 &= [1 - z_1 z] f(z) \mid_{z = 1/z_1} = \frac{\gamma_2 T_{m+1} + (\gamma_2 z_1 - 1) T_m + \gamma_3/(1 - z_1)}{\gamma_2 (z_1 - z_2)}, \\ J_2 &= [1 - z_2 z] f(z) \mid_{z = 1/z_2} = \frac{\gamma_2 T_{m+1} + (\gamma_2 z_2 - 1) T_m + \gamma_3/(1 - z_2)}{\gamma_2 (z_2 - z_1)}, \\ \text{and } J_3 &= (1 - z) f(z) \mid_{z=1} = \frac{-\gamma_3}{\gamma_1 - 1 + \gamma_2} = \frac{\lambda_L}{\lambda_L + \theta}. \end{aligned} \tag{19}$$

Then from (18) we have

$$T_{m+q} = J_1 z_1^q + J_2 z_2^q + \frac{\lambda_L}{\lambda_L + \theta}. \tag{20}$$

Because $\gamma_2(z_2 - z_1) \neq 0$, J_1 and J_2 always exist. Thus (18) and (20) are always valid. And it is clear that the series T_{m+q}, as defined in (20), satisfies (12).

Step 2 When an overflow occurs in state s, the probability that the next type-L arrival will also be overflowed approaches 1 as $s \to \infty$. Therefore, the time until the next overflow approaches an exponential random variable with rate λ_L as $s \to \infty$. We now formalize this.

Suppose we start in state $m + q$. Denote by $\tilde{R}_q$ the random amount of time it takes to reach the first state, L, in which it is possible to accept a new type-L call. Also, we denote by $\tilde{E}$ the random amount of time it takes for the next type-L call to arrive. Clearly $\tilde{E}$ is exponentially distributed with rate λ_L.

It is easy to show (by induction, for example) that $\tilde{R}_q$ is stochastically larger than the sum of $q + 1$ $\exp(\omega_m)$ random variables, one $\exp(\omega_{m-1})$ random variable, one $\exp(\omega_{m-2})$ random variable,

..., and one $\exp(\omega_{L+1})$ random variable. This sum, in turn, is stochastically larger than the sum of $q+m-L$ $\exp(\omega_m)$ random variables.

So if we denote a $\Gamma(q+m-L,\omega_m)$ random variable by $\tilde{X}_q$ and its *p.d.f.* by $g_q(t) = \frac{\omega_m e^{-\omega_m t}(\omega_m t)^{q+m-L-1}}{(q+m-L-1)!}$, we have

$$\begin{aligned}\mathbf{P}\left\{\tilde{R}_q \le \tilde{E}\right\} &= \int_0^\infty \mathbf{P}\left\{\tilde{R}_q < t\right\} \lambda_L e^{-\lambda_L t} dt \le \int_0^\infty \mathbf{P}\left\{X < t\right\} \lambda_L e^{-\lambda_L t} dt \\ &= \mathbf{P}\left\{X \le \tilde{E}\right\} = \int_0^\infty \mathbf{P}\left\{\tilde{E} \ge t\right\} g_q(t) dt = \int_0^\infty e^{-\lambda_L t} g_q(t) dt.\end{aligned}$$

Now $\forall\ \epsilon > 0, \exists\ \bar{t}$, s.t. $e^{-\lambda_L \bar{t}} < \epsilon$. Moreover, $g_q(t) = \frac{\omega_m e^{-\omega_m t}(\omega_m t)^{q+m-L-1}}{(q+m-L-1)!} \to 0$ uniformly on $t \in [0,\bar{t}]$ as $q \to \infty$. Hence $\exists Q_\epsilon$ s.t. $g_q(t) \le \epsilon$ for all $q > Q_\epsilon$ and $t \in [0,\bar{t}]$, and

$$\begin{aligned}\int_0^\infty e^{-\lambda_L t} g_q(t) dt &= \int_{\bar{t}}^\infty e^{-\lambda_L t} g_q(t) dt + \int_0^{\bar{t}} e^{-\lambda_L t} g_q(t) dt \\ &\le \epsilon \int_{\bar{t}}^\infty g_q(t) dt + \epsilon \int_0^{\bar{t}} e^{-\lambda_L t} dt \le \epsilon \left(1 + \frac{1}{\lambda_L}\right), \qquad \forall\ q > Q_\epsilon.\end{aligned}$$

Therefore, $\lim_{q\to\infty} \mathbf{P}\left\{\tilde{R}_q \le \tilde{E}\right\} = 0$.

If we denote the *c.d.f.* of $\tilde{T}_{m+q}$ by $F_q(t)$, then

$$F_q(t) = \mathbf{P}\left\{\tilde{T}_{m+q} \le t\right\} = \mathbf{P}\left\{\tilde{R}_q + \tilde{T}_L \le t\right\} \mathbf{P}\left\{\tilde{R}_q \le \tilde{E}\right\} + (1 - e^{-\lambda_L t}) \mathbf{P}\left\{\tilde{R}_q > \tilde{E}\right\}.$$

Therefore,

$$\begin{aligned}\mid F_q(t) - (1 - e^{-\lambda_L t}) \mid &\le \mathbf{P}\left\{\tilde{R}_q \le \tilde{E}\right\} \left[(1 - e^{-\lambda_L t}) + \left|\mathbf{P}\left\{\tilde{R}_q + \tilde{T}_{\bar{x}_L} \le t\right\}\right|\right] \\ &\le 2\mathbf{P}\left\{\tilde{R}_q \le \tilde{E}\right\}. \qquad (21)\end{aligned}$$

Because (21) holds for all t and $\lim_{q\to\infty} \mathbf{P}\left\{\tilde{R}_q \le \tilde{E}\right\} = 0$, we have $\lim_{q\to\infty} F_q(t) = 1 - e^{-\lambda_L t}$ uniformly. Moreover,

$$\begin{aligned}\lim_{q\to\infty} T_{m+q}(\theta) = \lim_{q\to\infty} \int_0^\infty e^{-st} dF_q(t) &\overset{*}{=} \lim_{q\to\infty} s \int_0^\infty e^{-st} F_q(t) dt \\ &\overset{\triangle}{=} s \int_0^\infty \lim_{q\to\infty} e^{-st} F_q(t) dt = s \int_0^\infty e^{-st}(1 - e^{-\lambda_L t}) dt = \frac{\lambda_L}{\lambda_L + \theta}.\end{aligned}$$

Here, (*) is in Wolff [52, p. 533, eq. (21)]; and ($\triangle$) follows from the uniform convergence of $F_q(t)$.

Step 3 Because $z_2(\theta) > 1$, the convergence of $T_{m+q}(\theta)$ implies that $J_2(\theta) = 0$. Consequently, $T_{m+q}(\theta) = J_1(\theta) z_1^q(\theta) + \frac{\lambda_L}{\lambda_L+\theta}$. Letting $q = 0$, we obtain $J_1(\theta) = T_m(\theta) - \frac{\lambda_L}{\lambda_L+\theta}$. Plugging this and $J_2(\theta) = 0$ into (20), we obtain

$$T_{m+q}(\theta) = \left(T_m(\theta) - \frac{\lambda_L}{\lambda_L + \theta}\right) z_1^q(\theta) + \frac{\lambda_L}{\lambda_L + \theta} = T_m(\theta) z_1^q(\theta) + \frac{\lambda_L}{\lambda_L + \theta}\left[1 - z_1^q(\theta)\right]. \qquad (22)$$

□

Proposition 4 solves the (infinitely many) tail equations (9) for $s > m$. To completely determine the system, we need (9) for $s \leq m$, (11), and one other equation. The following corollary provides this equation:

Corollary 1

$$\lambda_H T_{m+1}(\theta) + [\lambda_H z_2(\theta) - (\omega_m + \theta)]\, T_m(\theta) + \frac{\lambda_L}{1 - z_2(\theta)} = 0, \tag{23}$$

where $z_2(\theta) = \frac{(\omega_m+\theta)+\sqrt{(\omega_m+\theta)^2-4m\mu\lambda_H}}{2\lambda_H}$.

Proof

From the proof of Proposition 4 we know that $J_2(\theta) = 0$ for all θ. Therefore, from (19), we get

$$\frac{\gamma_2(\theta)T_{m+1}(\theta) + (\gamma_2(\theta)z_2(\theta) - 1)T_m(\theta) + \gamma_3(\theta)/(1 - z_2(\theta))}{\gamma_2(\theta)(z_2(\theta) - z_1(\theta))} = 0,$$

which simplifies to

$$\lambda_H T_{m+1}(\theta) + [\lambda_H z_2(\theta) - (\omega_m + \theta)]\, T_m(\theta) + \frac{\lambda_L}{1 - z_2(\theta)} = 0.$$

□

In what follows, we use the fact that the kth derivative of $-T_s(\theta)$ evaluated at 0 gives the kth moment of $\tilde{T}_s$ to derive expressions for the first two moments of the inter-overflow time, which is then used to calculate the CV.

