Comparing LP Bounds for Queueing Networks

Matthew C. Russell, Jeff Fraser, Stephen Rizzo, Michael H. Veatch

May 26, 2009

Abstract

We consider performance of open multiclass queueing networks under general scheduling policies. Two linear program (LP) performance bounds are compared, the achievable region method and the approximate linear program with quadratic approximation architecture. The latter is shown to give tighter upper and lower bounds.

I. INTRODUCTION

Efficient scheduling of queueing networks is an important topic in applications including manufacturing systems and communication networks. However, because of the intractability of their optimal control formulations and the complexity of optimal policies, most analysis has focused on simple policies or stability.

Given these limitations, one useful approach has been to compute bounds on the performance of any policy. This paper compares two of the proposed bounds, the achievable region method [1], [4] and approximate linear programming (ALP) with quadratic approximation architecture [6], [7], [8]. The ALP is shown in [6] to give a tighter upper bound than the achievable region method for reentrant lines. The main contribution of this paper is to extend their result to the more general setting of open multiclass networks with probabilistic routing. We also show that the ALP gives a tighter lower bound than the achievable region method.

Matthew C. Russell is with the Department of Mathematics, Taylor University, Upland, IN 46989. Email: matthew_russell@taylor.edu Phone: 765-998-4206 FAX: 765-998-4650

Jeff Fraser, Stephen Rizzo, Michael H. Veatch are with the Department of Mathematics, Gordon College, Wenham, MA 01984. Email: jeff.fraser@gordon.edu, stephen.rizzo@gordon.edu, mike.veatch@gordon.edu Phone: 978-867-4375 FAX: 978-867-4666

Michael H. Veatch is the corresponding author.

Both methods require only the solution of a linear program. Although the size of the LP is exponential in the number of buffers, it can be solved for fairly large networks. In [9], the ALP is solved for examples with up to 40 buffers. Solving the achievable region LP requires roughly the same effort: it has slightly fewer constraints and variables. Thus, the ALP is a practical method of bounding performance of any policy, and appears preferable to the achievable region method.

These methods bound the mean numbers of jobs in the system or, given a linear holding cost, the average cost. The lower bound on average cost of any policy is useful for performance analysis. The upper bound on average cost of nonidling polices has been used to establish stability [5]. Our proof is based on the dual relationship in [5] between their *performance* LP (the upper bound achievable region LP) and their *drift* LP. We also extend this relationship from reentrant lines to probabilistic routing.

Additional justifications for using the ALP approach in general are given in [2] and [3]. The accuracy of the ALPs used here are tested in [8]. The ALP approach also has the advantage that the differential cost approximation can be refined, resulting in a larger LP and tighter bound; see [7] and [8].

The performance and drift LPs are stated in Section II 2, with the derivation of the drift LP postponed to Appendix A. The ALP and the result comparing them are given in Section III 3. Upper bounds are stated in Section IV 4.

II. Performance and Drift LPs

The achievable region method provides an LP bound on average cost as follows. Consider the network and notation in [8]. Jobs arrive and move through different classes before exiting the network. There are n job classes and m stations, each of which serves one or more classes. Let $X_i(t)$ be the number of class i jobs at time t, including any that are being processed. Class i jobs are served by station $\sigma(i)$. We define α_i as the exogenous arrival rate to class i, and processing times are assumed to be independently exponentially distributed with mean $1/\mu_i$ in class i. Let λ_i be the effective arrival rate to class i; it is the unique solution to the traffic equation

$$\lambda = \alpha + P^T\lambda, \tag{1}$$

where $P = [p_{ij}]$ is the routing matrix with p_{ij} is the probability that a job finishing service at class i will be routed to class j, independent of all other history. We let p_{i0} represent the probability that a job finishing service at class i leaves the system; it is equal to $1-\sum_j p_{ij}$. For deterministic routing, adopting the conventions that $s(i)$ is the successor of class i, $s(i) = 0$ if class i has no successor, and $s(0) = 0$, we have $p_{i,s(i)} = 1$ and $p_{ij} = 0, j \neq s(i)$. We use the uniformized, discrete-time Markov chain and assume that the potential event rate is $\sum_{i=1}^n (\alpha_i + \mu_i) = 1$.