We first derive a complete solution of (9). Equation (11) solves (9) for $s > m$. The remaining equations in (9), together with (23), are linear, and they can be easily and quickly solved. In matrix-vector notation, (9) for states $s \leq m$, together with (23), can be expressed as:

$$Q(\theta)X(\theta) = y(\theta), \tag{24}$$

where $X(\theta) = \Big(T_0(\theta), \ldots, T_m(\theta), T_{m+1}(\theta)\Big)^{\mathsf{T}}$, $y(\theta) = \Big(0, \ldots, \frac{(1-p_L)\lambda_L}{\omega_L+\theta}, \frac{\lambda_L}{\omega_{L+1}+\theta}, \ldots, \frac{\lambda_L}{\omega_m+\theta}, -\frac{\lambda_L}{\lambda_H(1-z_2(\theta))}\Big)^{\mathsf{T}}$ and

$$Q(\theta) = \begin{pmatrix}
1, & -\frac{\omega_0}{\omega_0+\theta}, & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & -\frac{i\mu}{\omega_i+\theta}, & 1, & -\frac{\lambda_H+\lambda_L}{\omega_i+\theta}, & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & -\frac{L\mu}{\omega_i+\theta}, & 1, & -\frac{\lambda_H+p_L\lambda_L}{\omega_L+\theta}, & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & -\frac{i\mu}{\omega_i+\theta}, & 1, & -\frac{\lambda_H}{\omega_i+\theta}, & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & -\frac{m\mu}{\omega_m+\theta}, & 1, & -\frac{\lambda_H}{\omega_m+\theta} & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & z_2(\theta) - \frac{\omega_m+\theta}{\lambda_H}, & 1 & . & . & . & . & . & . & . & .
\end{pmatrix}.$$

Here the superscript T indicates transposition.

To derive the moments of the inter-overflow time, we repeatedly differentiate (24) and set $\theta = 0$:

Theorem 2

Let $\mathbf{1} \in R^{m+2}$ *be the vector whose components are all 1. Then the first two moments of the inter-overflow time are*

$$E(\tilde{T}) = \frac{\sum_{s=L}^{m-1} \xi_s(1-p_s)E(\tilde{T}_s) + \xi_m \left[\frac{E(\tilde{T}_m)-\frac{1}{\lambda_L}}{1-z_1(0)\frac{\lambda_H}{m\mu}} + \frac{1}{\lambda_L\left(1-\frac{\lambda_H}{m\mu}\right)}\right]}{\sum_{s\geq L}\xi_s(1-p_s)}, \tag{25}$$

where

$$\left(E(\tilde{T}_0),\ldots,E(\tilde{T}_{m+1})\right)^{\mathsf{T}} = Q^{-1}(0)\left[Q'(0)\mathbf{1} - y'(0)\right], \tag{26}$$

and

$$E(\tilde{T}^2) = \frac{1}{\sum_{s\geq L}\xi_s(1-p_s)}\left\{\sum_{s=L}^{m-1}\xi_s(1-p_s)E(\tilde{T}_s^2) + \xi_m\left[\frac{E(\tilde{T}_m^2)-\frac{2}{\lambda_L^2}}{1-z_1(0)\left(\frac{\lambda_H}{m\mu}\right)} + \frac{2}{\lambda_L^2\left(1-\frac{\lambda_H}{m\mu}\right)} + \frac{2\left(\frac{1}{\lambda_L}-E(\tilde{T}_m)\right)z_1'(0)\left(\frac{\lambda_H}{m\mu}\right)}{\left(1-z_1(0)\frac{\lambda_H}{m\mu}\right)^2}\right]\right\}, \tag{27}$$

where

$$\left(E(\tilde{T}_0^2),\ldots,E(\tilde{T}_{m+1}^2)\right)^{\mathsf{T}} = Q^{-1}(0)[2Q'(0)Q(0)^{-1}Q'(0)\mathbf{1} - 2Q'(0)Q(0)^{-1}y'(0) - Q''(0)\mathbf{1} + y''(0)]. \tag{28}$$

Proof

Differentiating (24) once, we obtain $Q'(\theta)X(\theta)|_{\theta=0} + Q(\theta)X'(\theta)|_{\theta=0} = y'(\theta)|_{\theta=0}$. Since $X(0) = \mathbf{1}$, yields (26). This gives us $E(\tilde{T}_s)$ for states $s \leq m+1$. For all the other states, we use (11):

$$\begin{aligned} E(\tilde{T}_{m+q}) &= -T'_{m+q}(0) \\ &= -\left[T'_m(0)z_1^q(0) + T_m(0)qz_1^{q-1}(0)z_1'(0) - \frac{1}{\lambda_L}\left(1-z_1^q(0)\right) - qz_1^{q-1}(0)z_1'(0)\right] \\ &= \left(E(\tilde{T}_m) - \frac{1}{\lambda_L}\right)z_1^q(0) + \frac{1}{\lambda_L}, \end{aligned} \tag{29}$$

where $E(\tilde{T}_m)$ is already determined in (26). Note that (29) is derived for $q \geq 1$, but clearly the result applies to $q = 0$ as well.

So the overall first moment of overflow is:

$$\begin{aligned} E(\tilde{T}) &= \frac{\sum_{i=L}^{m-1}\xi_i(1-p_i)E(\tilde{T}_i) + \sum_{q=0}^{\infty}\xi_{m+q}E(\tilde{T}_{m+q})}{\sum_{s\geq L}\xi_s(1-p_s)} \\ &= \frac{\sum_{i=L}^{m-1}\xi_i(1-p_i)E(\tilde{T}_i) + \xi_m\sum_{q=0}^{\infty}\left(\frac{\lambda_H}{m\mu}\right)^q\left[\left(E(\tilde{T}_m)-\frac{1}{\lambda_L}\right)z_1^q(0)+\frac{1}{\lambda_L}\right]}{\sum_{s\geq L}\xi_s(1-p_s)} \\ &= \frac{\sum_{i=L}^{m-1}\xi_i(1-p_i)E(\tilde{T}_i) + \xi_m\left[\frac{E(\tilde{T}_m)-\frac{1}{\lambda_L}}{1-z_1(0)\frac{\lambda_H}{m\mu}} + \frac{1}{\lambda_L\left(1-\frac{\lambda_H}{m\mu}\right)}\right]}{\sum_{s\geq L}\xi_s(1-p_s)}. \end{aligned} \tag{30}$$

Thus we have shown (25).

Now for the second moments. For states $s \leq m+1$ we will differentiate (24) twice and set $\theta = 0$: $y''(0) = Q''(0)\mathbf{1} + 2Q'(0)X'(0) + Q(0)X''(0)$ and

$$\begin{aligned}
&\left(E(\tilde{T}_0^2), \ldots, E(\tilde{T}_{m+1}^2)\right)^{\mathsf{T}} \\
= \;& X''(0) = Q^{-1}(0)[-2Q'(0)X'(0) - Q''(0)\mathbf{1} + y''(0)] \\
= \;& Q^{-1}(0)[2Q'(0)Q(0)^{-1}\left(Q'(0)\mathbf{1} - y'(0)\right) - Q''(0)\mathbf{1} + y''(0)] \\
= \;& Q^{-1}(0)[2Q'(0)Q(0)^{-1}Q'(0)\mathbf{1} - 2Q'(0)Q(0)^{-1}y'(0) - Q''(0)\mathbf{1} + y''(0)].
\end{aligned}$$

This proves (28). The second moments of $\tilde{T}$ corresponding to other states can be calculated similarly to (29). For $q \geq 2$:

$$\begin{aligned}
E(\tilde{T}_{m+q}^2) &= T''_{m+q}(0) \\
&= T''_m(0)z_1^q(0) + 2T'_m(0)qz_1^{q-1}(0)z'_1(0) + T_m(0)q(q-1)z_1^{q-2}(0)[z'_1(0)]^2 + T_m(0)qz_1^{q-1}(0)z''_1(0) \\
&\quad + \frac{2}{\lambda_L^2}\left(1 - z_1^q(0)\right) - \frac{2}{\lambda_L}\left[-qz_1^{q-1}(0)z'_1(0)\right] - [q(q-1)z_1^{q-2}(0)[z'_1(0)]^2 + qz_1^{q-1}(0)z''_1(0)] \\
&= T''_m(0)z_1^q(0) + 2T'_m(0)qz_1^{q-1}(0)z'_1(0) + \frac{2}{\lambda_L^2}\left(1 - z_1^q(0)\right) - \frac{2}{\lambda_L}\left[-qz_1^{q-1}(0)z'_1(0)\right] \\
&= E(\tilde{T}_m^2)z_1^q(0) - 2qE(\tilde{T}_m)z_1^{q-1}(0)z'_1(0) + \frac{2}{\lambda_L^2}\left(1 - z_1^q(0)\right) + \frac{2}{\lambda_L}qz_1^{q-1}(0)z'_1(0), \qquad (31)
\end{aligned}$$

where $E(\tilde{T}_m)$ and $E(\tilde{T}_m^2)$ are determined in (26) and (28) respectively.