The network has sequencing control: each server must decide which job class to work on next, or possibly to idle. Preemption is allowed. Let $u_i(t) = 1$ if class i is served at time t and 0 otherwise. We consider only stationary Markov policies, writing $u(t) = u(X(t))$. In a slight abuse of notation, we also let u denote an action. The overall objective is to minimize long-run average cost J, where the cost rate is $c^T x$ in state x. The minimum possible value of J under all policies is denoted by J^*.

Introduce the variables

$$z_{ij} = E[u_i(t)X_j(t)],\ x_j = E[X_j(t)],$$

where the expectation is with respect to the stationary distribution for some stabilizing policy u. The logical constraint $\sum\limits_{i:\sigma(i)=\sigma} u_i(t) \leq 1$ for each server σ leads to the constraint

$$\sum_{i:\sigma(i)=\sigma} z_{ij} - x_j \leq 0 \quad \text{for all } j, \sigma \tag{2}$$

Let $\Delta X_i(t) = X_i(t+1) - X_i(t)$. Defining $\Delta(X_i X_j)$ similarly, under the stationary distribution we must have

$$E[\Delta(X_i(t) X_j(t))] = 0 \text{ for } i \leq j$$

This constraint can be thought of as stationarity with respect to quadratic test functions. Other test functions can be considered, but do not lead to linear constraints. Now,

$$E[\Delta(X_i(t) X_j(t))] = E(\Delta X_i(t) X_j(t)) + E(X_i(t) \Delta X_j(t)) + E(\Delta X_i(t) \Delta X_j(t)).$$

For this network, adopt the conventions that $x_0 = z_{i0} = z_{0j} = 0$. Taking expectations with respect to the transition and then with respect to $X(t)$,

$$E\left(\Delta X_i\left(t\right)X_j\left(t\right)\right)=\alpha_i x_j+\sum_k \mu_k p_{ki} z_{kj}-\mu_i z_{ij}$$

$$E\left(\Delta X_i\left(t\right)\Delta X_j\left(t\right)\right)=\begin{cases}2\lambda_i\left(1-p_{ii}\right) & \text{if } i=j\\ -\left(\lambda_i p_{ij}+\lambda_j p_{ji}\right) & \text{otherwise}\end{cases}$$

The balance constraints, after dividing by 2 on the diagonal, are

$$\alpha_i x_i+\sum_k \mu_k p_{ki} z_{ki}-\mu_i z_{ii}=-\lambda_i\left(1-p_{ii}\right) \qquad \text{for all } i \tag{3}$$

$$\alpha_i x_j+\sum_k \mu_k p_{ki} z_{kj}-\mu_i z_{ij}+\alpha_j x_i+\sum_k \mu_k p_{kj} z_{ki}-\mu_j z_{ji}=\lambda_i p_{ij}+\lambda_j p_{ji} \qquad \text{for } i<j \tag{4}$$

The best performance LP

$$\min_{x,\,z} \quad c^T x$$

$$\text{Subject to:} \quad (2),\ (3),\ (4),\ \text{and } z_{ij}\geq 0$$

gives a lower bound on optimal average cost. Associate the dual variables $w_{\sigma,j}$ with (2) and q_{ij} with (3) and (4). Appendix A shows that the dual of the best performance LP simplifies to the following drift LP:

$$\text{(Drift)} \quad J_{\text{Drift}}=\max_{q_{ij}}\sum_i \lambda_i\left(q_{ii}-\sum_j p_{ij} q_{ij}\right) \tag{5}$$

$$\text{Subject to:} \quad \sum_i\left(\alpha_i q_{ij}+\sum_{k=0}^{n} u_i \mu_i p_{ik}\left(q_{kj}-q_{ij}\right)\right)\geq -c_j \quad \text{for all } j,u \tag{6}$$

with the conventions that $q_{ij}=q_{ji}$ (the lower diagonal variables can be eliminated) and $q_{i0}=q_{0i}=0$. An example of this dual relationship in the case of a two-stage series line can be found in Appendix B.

In the case of deterministic routing, the drift LP is

$$J_{\text{Drift}}=\max_{q_{ij}}\sum_i \lambda_i\left(q_{ii}-q_{i,s(i)}\right) \tag{7}$$

$$\text{Subject to:} \quad \sum_i\left(\alpha_i q_{ij}+u_i\mu_i\left(q_{s(i),j}-q_{ij}\right)\right)\geq -c_j \quad \text{for all } j,u \tag{8}$$

The drift LP for a series line, which is a special case of determinstic routing with $\alpha = (\alpha_1, 0, \ldots, 0)$, $p_{i,i+1} = 1$ for $i < n, p_{n0} = 1,$ and $\lambda_i = \alpha_1$ for all i, is

$$J_{\text{Drift}} = \max_{q_{ij}} \sum_i \alpha_1 \left(q_{ii} - q_{i,i+1}\right)$$

$$\text{Subject to:} \quad \alpha_1 q_{1j} + \sum_i u_i \mu_i \left(q_{i+1,j} - q_{ij}\right) \geq -c_j \quad \text{for all } j, u$$

A series line is a special case of a reentrant line where $\sigma(i_1) \neq \sigma(i_2)$ if $i_1 \neq i_2$. The drift LP above was stated in [4] and [5] for reentrant lines.

III. Comparison with Quadratic Approximate LP

The optimal average cost J^* satisfies the average cost optimality equation

$$J^* + h(x) = \min_u \ (T_u h)(x) \ \text{ for all } x \tag{9}$$

for some function h, where T_u is the dynamic programming operator for policy u and the minimum is over all stationary Markov policies. Because (9) only determines h up to an additive constant, we can set $h(0) = 0$. Instead of finding h from (9), consider the quadratic approximation $h(x) = \frac{1}{2} x^T Q x + px$ where Q is symmetric. Any J, h satisfying the average cost inequality

$$J + h(x) \leq (T_u h)(x) \quad \text{for all } x, u, \tag{10}$$

and a suitable growth condition on h give a lower bound on optimal average cost. The largest of these lower bounds is found by maximizing J. Extending [8] to probabilistic routing, we find that (10) simplifies to the following lower bound approximate linear program (ALP):

$$\text{(ALP)} \quad J_{\text{ALP}} = \max_{J,\, Q,\, p} J \tag{11}$$

$$\text{Subject to:} \quad J \leq d^u \quad \text{for all } u \tag{12}$$

$$c_i^u \geq 0 \quad \text{for all } i, u \tag{13}$$

where

$$d^u = \sum_k \left[u_k \left(c_k^u + \mu_k \sum_{j=0}^n p_{kj} \left(\frac{1}{2} q_{kk} + \frac{1}{2} q_{jj} - q_{kj} + p_j - p_k \right) \right) + \alpha_k \left(\frac{1}{2} q_{kk} + p_k \right) \right] \tag{14}$$

$$c_i^u = c_i + \sum_k \left(\alpha_k q_{ik} + \sum_{j=0}^n u_k \mu_k p_{kj} \left(q_{ji} - q_{ki} \right) \right) \tag{15}$$

For deterministic routing, (11), (12), and (13) still apply, but (14) and (15) can be rewritten as

$$d^u = \sum_i \left[u_i \left(c_i^u + \mu_i \left(\frac{1}{2} q_{ii} + \frac{1}{2} q_{s(i),s(i)} - q_{i,s(i)} + p_{s(i)} - p_i \right) \right) + \alpha_i \left(\frac{1}{2} q_{ii} + p_i \right) \right] \tag{16}$$

$$c_i^u = c_i + \sum_j \left[\alpha_i q_{ij} + u_j \mu_j \left(q_{i,s(j)} - q_{ij} \right) \right] \tag{17}$$