Note that even though (31) is derived for $q \geq 2$, it applies to $q = 0, 1$ as well: for $q = 0$, it is trivial; for $q = 1$ we have

$$\begin{aligned}
E(\tilde{T}_{m+1}^2) &= T''_{m+1}(0) \\
&= T''_m(0)z_1(0) + 2T'_m(0)z'_1(0) + \frac{2}{\lambda_L^2}\left(1 - z_1(0)\right) + \frac{2}{\lambda_L}z'_1(0) \\
&= E(\tilde{T}_m^2)z_1(0) - 2E(\tilde{T}_m)z'_1(0) + \frac{2}{\lambda_L^2}\left(1 - z_1(0)\right) + \frac{2}{\lambda_L}z'_1(0). \qquad (32)
\end{aligned}$$

Thus, the overall second moment of overflow is:

$$\begin{aligned}
E(\tilde{T}^2) &= \frac{1}{\sum_{s\geq L}\xi_s(1-p_s)}\left\{\sum_{s=L}^{m-1}\xi_s(1-p_s)E(\tilde{T}_s^2)\right. \\
&\quad \left. + \xi_m\sum_{q=0}^{\infty}\left(\frac{\lambda_H}{m\mu}\right)^q\left[E(\tilde{T}_m^2)z_1^q(0) - 2qE(\tilde{T}_m)z_1^{q-1}(0)z'_1(0) + \frac{2}{\lambda_L^2}(1 - z_1^q(0)) + \frac{2}{\lambda_L}qz_1^{q-1}(0)z'_1(0)\right]\right\} \\
&= \frac{1}{\sum_{s\geq L}\xi_s(1-p_s)}\left\{\sum_{s=L}^{m-1}\xi_s(1-p_s)E(\tilde{T}_s^2)\right. \\
&\quad \left. + \xi_m\left[\frac{E(\tilde{T}_m^2) - \frac{2}{\lambda_L^2}}{1 - z_1(0)\left(\frac{\lambda_H}{m\mu}\right)} + \frac{2}{\lambda_L^2\left(1 - \frac{\lambda_H}{m\mu}\right)} + \frac{2\left(\frac{1}{\lambda_L} - E(\tilde{T}_m)\right)z'_1(0)\left(\frac{\lambda_H}{m\mu}\right)}{\left(1 - z_1(0)\frac{\lambda_H}{m\mu}\right)^2}\right]\right\}.
\end{aligned}$$

Thus we have shown (27). □

The first two moments of the inter-overflow time, (25) and (27), allow us to easily calculate the CV of the inter-overflow time, $CV = \sqrt{E(\tilde{T}^2) - E^2(\tilde{T})} \,/\, E(\tilde{T})$. We use this approach for the numerical analysis in Section 6.4.

E Inter-Overflow-Time CC in the Pooled-Overflow System

The analysis in this appendix characterizes the serial correlation between successive inter-overflow times of type-L calls from the client company to the outsourcer in the pooled-overflow scheme. This information is used to numerically evaluate the correlation coefficient (CC) of the inter-overflow time in the examples in Section 6.4

For notational convenience, we drop the subscript, I, from the number of in-house CSRs, m_I. Thus, the total number of in-house CSRs is referred to as "m" below.

E.1 1-step correlation

The 1-step serial correlation of the inter-overflow time is defined as

$$CC = \frac{E(\tilde{T}_i\tilde{T}_{i+1}) - E(\tilde{T}_i)E(\tilde{T}_{i+1})}{\sigma_{\tilde{T}_i}\sigma_{\tilde{T}_{i+1}}}, \tag{33}$$

where $\tilde{T}_i$ and $\tilde{T}_{i+1}$ are two consecutive inter-overflow times.

Given a stationary system, $E(\tilde{T}_i) = E(\tilde{T}_{i+1}) = E(\tilde{T})$ and $\sigma_{\tilde{T}_i} = \sigma_{\tilde{T}_{i+1}} = \sqrt{E(\tilde{T}^2) - E^2(\tilde{T})}$, where $E(\tilde{T})$ and $E(\tilde{T}^2)$ are given in (25) and (27). So it remains to derive $E(\tilde{T}_i\tilde{T}_{i+1})$.

Some additional notation is introduced in Table 2.

$F_{ij}(t)$:	the probability that starting in state i the next overflow will occur in state j and it will take no more than t for that to occur
$f_{ij}(t)$:	$= F'_{ij}(t)$
$f_i(t)$:	the PDF of the time to next overflow if the current state is i
$F^k_{ij}(t)$:	the probability that starting in state i the subsequent k-th overflow will occur in state j and it will take no more than t for that to occur
$f^k_{ij}(t)$:	the corresponding PDF
E_i:	the expected time to next overflow if the current state is i, shorthand for $E(\tilde{T}_i)$

Table 2: Notation

When an (L, p_L) policy is used, an overflow can only occur in states L and above. Let π be the steady-state distribution of the beginning state of an overflow. Recall that $p_0 = \cdots = p_{L-1} = 1$ and

$p_{L+1} = \cdots = p_{m-1} = 0$. Thus, $\pi_i = \frac{(1-p_i)\xi_i}{\sum_j (1-p_j)\xi_j}, \forall i$, and

$$\begin{aligned} E(\tilde{T}_i \tilde{T}_{i+1}) &= \int_0^\infty \int_0^\infty xy \left(\sum_{i,j} \pi_i f_{ij}(x) f_j(y) \right) dxdy = \sum_{i,j} \pi_i E_j \int_0^\infty x f_{ij}(x) dx \\ &= -\sum_{i,j} \pi_i E_j \hat{f}'_{ij}(s)|_{s=0} = -(\pi_0, \pi_1, \dots) \cdot \hat{f}'(0) \cdot \begin{pmatrix} E_0 \\ E_1 \\ \dots \end{pmatrix}, \end{aligned} \tag{34}$$

where E_j is the shorthand for $E(\tilde{T}_j)$, and $\hat{f}(s)$ is the matrix containing $\hat{f}_{ij}(s)$, the Laplace transform of $f_{ij}(x)$ for all i, j. So it remains for us to solve for the matrix $\hat{f}'(0) = \{\hat{f}'_{ij}(s)|_{s=0}\}_{i,j}$.

1) For $j \neq i$,

$$\begin{aligned} F_{ij}(t) &= \Pr(\text{starting in } i, \text{ it takes no more than } t \text{ for the next overflow to occur in } j) \\ &= \lim_{\delta t \to 0} \Big\{ \Pr(0 \text{ event in } \delta t) F_{ij}(t-\delta t) + \Pr(\text{one event}) \Big[\frac{\mu_i}{\omega_i} F_{i-1,j}(t-\delta t) + \frac{(1-p_i)\lambda_L}{\omega_i} 0 \\ &\qquad + \frac{\lambda_H + p_i \lambda_L}{\omega_i} F_{i+1,j}(t-\delta t) \Big] + o(\delta t) \Big\} \\ &= \lim_{\delta t \to 0} \Big\{ e^{-\omega_i \delta t} F_{ij}(t-\delta t) + e^{-\omega_i \delta t} \delta t \left[\mu_i F_{i-1,j}(t-\delta t) + (\lambda_H + p_i \lambda_L) F_{i+1,j}(t-\delta t) \right] + o(\delta t) \Big\}. \end{aligned}$$

In turn,

$$\begin{aligned} \lim_{\delta t \to 0} \frac{F_{ij}(t) - F_{ij}(t-\delta t)}{\delta t} &= \lim_{\delta t \to 0} \left\{ \frac{e^{-\omega_i \delta t} - 1}{\delta t} F_{ij}(t-\delta t) + e^{-\omega_i \delta t} \left[\mu_i F_{i-1,j}(t-\delta t) + (\lambda_H + p_i \lambda_L) F_{i+1,j}(t-\delta t) \right] \right\} \\ f_{ij}(t) &= -\omega_i F_{ij}(t) + \mu_i F_{i-1,j}(t) + (\lambda_H + p_i \lambda_L) F_{i+1,j}(t). \end{aligned} \tag{35}$$

2) For $j = i$,

$$\begin{aligned} F_{ii}(t) &= \Pr(\text{starting in } i, \text{ it takes no more than } t \text{ for the next overflow to occur in } i) \\ &= \lim_{\delta t \to 0} \Big\{ \Pr(0 \text{ event}) F_{ii}(t-\delta t) + \Pr(\text{one event}) \Big[\frac{(1-p_i)\lambda_L}{\omega_i} 1 + \frac{\mu_i}{\omega_i} F_{i-1,i}(t-\delta t) \\ &\qquad + \frac{\lambda_H + p_i \lambda_L}{\omega_i} F_{i+1,i}(t-\delta t) \Big] + o(\delta t) \Big\} \\ &= \lim_{\delta t \to 0} \Big\{ e^{-\omega_i \delta t} F_{ii}(t-\delta t) + e^{-\omega_i \delta t} \delta t \left[(1-p_i)\lambda_L + \mu_i F_{i-1,i}(t-\delta t) + (\lambda_H + p_i \lambda_L) F_{i+1,i}(t-\delta t) \right] + o(\delta t) \Big\}. \end{aligned}$$

In turn,

$$\begin{aligned} \lim_{\delta t \to 0} \frac{F_{ii}(t) - F_{ii}(t-\delta t)}{\delta t} &= \lim_{\delta t \to 0} \Big\{ \frac{e^{-\omega_i \delta t} - 1}{\delta t} F_{ii}(t-\delta t) + e^{-\gamma_i \delta t} \left[(1-p_i)\lambda_L + \mu_i F_{i-1,i}(t-\delta t) \right. \\ &\qquad \left. + (\lambda_H + p_i \lambda_L) F_{i+1,i}(t-\delta t) \right] \Big\} \\ f_{ii}(t) &= -\omega_i F_{ii}(t) + (1-p_i)\lambda_L + \mu_i F_{i-1,i}(t) + (\lambda_H + p_i \lambda_L) F_{i+1,i}(t). \end{aligned}$$

So, if we let

$$\Gamma = \begin{pmatrix} 0 & 0 & 0 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & (1-p_L)\lambda_L & 0 & \dots \\ \dots & \dots & 0 & \lambda_L & 0 \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix} \leftarrow \text{row corresponding to state } L,$$

$$A = \begin{pmatrix} -\omega_0 & \lambda_H+\lambda_L & & & & & & & \\ & \cdots & & & & & & & \\ & \mu_i & -\omega_i & \lambda_H+\lambda_L & & & & & \\ & & & \cdots & & & & & \\ & & & \mu_L & -\omega_L & \lambda_H+p_L\mu_L & & & \\ & & & & & \cdots & \lambda_H & & \\ & & & & & \mu_m & -\omega_m & \lambda_H & \\ & & & & & & \mu_{m+1} & -\omega_{m+1} & \lambda_H \\ & & & & & & & \mu_{m+2} & \cdots \end{pmatrix}, \text{ and } F = \begin{pmatrix} F_{00} & F_{01} & \cdots \\ F_{10} & F_{11} & \cdots \\ \cdots & \cdots & \cdots \end{pmatrix},$$

then

$$F' = \Gamma + AF.^1 \tag{36}$$

Since $\hat{f}(s) = s\hat{F}(s)$, we have:

$$\hat{f}(s) = \frac{\Gamma}{s} + \frac{A}{s}\hat{f}(s) \quad \Rightarrow \quad (sI - A)\hat{f}(s) = \Gamma. \tag{37}$$

Differentiating this we obtain $\hat{f}(s) + (sI - A)\hat{f}'(s) = 0$, and $\hat{f}(s)|_{s=0} = A\hat{f}'(s)|_{s=0}$. From (37) we know that $\hat{f}(s)|_{s=0} = -A^{-1}\Gamma$. Thus $\hat{f}'(s)|_{s=0} = -A^{-2}\Gamma$.

So (34) becomes:

$$\begin{aligned} E(\tilde{T}_i\tilde{T}_{i+1}) &= (\pi_0, \pi_1, \dots, \pi_i, \dots) \cdot A^{-2} \cdot \Gamma \cdot \begin{pmatrix} E_0 \\ E_1 \\ \cdots \end{pmatrix}. \\ &= (\pi_0, \pi_1, \dots, \pi_i, \dots) \cdot A^{-2} \cdot \begin{pmatrix} 0 \\ \cdots \\ (1-p_L)\lambda_L E_L \\ \lambda_L E_{L+1} \\ \cdots \\ \lambda_L E_j \\ \cdots \end{pmatrix}. \end{aligned}$$

Clearly it would be hard to invert the infinite matrix A. So we will let U and V be such that

$$U \cdot A = (\pi_0, \pi_1, \dots, \pi_i, \dots), \tag{38}$$

$$A \cdot V = \begin{pmatrix} 0 \\ \cdots \\ (1-p_L)\lambda_L E_L \\ \lambda_L E_{L+1} \\ \cdots \\ \lambda_L E_j \\ \cdots \end{pmatrix}, \tag{39}$$

and derive U and V directly. $E(\tilde{T}_i\tilde{T}_{i+1}) = U \cdot V$ follows.

[1] The solution of (36) is straightforward: $F(t) = \left(e^{At} - I\right) A^{-1}\Gamma$ and $f(t) = e^{At}\Gamma$. This is consistent with equation (5) in Fischer and Meier-Hellstern [19]. The matrices are related as follows: $\Gamma = \Lambda$, $A = Q - \Lambda$. However, the Laplace transform approach we adopt here can be extended to calculate any k-lag correlation.

Our approach is to solve the tails of U and V. The rest of the probabilities can be obtained by inverting a finite portion of A.

Vector V: We know from (39) that, for states $i \geq m$,

$$m\mu v_{m+q} - (m\mu + \lambda_H + \lambda_L)v_{m+q+1} + \lambda_H v_{m+q+2} = \lambda_L E_{m+q+1}.$$

Let $f_v(s) = \sum_{q=0}^{\infty} v_{m+q}s^q$, then

$$f_v(s) = \frac{g_v(s)s^2 - (m\mu + \lambda_H + \lambda_L)v_m s + \lambda_H \left[v_m + v_{m+1}s\right]}{m\mu s^2 - (m\mu + \lambda_H + \lambda_L)s + \lambda_H}, \tag{40}$$

where due to (29),

$$g_v(s) = \sum_{q=0}^{\infty} \lambda_L E_{m+q+1}s^q = \frac{\lambda_L E_m - 1}{1/z_1(0) - s} + \frac{1}{1-s}. \tag{41}$$

Let s_1 and s_2 be the two roots of the denominator of (40):

$$s_1 = \frac{(m\mu + \lambda_H + \lambda_L) + \sqrt{(m\mu + \lambda_H + \lambda_L)^2 - 4m\mu\lambda_H}}{2m\mu} > 1 \tag{42}$$

$$s_2 = \frac{(m\mu + \lambda_H + \lambda_L) - \sqrt{(m\mu + \lambda_H + \lambda_L)^2 - 4m\mu\lambda_H}}{2m\mu} < 1. \tag{43}$$

It's important to note that $s_1 = 1/z_1(0)$, where $z_1(0)$ is previously defined in Proposition 4 and equation (17). Then (40) and (41) combine to yield:

$$f_v(s) = \frac{a_v}{s - s_1} + \frac{b_v}{s - s_2} + \frac{c_v}{s - 1} + \frac{d_v}{(s - s_1)^2}, \tag{44}$$

where a_v, b_v, c_v, and d_v are constants that can be determined as

$$a_v = \frac{df_v(s)\,(s - s_1)^2}{ds}\,|_{s=s_1}, b_v = f_v(s)\,(s - s_2)\,|_{s=s_2}, c_v = f_v(s)\,(s - 1)\,|_{s=1}, d_v = f_v(s)\,(s - s_1)^2\,|_{s=s_1}\,. \tag{45}$$

Expanding (44), we obtain the tail of v:

$$v_{m+q} = -a_v s_1^{-1-q} - b_v s_2^{-1-q} - c_v + d_v(q+1)s_1^{-2-q}. \tag{46}$$

We note that because $0 < s_2 < 1$, we must have $b_v = 0$, which yields

$$\left[(m\mu + \lambda_H + \lambda_L)s_2 - \lambda_H\right] v_m - \lambda_H s_2\; v_{m+1} = \left(\frac{\lambda_L E_m - 1}{s_1 - s_2} + \frac{1}{1 - s_2}\right) s_2^2. \tag{47}$$

(46) solves the tail for V using the equations $i > m$ in (39). The rest of the equations in (39), along with (47), give us the rest of the vector V:

$$\begin{pmatrix} v_0 \\ \cdots \\ v_i \\ \cdots \\ v_m \\ v_{m+1} \end{pmatrix} = \begin{pmatrix} -\omega_0 & \bar{\lambda} & & & & \\ & \cdots & & & & \\ & \mu_i & -\omega_i & \lambda_H + p_i\lambda_L & & \\ & & & \cdots & & \\ & & & \mu_m & -\omega_m & \lambda_H \\ & & & & \omega_m s_2 - \lambda_H & -\lambda_H s_2 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ \cdots \\ (1 - p_i)\lambda_L E_i \\ \cdots \\ \lambda_m E_m \\ \left(\frac{\lambda_L E_m - 1}{s_1 - s_2} + \frac{1}{1 - s_2}\right) s_2^2 \end{pmatrix}. \tag{48}$$

Vector U: Similarly we can find the tail of U. We know from (38) that, for states $i \geq m$,

$$\lambda_H u_{m+q} - (m\mu + \lambda_H + \lambda_L)u_{m+q+1} + m\mu u_{m+q+2} = \pi_{m+q+1}.$$