This ALP is also given in [6] for a reentrant line, in which case

$$d^u = \sum_i \left[u_i \left(c_i^u + \mu_i \left(\frac{1}{2} q_{ii} + \frac{1}{2} q_{i+1,i+1} - q_{i,i+1} + p_{i+1} - p_i \right) \right) \right] + \alpha_1 \left(\frac{1}{2} q_{11} + p_1 \right)$$

$$c_i^u = c_i + \alpha_1 q_{1i} + \sum_j u_j \mu_j \left(q_{i,j+1} - q_{ij} \right)$$

Theorem 1. *(ALP) provides an equal or better bound on J^* than (Drift); i.e. $J_{ALP} \geq J_{Drift}$.*

Proof: Take any Q satisfying (Drift). Since (6) is equivalent to the (ALP) constraint (13), we need only to find a choice of p such that the constraint (12) is satisfied. Take J as the objective function value from (5) and substitute into (12). Then $J \leq d^u$ becomes

$$\sum_i \lambda_i \left(q_{ii} - \sum_j p_{ij} q_{ij} \right) \leq \sum_i \left[u_i \left(c_i^u + \mu_i \sum_{j=0}^n p_{ij} \left(\frac{1}{2} q_{ii} + \frac{1}{2} q_{jj} - q_{ij} + p_j - p_i \right) \right) + \alpha_i \left(\frac{1}{2} q_{ii} + p_i \right) \right]. \tag{18}$$

By (13), the smallest c_i^u can be is 0. Thus, we set $c_i^u = 0$ for all i, u and remove it from the equation. Let $n_{ij} = \left[(I-P)^{-1} \right]_{ij}$ be the expected number of visits of a job to class j, given that the job is currently in class i. Note that

$$n_{ij} = \begin{cases} \sum_k p_{ik} n_{kj} & \text{for } i \neq j \\ \sum_k p_{ik} n_{kj} + 1 & \text{for } i = j \end{cases} \tag{19}$$

Now, define p by

$$p_i = -\frac{1}{2} q_{ii} + \sum_k n_{ik} \sum_{l=0}^n p_{kl} \left(q_{kk} - q_{kl} \right). \tag{20}$$

Then,

$$\begin{aligned} \sum_{j=0}^n p_{ij} p_j &= \sum_{j=0}^n p_{ij} \left(-\frac{1}{2} q_{jj} + \sum_k n_{jk} \sum_{l=0}^n p_{kl} \left(q_{kk} - q_{kl} \right) \right) \\ &= -\frac{1}{2} \sum_{j=0}^n p_{ij} q_{jj} + \sum_k n_{ik} \sum_{l=0}^n p_{kl} \left(q_{kk} - q_{kl} \right) - \sum_{l=0}^n p_{il} \left(q_{ii} - q_{il} \right). \end{aligned}$$

Hence,

$$p_i - \sum_{j=0}^{n} p_{ij} p_j = \frac{1}{2} q_{ii} + \sum_{j=0}^{n} p_{ij} \left(\frac{1}{2} q_{jj} - q_{ij} \right).$$

That makes the coefficient of $u_i \mu_i$ in the right side of (18) become

$$\frac{1}{2} q_{ii} + \sum_{j=0}^{n} p_{ij} (\frac{1}{2} q_{jj} - q_{ij} + p_j) - p_i = 0$$

and (18) reduces to

$$\sum_i \lambda_i \left(q_{ii} - \sum_j p_{ij} q_{ij} \right) \le \sum_i \alpha_i \left(\sum_k n_{ik} \sum_{l=0}^{n} p_{kl} \left(q_{kk} - q_{kl} \right) \right). \tag{21}$$

To simplify the left-hand side of (21), we solve (1) for λ, and use it to produce

$$\begin{aligned} \sum_i \lambda_i \left(q_{ii} - \sum_j p_{ij} q_{ij} \right) &= \sum_i \left[\left(I - P^T \right)^{-1} \alpha \right]_i \left(q_{ii} - \sum_j p_{ij} q_{ij} \right) \\ &= \sum_k \alpha_k \left(\sum_i n_{ki} \sum_{j=0}^{n} p_{ij} \left(q_{ii} - q_{ij} \right) \right). \end{aligned} \tag{22}$$