Define $f_u(s) = \sum_{q=0}^{\infty} u_{m+q}s^q$, then

$$f_u(s) = \frac{\frac{\pi_m}{m\mu/\lambda_H - s}s^2 - (m\mu + \lambda_H + \lambda_L)u_m s + m\mu\left[u_m + u_{m+1}s\right]}{\lambda_H s^2 - (m\mu + \lambda_H + \lambda_L)s + m\mu} \tag{49}$$

$$= \frac{h_u(s)}{(s - 1/s_1)(s - 1/s_2)(s - m\mu/\lambda_H)}. \tag{50}$$

Because $h_u(s)$ is a quadratic polynomial of s, it can be further simplified to (it is easy to check that $m\mu/\lambda_H \neq 1/s_1$ and $m\mu/\lambda_H \neq 1/s_2$):

$$f_u(s) = \frac{a_u}{s - 1/s_1} + \frac{b_u}{s - 1/s_2} + \frac{c_u}{s - m\mu/\lambda_H}, \tag{51}$$

where a_u, b_u, and c_u are constants that can be determined as

$$a_u = f(s)\left(s - 1/s_1\right)|_{s=1/s_1}, \quad b_u = f(s)\left(s - 1/s_2\right)|_{s=1/s_2}, \quad c_u = f(s)\left(s - m\mu/\lambda_H\right)|_{s=m\mu/\lambda_H}. \tag{52}$$

Because $s_1 > 1$, we must have $a_u = 0$, which yields:

$$\left[(m\mu + \lambda_H + \lambda_L)/s_1 - m\mu\right]u_m - (m\mu/s_1)u_{m+1} = \frac{\pi_m}{m\mu/\lambda_H - 1/s_1}/s_1^2. \tag{53}$$

Expanding (51), and using (53), we obtain the tail of u:

$$u_{m+q} = -b_u s_2^{1+q} - c_u\left(\lambda_H/(m\mu)\right)^{1+q}. \tag{54}$$

And the rest of the vector U can be solved as follows:

$$\begin{pmatrix} u_0 \\ \cdots \\ u_i \\ \cdots \\ u_m \\ u_{m+1} \end{pmatrix} = \left[\begin{pmatrix} -\omega_0 & \bar{\lambda} & & & & \\ & \cdots & & & & \\ & \mu_i & -\omega_i & \lambda_H + p_i\lambda_L & & \\ & & & \cdots & \lambda_H & \\ & & & \mu_m & -\omega_m & \frac{\omega_m}{s_1} - \mu_m \\ & & & & \mu_{m+1} & -\frac{\mu_{m+1}}{s_1} \end{pmatrix}^T\right]^{-1} \begin{pmatrix} \pi_0 \\ \cdots \\ \pi_i \\ \cdots \\ \pi_m \\ \frac{\pi_m/s_1^2}{\frac{\mu_{m+1}}{\lambda_H} - \frac{1}{s_1}} \end{pmatrix}. \tag{55}$$

$U \cdot V$: (46) and (54) allow us to express $U \cdot V$ as

$$U \cdot V = \sum_{i=0}^{m-1} u_i v_i + b_u a_v \frac{s_2}{s_1 - s_2} + c_u a_v \frac{\lambda_H}{m\mu\, s_1 - \lambda_H} + b_u c_v \frac{s_2}{1 - s_2}$$
$$+ c_u c_v \frac{\lambda_H}{m\mu - \lambda_H} - b_u d_v \frac{s_2}{(s_1 - s_2)^2} - c_u d_v \frac{m\mu\lambda_H}{(m\mu s_1 - \lambda_H)^2}. \tag{56}$$

E.2 k-step correlation

Our approach to the 1-step correlation can be extended to the k-step correlation for any k:

$$f^k = \Gamma F^{k-1} + AF^k. \tag{57}$$

The Laplace transform of this gives us:

$$\hat{f}^k(s) = \frac{\Gamma}{s}\hat{f}^{k-1}(s) + \frac{A}{s}\hat{f}^k(s) \quad \Rightarrow (sI - A)\hat{f}^k(s) = \Gamma\hat{f}^{k-1}(s). \tag{58}$$

This easily gives us that

$$\hat{f}^k(s)|_{s=0} = -A^{-1}\Gamma\hat{f}^{k-1}(s)|_{s=0} = \left(-A^{-1}\Gamma\right)^k.$$

To avoid double superscripts, we will use the overhead dot to represent derivatives. Then differentiating (58) we also obtain:

$$\begin{aligned}
\hat{f}^k(s)|_{s=0} + (sI - A)\dot{\hat{f}}^k(s)|_{s=0} &= \Gamma\dot{\hat{f}}^{k-1}(s)|_{s=0} && (59)\\
\left(-A^{-1}\Gamma\right)^k - A\dot{\hat{f}}^k(0) &= \Gamma\dot{\hat{f}}^{k-1}(0) && (60)\\
\dot{\hat{f}}^k(0) &= A^{-1}\left(-A^{-1}\Gamma\right)^k - A^{-1}\Gamma\dot{\hat{f}}^{k-1}(0). && (61)
\end{aligned}$$

Given that $\dot{\hat{f}}^1(0) = \hat{f}'(0) = -A^{-2}\Gamma$, this recursive definition can be used to find the k-step serial correlations. The details are complex and are omitted here.

F The Pooled-Overflow System when $\mu_{\mathbf{H}}$ May Differ From $\mu_{\mathbf{L}}$

In this appendix, we consider a more general class of systems in which μ_H may differ from μ_L. (Systems in which $\mu_H = \mu_L$ represent a special case.) We note that, when $\mu_H \neq \mu_L$, our analysis of the dedicated-overflow and inverted-V schemes is not affected, since capacity for type-H and type-L calls is partitioned.

Because the pooled-overflow scheme shares in-house capacity across both types of calls, its analysis does become more complex, however. Appendix F.1 proves that, among type-H priority policies, there are stationary, type-H work-conserving policies that maximize the in-house throughput of type-L calls. It then formulates a linear program with $O(m_I^2)$ variables and $O(m_I^2)$ constraints that identifies such an optimal policy. Appendix F.2 extends Appendix D's analysis of the moments of the inter-overflow time CV to include cases in which μ_H may not equal μ_L.

For notational convenience, we drop the subscript, I, from the number of in-house CSRs, m_I. Thus, the total number of in-house CSRs is referred to as "m" below.

F.1 An LP for Finding Optimal Type-H Priority Policies

In Section 4.1, we derived the optimality of (L, p_L) policies using the NLP (3)–(6). For the more general case in which μ_H may not equal μ_L, we use a different, linear programming (LP) approach to solve the constrained optimization problem.

As before, we define the state and action spaces, S and A, in terms of numbers of calls. The dimensionality of the state space in the LP will be greater than that in the NLP of Section 4.1, however. Instead of simply tracking the total number of calls in the system, we now must mark how many CSRs are busy with type-H and type-L calls, respectively. Furthermore, we will track system performance before (rather than after) action, and we need to record the presence or absence of an arriving type-L call.

Formally, we now define each state of the system, $s \in S$, at discrete event epoch t as follows. Let

$$S = \{(i^H, i^L, q^H, q^L) : i^H, i^L \geq 0;\ i^H + i^L \leq m;\ q^H \geq 0;\ q^L \in \{0, 1\}\} \tag{62}$$

and $s_t \in S$ be the number of calls in service or in queue at t: i^H represents the number of type-H calls in service; i^L, the number of type-L calls in service; and q^H, the number of type-H calls in queue. If the event at t is an arrival of a type-L call, then $q^L = 1$; otherwise $q^L = 0$.

Again, $s_t = (i_t^H, i_t^L, q_t^H, q_t^L)$ represents the state of the system at transition t, *before* any action is taken. In contrast, the analysis of Section 4.1 focused on the states after actions are taken, the *after-action states*. Nevertheless, in the following analysis, it will sometimes be useful for us to refer to after-action states. We do so by denoting the after-action states with an overbar above the state descriptor: that is, after-action states are $\bar{s} \in \bar{S}$.

Because arriving type-L calls are either immediately put into service or routed to the outsourcer, there never exists a type-L call in queue after action, and we drop element $\bar{q}^L$ from the descriptor. Thus the after-action state space becomes

$$\bar{S} = \{(\bar{\imath}^H, \bar{\imath}^L, \bar{q}^H) : \bar{\imath}^H, \bar{\imath}^L \geq 0;\ \bar{\imath}^H + \bar{\imath}^L \leq m;\ \bar{q}^H \geq 0\}\,, \tag{63}$$

and $\bar{s}_t = (\bar{\imath}_t^H, \bar{\imath}_t^L, \bar{q}_t^H)$ denotes the after-action state at transition t.