Substituting (22) into (21) shows that it is satisfied with equality for all values of Q. ■

Note that (ALP) contains $n(n+1)/2 + n + 1$ variables and $(n+1)|\mathcal{A}|$ constraints, where $\mathcal{A}$ is the set of actions, while (Drift) contains $n(n+1)/2$ variables and $n|\mathcal{A}|$ constraints. Thus, the two LPs are essentially the same size. Typically, $|\mathcal{A}|$, and the size of the LPs, is exponential in the number of classes n. For example, if each station serves one of two classes then $n = 2m$ and $|\mathcal{A}| = 3^m$. If nonidling is assumed, then each state has at most 2^m actions and some constraints can be omitted; see [8].

A. Numerical Results

To explore the relative tightness of the bounds provided by (ALP) and (Drift), we computed $J_{\text{ALP}}, J_{\text{Drift}}$, and the optimal average cost for a two-stage series queue using dynamic programming value iteration on a truncated state space. Parameters used here include service rates $\mu = (1.5, 1.25)$ and holding costs $c = (1, 2)$, while the arrival rate α_1 (and thus the maximum traffic intensity $\rho_{\max} = \max_i \{\lambda_i/\mu_i\}$) was varied. It should be noted that these parameters have not yet been scaled to satisfy $\sum_{i=1}^{n} (\alpha_i + \mu_i) = 1$. In Figure 1, the ratios J_{ALP}/J^* and J_{Drift}/J^* are plotted against $\rho_{\max}$. From the graph, it is apparent that (ALP) always gives a superior bound than (Drift), with the difference between the two being most pronounced in the cases with moderate traffic.

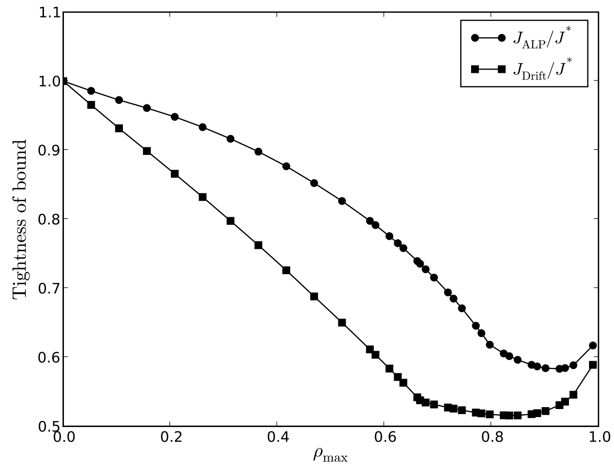

Fig. 1. Tightness of (ALP) and (Drift) bounds vs. maximum traffic intensity

We also computed the bounds for an 10-stage series queue with parameters $\mu = (1.9, 1.8, \ldots, 1)$, $c = (1, 10/9, 11/9, \ldots, 2)$, $\alpha_1 = 0.9$, and $\rho_{\max} = 0.9$. For this problem, $J_{\text{Drift}}/J_{\text{ALP}} = 0.762$; for an 18-stage series queue with similar parameters (μ ranging from 2.7 to 1 and c ranging from 1 to 2) the ratio is 0.692, showing that (ALP) can be significantly more accurate on large networks.