In any state, a system controller may put one or more calls into service, or it may do nothing. Accordingly, let c^H and c^L be the numbers of type-H and type-L calls put into service at an arbitrary event epoch. We define the set of feasible actions in state $s \in S$ to be

$$A_s = \{(c^H, c^L) :\ c^H, c^L \geq 0;\ c^H + c^L \leq m - (i^H + i^L);\ c^H \leq q^H;\ c^L \leq q^L\}, \tag{64}$$

and the action taken at time t to be $a_t \in A_{s_t}$. We denote the superset of all feasible actions as $A = \{(c^H, c^L) : 0 \leq c^H \leq m;\ c^L \in \{0, 1\};\ (c^H + c^L) \leq m\} \supseteq A_s$ for all $s \in S$. Observe that A is finite.

A *policy* is a rule that the system controller uses to choose an action to take at each event epoch t. Let $\mathcal{H}_t = \{(s_0, a_0), \ldots, (s_{t-1}, a_{t-1}) \cup s_t\}$, be the *history* of the system up to event epoch t. Then

a *non-anticipating* policy is a rule which, given $\mathcal{H}_t$, chooses an action a_t, possibly at random, among the actions of A_{s_t}. A type-H *priority* policy never puts type-L calls into service when there is a type-H call in queue. We define Π to be the class of all non-anticipating, type-H priority policies.

The objective is to find a policy, $\pi \in \Pi$, that maximizes the rate at which type-L calls are served in house. For a given before-action state, $s \in S$, define the *reward* associated with action a to be $R(s,a)$. We let $R(s,a)$ equal the number of type-L calls put into service. In turn, we define

$$\bar{R}_\pi(s) \stackrel{\text{def}}{=} \liminf_{n\to\infty} \frac{1}{n} \mathsf{E}_\pi \left[\sum_{t=0}^{n-1} R(s_t, a_t) | s_0 = s \right] \tag{65}$$

to be the long run average rate at which a policy $\pi \in \Pi$ serves type-L calls.

Because the state space is defined in terms of system occupancy, it is convenient for us to account for type-H calls' service-level in terms of occupancy as well. We denote by $D(s,a)$ the "delay cost" associated with state-action combination (s,a), and we let $D(s_t, a_t)$ be some non-negative function of the number of type-H calls remaining in queue after the policy's action at t.

In particular, we let $d(\bar{q})$ be the delay-cost function associated with the after-action queue length q, and let the constraint on the type-H service level be that the long-run average delay cost be no more than D^*. For the delay-cost function $d(\bar{q})$, we assume:

Assumption 1

i) $d(0) = 0$ *and* $d(\bar{q})$ *is nondecreasing in* $\bar{q}$*;*

ii) $\sup_{\bar{q}} d(\bar{q}) > D^*$*; and*

iii) $\tilde{d}(\alpha) \stackrel{\text{def}}{=} \sum_{\bar{q}=0}^{\infty} \alpha^{\bar{q}}\, d(\bar{q}) < \infty$ *for all* $\alpha \in (0,1)$.

Item (i) ensures that the cost increases as the backlog grows. Let $H(n)$ be the number of type-H arrivals put into serve in the first n transitions. Then together items (i) and (ii) imply that any sample path for which $\lim_{n\to\infty} H(n)/n < \lambda_H$ is also one for which $\lim_{n\to\infty} d(\bar{q})/n > D^*$, and thus violates the service level constraint. Finally, item (iii) defines the generating function $\tilde{d}$ and implies that the service-level cost of occupancy grows sub-exponentially. All of these restrictions are satisfied by formulations of standard service-level constraints, such as bounds on expected occupancy and on the tail distribution of occupancy.

In turn, we define

$$\bar{D}_\pi(s) \stackrel{\text{def}}{=} \limsup_{n\to\infty} \frac{1}{n} \mathsf{E}_\pi \left[\sum_{t=0}^{n-1} D(s_t, a_t) | s_0 = s \right], \tag{66}$$

and we require that $\bar{D}_\pi(s) \le D^*$, where D^* is an exogenously defined upper bound on the average backlog cost.

Although $D(s,a)$ and $\bar{D}_\pi(s)$ are defined as functions of queue occupancy, rather than delay in queue, in many cases they are equivalent. In particular, given the use of a stationary policy, one can use Little's Law to translate between several common versions of occupancy and delay constraints. (See Gans and Zhou [21].)

Using these definitions of reward and cost, we can formally state the problem of maximizing the throughput of type-L calls, subject to the service level constraint on type-H queue, as follows:

$$\sup_{\pi\in\Pi} \quad \bar{R}_\pi(s) \qquad \text{s.t.} \quad \bar{D}_\pi(s) \quad \leq \quad D^* \,. \tag{67}$$

Any policy π that satisfies the constraint in (67) is called *feasible*. If it also achieves the supremum in (67), then it is *optimal* or solves the constrained optimization problem (COP).

In turn, we can characterize an easily computable class of policies that solves the COP. For the general case, in which μ_H may not equal μ_L, there exist optimal policies $\pi \in \Pi$ that are type-H work conserving. These *work-conserving* policies never allow a type-H call to queue when there is an idle CSR. Furthermore, among type-H priority, type-H work-conserving policies, there are, in turn, *stationary* (history-independent) policies that are optimal. We call the class of stationary, type-H priority, type-H work-conserving policies Π^*. Below, we will show that, among all policies in Π, there exist policies within Π^* that maximize the throughput rate of type-L calls.

The first step is to show that, by considering only type-H work conserving policies we do not unintentionally degrade system performance. The lemma can be proved using the argument that proves Lemma 7 in [21].

Lemma 3 *Suppose there exists a feasible type-H priority policy, π. Then there exists a feasible type-H priority, type-H work-conserving policy π with the same throughput as π.*

The lemma allows us to significantly simplify the problem. If a policy gives priority to and is work conserving with respect to type-H calls, then it must be the case that $\bar{\imath}^H < m \implies \bar{q}^H = 0$ and $\bar{q}^H > 0 \implies \bar{\imath}^H = m$. This allows us to reduce the state-space, any eliminating a separate identifier for the state of the type-H queue. We therefore let

$$S = \{(i^H, i^L, q^L) : i^H \geq 0;\ 0 \leq i^L \leq m;\ q^L \in \{0,1\}\}, \tag{68}$$

where i^L and q^L are defined as before. Similarly, the after-action state space becomes

$$\bar{S} = \{(\bar{\imath}^H, \bar{\imath}^L) : \bar{\imath}^H \geq 0;\ 0 \leq \bar{\imath}^L \leq m\}. \tag{69}$$

Actions also simplify. At time 0, a type-H work-conserving policy puts as many type-H calls into service as possible. Then at subsequent event epochs there will never be the opportunity to put more than one type-H call into service at a time – otherwise the policies will not have been work conserving. By the same argument, it will never be the case that a type-H and type-L call are put

into service at the same time. Therefore, at any event after which there exists a type-H call in queue, there is only one optimal action – put the type-H call into service. We can embed this action into the state transitions of the Markov chain.

Without loss of generality, we define the time scale so that $\lambda_H + \lambda_L + m(\mu_H + \mu_L) = 1$. Therefore, we may view transition rates as probabilities. For example, $\lambda_H = \frac{\lambda_H}{\lambda_H + \lambda_L + m(\mu_H + \mu_L)}$ equals the expected number of type-H arrivals per period, as well as the probability that the next event is a type-H arrival.

The state transitions of the Markov chain are as follows:

$$(i^H_{t+1}, i^L_{t+1}, q^L_{t+1}) \quad = \quad \begin{cases} (\bar{i}^H_t + 1, \bar{i}^L_t, 0), & \text{w. p. } \lambda_H; \\ (\bar{i}^H_t, \bar{i}^L_t, 1), & \text{w. p. } \lambda_L; \\ (\bar{i}^H_t - 1, \bar{i}^L_t, 0), & \text{w. p. } \min\{\bar{i}^H_t, m - \bar{i}^L_t\}\, \mu_H; \\ (\bar{i}^H_t, \bar{i}^L_t - 1, 0), & \text{w. p. } \bar{i}^L_t\, \mu_L; \text{ and} \\ (\bar{i}^H_t, \bar{i}^L_t, 0), & \text{w. p. } m(\mu_H + \mu_L) - \min\{\bar{i}^H_t, m - \bar{i}^L_t\}\, \mu_H - \bar{i}^L_t \mu_L. \end{cases} \tag{70}$$

Here, $\min\{\bar{i}^H_t, m - \bar{i}^L_t\}$ represents the number of type-H calls in service, after action, at epoch t. In turn, feasible actions reduce to

$$A_s = \{c^L : c^L \in \{0, 1\};\ c^L \leq (m - (i^H + i^L))^+\}, \tag{71}$$

where $(m - (i^H + i^L))^+$ denotes the number of idle servers. In turn, $A = \{0, 1\}$.

Thus, the complexity of the routing problem has been reduced substantially. There exist only $\frac{m(m+1)}{2}$ states in which a type-L call has arrived to a system with at least one CSR free, and in each of these states there are only two feasible actions: accept or reject the call. When all m CSRs become busy, there are no decisions to be made, and the evolution of the system states follows a Markov chain.