IV. Upper Bounds on Average Cost

Because they are interested in stability, Kumar & Meyn [5] use a performance LP that gives an upper bound on the average cost. For this upper bound to be finite, one must restrict the class of policies considered; they consider nonidling policies. Thus, the server constraint (2) is replaced by the nonidling and server constraints

$$x_j - \sum_{i:\sigma(i)=\sigma(j)} z_{ij} = 0 \quad \text{for all } j \tag{23}$$

$$\sum_{i:\sigma(i)=\sigma(j)} z_{ij} - x_j \leq 0 \quad \text{for all } j, \sigma \neq \sigma(j) \tag{24}$$

The (worst) performance LP is

$$\max_{x,\, z} \quad c^T x$$

$$\text{Subject to:} \quad (3) \text{ and } (4), (23), (24), \text{ and } z_{ij} \geq 0$$

and its dual, which [5] calls the drift LP, is

$$J_{\text{UB-Drift}} = \min_{q_{ij}} \sum_i \lambda_i \left(q_{ii} - \sum_j p_{ij} q_{ij} \right)$$

$$\text{Subject to:} \quad \sum_i \left(\alpha_i q_{ij} + \sum_{k=0}^{n} u_i \mu_i p_{ik} \left(q_{kj} - q_{ij} \right) \right) \leq -c_j \quad \text{for all } j, u \text{ such that} \sum_{i:\sigma(i)=\sigma(j)} u_i = 1.$$

Note that the constraints are indexed by actions u that are nonidling at server σ_j and that the dual variables associated with (3) and (4) are now $-q_{ij}$. In [5] the constraints are written in the nonlinear form

$$\sum_i \alpha_i q_{ij} + \max_{i:\sigma(i)=\sigma(j)} \left\{ \sum_{k=0}^{n} \mu_i p_{ik} \left(q_{kj} - q_{ij} \right) \right\} + \sum_{\sigma \neq \sigma(j)} \left[\max_{i:\sigma(i)=\sigma} \left\{ \sum_{k=0}^{n} \mu_i p_{ik} \left(q_{kj} - q_{ij} \right) \right\} \right]^+ \leq -c_j \quad \text{for all } j$$

where $[x]^+ = \max\{x, 0\}$. The corresponding upper bound ALP is then

$$\text{(UB-ALP)} \quad J_{\text{UB-ALP}} = \min_{J,\, Q,\, p} J$$

$$\text{Subject to:} \quad J \geq d^u \quad \text{for all } i, u$$

$$c_i^u \leq 0 \quad \text{for all } i, u \text{ such that } \sigma(i) \text{ is busy}$$

using (14) and (15) for d^u and c_i^u. In this case, we can similarly prove that the upper bound ALP still gives an equal or better upper bound on the performance of any non-idling policy than the upper bound Drift LP: $J_{\text{UB-ALP}} \leq J_{\text{UB-Drift}}$. The proof is essentially the same as for Theorem 1.

Acknowledgements

This work was supported in part by NSF grant CMMI-0620787. Numerical work was performed by Chris Pfohl and Jonathan Senning; Jane Eisenhauer assisted with the constraint reduction.

REFERENCES

[1] D. Bertsimas, I.Ch. Paschaladis, and J.N. Tsitsiklis. Optimization of multiclass queueing networks: Polyhedral and nonlinear characterizations of achievable performance. *Ann. Appl. Probab.*, 4:43-75, 1994.

[2] D.P. de Farias and B. Van Roy. The linear programming approach to approximate dynamic programming. *Oper. Res.*, 51(6):850-865, 2003.

[3] D.P. de Farias and B. Van Roy. A linear program for Bellman error minimization with performance guarantees. In *Advances in Neural Information Processing Systems 17*. MIT Press, 2005.

[4] S. Kumar and P.R. Kumar. Performance bounds for queueing networks and scheduling policies. *IEEE Trans. Automat. Control*, vol. AC-39:1600-1611, 1994.

[5] P.R. Kumar and S.P. Meyn. Duality and linear programs for stability and performance analysis of queueing networks and scheduling policies. *IEEE Trans. Automat. Control*, 41(4):4-17, 1996.

[6] J.R. Morrison and P.R. Kumar. New linear program performance bounds for queueing networks. *J. Optim. Theory Appl.*, 100(3):575-597, 1999.

[7] J.R. Morrison and P.R. Kumar. Computational performance bounds for Markov chains with applications. *IEEE Trans. Automat. Control*, 53(5):1306-1311, 2008.

[8] M.H. Veatch. Approximate dynamic programming for networks: Fluid models and constraint reduction. Working paper, Gordon College, Dept. of Math. Available at faculty.gordon.edu/ns/mc/Mike_Veatch, May 2005.

[9] M.H. Veatch and N. Walker. Approximate linear programming for network control: Column generation and subproblems. Working paper, Gordon College, Dept. of Math. Available at faculty.gordon.edu/ns/mc/Mike_Veatch, Feb. 2008.

APPENDIX A THE DUAL OF THE PERFORMANCE LP

Proceeding as in [5], associate the dual variables $w_{\sigma,j}$ with (2) and q_{ij} with (3) and (4). To simplify notation, we will extend Q to be symmetric and let $q_{i0} = q_{0i} = 0$. Rewriting the primal objective as $-J = \max -c^T x$, the dual objective is

$$-J = \min \sum_i \lambda_i \left(-q_{ii} + \sum_j p_{ij} q_{ij} \right)$$

and the dual can be written

$$J = \max \quad \sum_i \lambda_i \left(q_{ii} - \sum_j p_{ij} q_{ij} \right) \tag{25}$$

$$\text{Subject to:} \quad \sum_i \alpha_i q_{ij} - \sum_\sigma w_{\sigma,j} \geq -c_j \quad \text{for all } j \qquad x_j \tag{26}$$

$$\mu_i \left(\sum_k p_{ik} q_{kj} - q_{ij} \right) + w_{\sigma(i),j} \geq 0 \quad \text{for all } i, j \qquad z_{ij} \tag{27}$$

$$w_{\sigma,j} \geq 0 \quad \text{for all } \sigma, j \tag{28}$$

Only the upper triangular portion of Q is used in the dual, e.g., q_{21} should be read as q_{12}. To compare (26)–(28) to the ALP, observe that (27) gives a lower bound on $w_{\sigma,j}$ for each class served by σ. One more lower bound is (28). Since each $w_{\sigma,j}$ appears only once in (26), $w_{\sigma,j}$ can be eliminated, creating a constraint when each lower bound is substituted into (26). Equivalently, there is one constraint for each j and action u; it uses (27) for classes i with $u_i = 1$ (recall that $\sum_{i:\sigma(i)=\sigma} u_i \leq 1$) and (28) if server σ is idle. Thus, (26)–(28) are replaced by

$$\sum_i \left(\alpha_i q_{ij} + u_i \mu_i \left(\sum_k p_{ik} q_{kj} - q_{ij} \right) \right) \geq -c_j \quad \text{for all } j, u. \tag{29}$$

The form (25) and (29) is similar to the drift LP of [5]. They consider only reentrant lines and costs $c_i = 1$. They also include constraints for nonidling policies and start with a maximization problem, finding an upper bound on the average cost of any nonidling policy.

Appendix B LPs for a Series Line

We illustrate the dual relationship using a two-stage series queue with entries at class one at an arrival rate α_1, service rates μ_1 and μ_2, action $u = (u_1, u_2)$, and holding cost $c = (c_1, c_2)$. Recall from Section 2 that $q_{ij} = q_{ji}$ and $qi0 = q0i = 0$. The (best performance) ALP constraints (13) are:

$$\begin{array}{llllll}
u = (0,0) & c_1 + \alpha_1 q_{11} & & & \geq 0 & \\
& c_2 & + \alpha_1 q_{12} & & \geq 0 & \\
u = (0,1) & c_1 + \alpha_1 q_{11} & - \mu_2 q_{12} & & \geq 0 & \\
& c_2 & + \alpha_1 q_{12} & - \mu_2 q_{22} & \geq 0 & (30) \\
u = (1,0) & c_1 + (\alpha_1 - \mu_1) q_{11} & + \mu_1 q_{12} & & \geq 0 & (31) \\
& c_2 & + (\alpha_1 - \mu_1) q_{12} & - \mu_1 q_{22} & \geq 0 & \\
u = (1,1) & c_1 + (\alpha_1 - \mu_1) q_{11} & + (\mu_1 - \mu_2) q_{12} & & \geq 0 & (32) \\
& c_2 & + (\alpha_1 - \mu_1) q_{12} & + (\mu_1 - \mu_2) q_{22} & \geq 0 & (33)
\end{array}$$