Furthermore, using arguments analogous to those used in [21], we can show that each type-H priority, type-H work conserving policy that is stationary and deterministic also: i) induces a single, positive recurrent class of states, with expected absorption time into that class that is finite; ii) has limiting state-action frequencies which correspond to the stationary distribution of the induced Markov chain; and iii) has uniformly integrable one-period revenues. Therefore, we can appeal to Theorem 7.1 in Altman and Schwartz [2] to show that there exist stationary, type-H priority, type-H work conserving policies that are optimal:

Lemma 4 *If there exists a policy $\pi \in \Pi$ that is feasible, then exists a policy $\pi \in \Pi^*$ that is optimal.*

The result in [2] also implies that we can formulate an LP whose optimal solution identifies the optimal policy.

Because the set of states in which all m CSRs are busy is infinite, the LP has an infinite set of balance equations. Nevertheless, we can use algebraic substitution to develop closed-form expressions for essential quantities related to these tail states and make the LP finite. More specifically, under

the following assumption, we can collapse the states of the Markov chain when all m CSRs are busy into a set of $O(m^2)$ linear equations.

Assumption 2

i) *The pool-H queue is stable:* $\rho < 1$.

ii) *Either* $\mu_H \leq \mu_L$ *or* $\lambda_H \neq m(\mu_H - \mu_L)$.

The first condition allows us to show that any policy $\pi \in \Pi^*$ achieves a steady state with uniformly bounded costs. The second condition, which occurs almost surely, allows us to use a set of generating functions to collapse the "tail" states of the Markov chain induced by a stable, stationary policy.

Remark 1 Part (ii) of Assumption 2 is a technical assumption that assures that the denominator of the generating function used in Lemma 5, below, has distinct roots. This simplifies the expression of the partial fraction expansion. In the case that part (ii) of Assumption 2 is violated, we can still use the generating-function approach to formulate an analogous, finite LP, albeit one with more complex expressions. For details, see [21]. Because the assumption is essentially never violated, however, we omit the treatment of this case, here.

Let $\xi_{i,j,k}(a)$ be the stationary probability of entering (before-action) state $(i^H = i, i^L = j, q^L = k)$ and taking action $a \equiv c^L$. Then the following lemma shows that the tail probabilities can be conveniently characterized:

Lemma 5 (Gans and Zhou [21])
Let

$$\gamma_1(j) \stackrel{\text{def}}{=} \frac{\lambda_H}{\lambda_H + (m-j)\mu_H + j\mu_L}, \tag{72}$$

$$\gamma_2(j) \stackrel{\text{def}}{=} \frac{(m-j)\mu_H}{\lambda_H + (m-j)\mu_H + j\mu_L}, \text{ and} \tag{73}$$

$$\gamma_3(j) \stackrel{\text{def}}{=} \frac{(j+1)\mu_L}{\lambda_H + (m-j)\mu_H + j\mu_L}. \tag{74}$$

If the conditions of Assumption 2 hold, then for each j the quadratic equation

$$g(j,z) \stackrel{\text{def}}{=} \gamma_2(j)z_j^2 - z_j + \gamma_1(j) = 0 \tag{75}$$

has roots $z_j \leq z_j'$ with $0 < z_j < 1$. In turn, for any policy $\pi \in \Pi^*$ there exists constants $a_{j,l}$ such that

$$a_{m,m} = \sum_{a=0}^{1} \xi_{0,m-a,a}(a) \tag{76}$$

$$a_{j,j} = \sum_{a=0}^{1} \xi_{m-j,j-a,a}(a) + \frac{\gamma_3(j)}{\gamma_2(j)} \sum_{a=0}^{1} \xi_{m-j-1,j+1-a,a}(a)$$

$$+ \frac{\gamma_3(j) z_j}{\gamma_2(j)(z'_j - z_j)} \sum_{k \geq j+1} \frac{a_{j+1,k}}{1 - z_k/z_j} - \frac{\gamma_3(j) z'_j}{\gamma_2(j)(z'_j - z_j)} \sum_{k \geq j+1} \frac{a_{j+1,k}}{1 - z_k/z'_j}$$

$$\forall\ 0 \leq j \leq m-1 \tag{77}$$

$$a_{j,k} = \frac{-\gamma_3(j)}{\gamma_2(j)(1 - z_j/z_k)(1 - z'_j/z_k)} a_{j+1,k} \qquad \forall\ 0 \leq j < k \leq m, \tag{78}$$

and

$$\sum_{l=0}^{1} \xi_{m-j+q,j,l}(0) = \sum_{k=j}^{m} a_{j,k} z_k^q, \qquad \forall 0 \leq j \leq m, \forall q \geq 1. \tag{79}$$

Thus, given Assumption 2, for any policy $\pi \in \Pi^*$, the tail states of the Markov chain can be represented as mixtures of geometric series, and we can use (79) to collapse the tail and formulate a finite-dimension LP.

We define the LP that finds a policy $\pi \in \Pi^*$ that maximizes the throughput of type-L calls. Specifically, let $s = (i', j', k')$ be shorthand for a given before-action state, and let $K(i,j)$ be the set of state-action pairs (s,a) that land the system in *after-action* state (i,j). If $K(i,j) = \emptyset$ then interpret the associated summation as equal to zero. Then, using (72)-(79) from Lemma 5, the following LP finds a constrained optimal policy:

$$\max \sum_{i+j<m} \xi_{i,j,1}(1) \tag{80}$$

s. t.

$$\xi_{i,j,0}(0) = \lambda_H \sum_{(s,a) \in K(i-1,j)} \xi_s(a) + (i+1)\mu_H \sum_{(s,a) \in K(i+1,j)} \xi_s(a) + (j+1)\mu_L \sum_{(s,a) \in K(i,j+1)} \xi_s(a)$$
$$+ ((m-i)\mu_H + (m-j)\mu_L) \sum_{(s,a) \in K(i,j)} \xi_s(a) \qquad 0 \leq i+j \leq m-1 \tag{81}$$

$$\sum_{a=0}^{1} \xi_{i,j,1}(a) = \lambda_L \sum_{(s,a) \in K(i,j)} \xi_s(a) \qquad 0 \leq i+j \leq m-1 \tag{82}$$

$$\xi_{m-j,j,0}(0) = \lambda_H \sum_{(s,a) \in K(m-j-1,j)} \xi_s(a) + (j+1)\mu_L \sum_{k=j+1}^{m} a_{j+1,k} z_k + (m-j)\mu_H \sum_{k=j}^{m} a_{j,k} z_k$$
$$+ (j\mu_H + (m-j)\mu_L) \sum_{(s,a) \in K(m-j,j)} \xi_s(a) \qquad 0 \leq j \leq m-1 \tag{83}$$

$$\xi_{0,m,0}(0) = m\mu_H \sum_{(s,a) \in K(0,m)} \xi_s(a) \tag{84}$$

$$\xi_{m-j,j,1}(0) = \lambda_L \sum_{(s,a) \in K(m-j,j)} \xi_s(a) \qquad 0 \leq j \leq m \tag{85}$$

$$a_{m,m} = \sum_{a=0}^{1} \xi_{0,m-a,a}(a) \tag{86}$$

$$a_{j,j} = \sum_{a=0}^{1} \xi_{m-j,j-a,a}(a) + \frac{\gamma_3(j)}{\gamma_2(j)} \sum_{a=0}^{1} \xi_{m-j-1,j+1-a,a}(a)$$

$$+ \frac{\gamma_3(j) z_j}{\gamma_2(j)(z'_j - z_j)} \sum_{k \geq j+1} \frac{a_{j+1,k}}{1 - z_k/z_j} - \frac{\gamma_3(j) z'_j}{\gamma_2(j)(z'_j - z_j)} \sum_{k \geq j+1} \frac{a_{j+1,k}}{1 - z_k/z'_j}$$

$$\forall\ 0 \leq j \leq m-1 \qquad (87)$$

$$a_{j,k} = \frac{-\gamma_3(j)}{\gamma_2(j)(1 - z_j/z_k)(1 - z'_j/z_k)} a_{j+1,k} \qquad \forall\ 0 \leq j < k \leq m \qquad (88)$$

$$\sum_{j=0}^{m} \sum_{k=j}^{m} a_{j,k} \tilde{d}(z_k) \leq D^* , \qquad (89)$$

$$\sum_{i+j \leq m} \sum_{k=0}^{1} \sum_{a \in A_s} \xi_{i,j,k}(a) + \sum_{j=0}^{m} \sum_{k=j}^{m} a_{j,k} \frac{z_k}{1 - z_k} = 1 \qquad (90)$$

$$\xi_s(a) \geq 0 \qquad \forall s \in S, a \in A_s . \qquad (91)$$

Here, the objective function (80) maximizes the rate at which type-L calls are put into service. The constraints (81)–(88) are the system's balance constraints. Of these, (86)–(88) define the $\xi_{i,j,k}$s associated with boundary states in terms of the geometric series of Lemma 5. Constraint (89) ensures that the service-level is met and (90) that probabilities sum to one. Constraints (91) ensure that the probabilities are non-negative. (Note that the LP formulation must drop one redundant balance equation. See §8.8 in Puterman [39].)

Thus, an optimal policy $\pi \in \Pi^* \subset \Pi$ can be found via the solution of the LP. The following theorem summarizes and formalizes all the results so far:

Theorem 3

Suppose Assumptions 1 and 2 hold. Then there exists an LP (80)–(91) with $O(m^2)$ variables and $O(m^2)$ constraints which is feasible if and only if there exists a policy $\pi \in \Pi$ that is feasible as well. The optimal solution of the LP determines a policy $\pi \in \Pi^*$ that solves the COP (67).

We note that, although the proof of Theorem 3 is a direct analogue to that used when $\lambda_L = \infty$, differences in the two systems' type-L call dynamics leads the LP in the current paper to differ from – and to have fewer decision variables than – the LP in [21]. In particular, the fact that $\lambda_L < \infty$ implies that at most one type-L call can be put into service at a time. This is not the case when $\lambda_H = \infty$, however. Therefore, the current LP has $\mathrm{O}(m^2)$, rather than the $\mathrm{O}(m^3)$, decision variables.

F.2 Inter-Overflow Time CV in the Pooled-Overflow System

Next, we extend the results regarding the moments of inter-overflow time, in particular Proposition 4, to the case in which μ_H may not equal μ_L. Rather than limiting the analysis to randomized threshold-reservation policies, (L, p_L), our analysis in this section holds for the overflow process that results from any stationary, type-H priority, type-H work conserving policy $\pi \in \Pi^*$. For example, the policies can be derived using Theorem 3 and the associated LP formulation (80)-(91).

For brevity, let $\tilde{T}_{\bar{\jmath},\bar{q}}$ denote the time to the next type-L call "overflow" starting in after-action state $(m-\bar{\jmath}+\bar{q},\bar{\jmath})$, and let $T_{\bar{\jmath},\bar{q}}(\theta)$ be its Laplace transform. Then,

Proposition 5

$$T_{\bar{\jmath},\bar{q}}(\theta)=\sum_{k=0}^{\bar{\jmath}}a_{\bar{\jmath},k}z_k(\theta)^{\bar{q}}+\frac{\lambda_L}{\lambda_L+\theta},$$

where

$$z_k(\theta)=\frac{(\lambda_H+\lambda_L+(m-k)\mu_H+k\mu_L+\theta)-\sqrt{(\lambda_H+\lambda_L+(m-k)\mu_H+k\mu_L+\theta)^2-4(m-k)\mu_H\lambda_H}}{2\lambda_H}$$

and $a_{\bar{\jmath},k}$, $\forall\ 0\le k\le\bar{\jmath}\le m$, *are constants uniquely determined by* $\{T_{\bar{\jmath},0}(\theta),T_{\bar{\jmath},1}(\theta)\}$.

Proof

The proof will proceed in three steps:

1. Show that
$$T_{j,\bar{q}}(\theta)=\sum_{k=0}^{j}a_{j,k}(z_k(\theta))^{\bar{q}}+a'_j(z'_j(\theta))^{\bar{q}}+\frac{\lambda_L}{\lambda_L+\theta},\forall j,\bar{q}, \tag{92}$$
for the z_ks given in the proposition statement and some $z'_j>1$.

2. Show that $\lim_{\bar{q}\to\infty}T_{j,\bar{q}}(\theta)=\frac{\lambda_L}{\lambda_L+\theta},\forall\ j$.

3. From the first two steps, we must have $a'_j=0,\forall j$, because $z'_j>1$.

Let $f_j(z,\theta)\stackrel{\text{def}}{=}\sum_{\bar{q}=0}^{\infty}T_{j,\bar{q}}(\theta)z^{\bar{q}}$, then it suffices to show

$$f_j(z,\theta)=\sum_{k=0}^{j}\frac{a_{j,k}}{1-z_k(\theta)z}+\frac{a'_j}{1-z'_j(\theta)z}+\frac{\lambda_L/(\lambda_L+\theta)}{1-z}\qquad\forall\ j.$$

In what follows, when it is clear from the context we will again suppress the θ notation to simplify the exposition.

Again, we let $\tilde{T}_s$ denote the time to next overflow if the in-house call center is in state s. Based on (7), these random variables are sufficient to specify the general inter-overflow time distribution. Moreover, it is sufficient for us to focus on the states where all CSRs are busy. For brevity, let $\tilde{T}_{j,\bar{q}}$ denote the time to next overflow if the in house call center is in state $(m-j,j,\bar{q})$. Moreover, let $\omega_j=\lambda_H+\lambda_L+(m-j)\mu_H+j\mu_L$, and let $\tilde{e}_{\omega_j}$ denote an exponential random variable with rate ω_j. Then we have the following equations, which are analogues to (8) for the case in which $\mu_H=\mu_L$:

$$\tilde{T}_{j,\bar{q}}\ =\ \tilde{e}_{\omega_j}+\begin{cases}\tilde{T}_{j,\bar{q}-1} & \text{w.p. } (m-j)\mu_H/\omega_j\\ 0 & \text{w.p. } \lambda_L/\omega_j\\ \tilde{T}_{j,\bar{q}+1} & \text{w.p. } \lambda_H/\omega_j\\ \tilde{T}_{j-1,\bar{q}-1} & \text{w.p. } j\mu_L/\omega_j\end{cases}\qquad\forall\ \bar{q}\ge 1 \tag{93}$$

and

$$\tilde{T}_{0,\bar{q}} = \tilde{e}_{\omega_0} + \begin{cases} \tilde{T}_{0,\bar{q}-1} & \text{w.p. } m\mu_H/\omega_0 \\ 0 & \text{w.p. } \lambda_L/\omega_0 \\ \tilde{T}_{0,\bar{q}+1} & \text{w.p. } \lambda_H/\omega_0 \end{cases} \qquad \forall\ \bar{q} \geq 1. \tag{94}$$

If we let $\gamma_1(j) = \frac{(m-j)\mu_H}{\omega_j+\theta}$, $\gamma_2(j) = \frac{\lambda_H}{\omega_j+\theta}$, $\gamma_3(j) = \frac{\lambda_L}{\omega_j+\theta}$, and $\gamma_4(j) = \frac{j\mu_L}{\omega_j+\theta}$, then the above transition equations result in the following equations for the corresponding Laplace transforms:

$$\begin{aligned} T_{j,\bar{q}} &= \frac{(m-j)\mu_H}{\omega_j+\theta}T_{j,\bar{q}-1} + \frac{\lambda_L}{\omega_j+\theta} + \frac{\lambda_H}{\omega_j+\theta}T_{j,\bar{q}+1} + \frac{j\mu_L}{\omega_j+\theta}T_{j-1,\bar{q}-1} \\ &= \gamma_1(j)T_{j,\bar{q}-1} + \gamma_3(j) + \gamma_2(j)T_{j,\bar{q}+1} + \gamma_4(j)T_{j-1,\bar{q}-1}\ \forall \bar{q} \geq 1, j \geq 1 \end{aligned} \tag{95}$$

and

$$T_{0,\bar{q}} = \frac{m\mu_H}{\omega_0+\theta}T_{0,\bar{q}-1} + \frac{\lambda_L}{\omega_0+\theta} + \frac{\lambda_H}{\omega_0+\theta}T_{0,\bar{q}+1}. \tag{96}$$

Note that we have $\gamma_1(j) + \gamma_2(j) < 1,\ \forall j$. So if we let z_j and z'_j be the two roots of $\gamma_2(j)z^2 - z + \gamma_1(j) = 0$ such that $z_j \leq z'_j$, then we must have $0 < z_j < 1 < z'_j, \forall j$.

We first solve (96). Using the same generating function approach we used to prove Proposition 4, we obtain the following solution:

$$T_{0,\bar{q}} = a_{0,0}z_0^{\bar{q}} + a'_{0,0}z_0^{'\bar{q}} + \frac{\lambda_L}{\lambda_L+\theta}.$$

To complete the first step, we then use induction on j (starting from $j = 0$) to show that the solution to (95) is (92). The induction proof is similar to that in [21] for Lemma 10. Moreover, the proof of steps 2 and 3 are similar to that for Proposition 4; we will not repeat them here. □

As in the case of $\mu_H = \mu_L$, Proposition 5 allows us to reduce the infinite number of linear equations (of the Laplace transforms) to a finite number, and we can solve these linear equations completely. By (7), we obtain the Laplace transform of the inter-overflow time distribution. Even though the Laplace transform may be difficult to invert, we can repeatedly differentiate it to obtain its moments, just as we did in (25) and (27) for the case of $\mu_H = \mu_L$.