For simplicity, we will convert to the upper bounds of Section IV before considering the dual. The nonidling constraints are (30)–(33). To switch to worst-performance, we need to change the direction of the inequalities. Doing this, we have:

$$\alpha_1 q_{12} - \mu_2 q_{22} \leq -c_2 \tag{34}$$

$$\left(\alpha_1 - \mu_1\right) q_{11} + \mu_1 q_{12} \leq -c_1 \tag{35}$$

$$\left(\alpha_1 - \mu_1\right) q_{11} + \left(\mu_1 - \mu_2\right) q_{12} \leq -c_1 \tag{36}$$

$$\left(\alpha_1 - \mu_1\right) q_{12} + \left(\mu_1 - \mu_2\right) q_{22} \leq -c_2 \tag{37}$$

The worst performance LP from Section IV is

$$\begin{array}{rll}
\max & c_1 x_1 + c_2 x_2 & \\
\text{Subject to:} & \alpha_1 x_1 - \mu_1 z_{11} = -\alpha_1 & -q_{11} \\
& \alpha_1 x_2 + \mu_1 z_{11} - \mu_1 z_{12} - \mu_2 z_{21} = \alpha_1 & -q_{12} \\
& \mu_1 z_{12} - \mu_2 z_{22} = -\alpha_1 & -q_{22} \\
& x_1 - z_{11} = 0 & w_{11} \\
& x_2 - z_{22} = 0 & w_{22} \\
& -x_1 + z_{21} \leq 0 & w_{21} \\
& -x_2 + z_{12} \leq 0 & w_{12} \\
& x_i, z_{ij} \geq 0 &
\end{array}$$

Taking the dual, using the dual variables listed by each constraint, gives

$$\begin{aligned}
\min \quad & -\alpha_1(-q_{11}) + \alpha_1(-q_{12}) - \alpha_1(-q_{22}) & & \\
\text{Subject to:} \quad & \alpha_1(-q_{11}) + w_{11} - w_{21} \geq c_1 & x_1 & \qquad (38)\\
& \alpha_1(-q_{12}) + w_{22} - w_{12} \geq c_2 & x_2 & \qquad (39)\\
& -\mu_1(-q_{11}) + \mu_1(-q_{12}) - w_{11} \geq 0 & z_{11} & \qquad (40)\\
& -\mu_1(-q_{12}) + \mu_1(-q_{22}) + w_{12} \geq 0 & z_{12} & \qquad (41)\\
& -\mu_2(-q_{12}) + w_{21} \geq 0 & z_{21} & \qquad (42)\\
& -\mu_2(-q_{22}) - w_{22} \geq 0 & z_{22} & \qquad (43)\\
& q_{11}, q_{12}, q_{22}, w_{11}, w_{22} \text{ u.r.s.} & & \\
& w_{12}, w_{21} \geq 0 & &
\end{aligned}$$

which, after substituting (40)–(43) and $w_{12}, w_{21} \geq 0$ into (38) and (39), simplifies to

$$\begin{aligned}
\min \quad & \alpha_1 q_{11} - \alpha_1 q_{12} + \alpha_1 q_{22} & \\
\text{Subject to:} \quad & \alpha_1 q_{12} - \mu_2 q_{22} \leq -c_2 & (44)\\
& (\alpha_1 - \mu_1) q_{11} + \mu_1 q_{12} \leq -c_1 & (45)\\
& (\alpha_1 - \mu_1) q_{11} + (\mu_1 - \mu_2) q_{12} \leq -c_1 & (46)\\
& (\alpha_1 - \mu_1) q_{12} + (\mu_1 - \mu_2) q_{22} \leq -c_2 & (47)
\end{aligned}$$

The (ALP) constraints (34)–(37) match exactly the (Drift) constraints (44)–(47).