Fluid Analysis of an Input Control Problem

Michael H. Veatch* Jonathan R. Senning†

January 17, 2006

Abstract

A two-station network with controllable inputs and sequencing control, proposed by Wein [23], is analyzed. A control is sought to minimize holding cost subject to a throughput constraint. In a Lagrangian formulation, input vanishes in the fluid limit. Several alternative fluid models, including workload formulations, are analyzed to develop a heuristic policy for the stochastic network. Both the fluid heuristic and Wein's diffusion solution are compared with the optimal policy by solving the dynamic program. Examples with up to six customer classes, using Poisson arrival and service processes, are presented. The fluid heuristic does well at sequencing control but the diffusion gives additional, and better, information on input control. The fluid analysis, in particular whether the fluid priorities are greedy, aids in determining whether the fluid heuristic contains useful information.

Running title: Fluid input control

Keywords: input control, fluid models, stochastic networks

*Department of Mathematics, Gordon College, Wenham, MA 01984, (978) 867-4375, mike.veatch@gordon.edu

†Department of Mathematics, Gordon College, Wenham, MA 01984, (978) 867-4376, jonathan.senning@gordon.edu

1 Introduction

The intractability of most stochastic network control problems has led to the development and testing of many kinds of heuristic control policies. Two of these, obtained from fluid and Brownian diffusion models, enjoy the additional justification of some form of asymptotic optimality–at least for certain problems. Many studies have demonstrated that policies based on diffusion models often outperform naive policies such as first-in-first-out (FIFO) or last-buffer-first-served (LBFS). Recent work has also shown how policies derived from fluids, sometimes augmented with buffers or corrections for stochasticity, perform well for certain problems [12], [17], [21]. However, little work has been done to directly compare diffusion policies with fluid policies. In this paper a network that lends itself to diffusion analysis–the state space collapses and performance of the diffusion policy is reasonable–is analyzed using a fluid approach and the resulting policies compared. As a benchmark, the optimal policy is also found for the examples using dynamic programming.

We consider a multiclass queueing network (MQNET) with two servers and n classes first studied by Wein [23]. In addition to the usual sequencing decisions of which class to serve, input into the network is controlled. For example, the acceptance of messages or the release of jobs into a system might be controlled. Unlike arrival control, input occurs essentially immediately. However, customers can arrive at more than one class and the mix of arrivals cannot be controlled. The objective is to minimize average holding cost subject to an average throughput constraint. Arrival and service processes are assumed to be Poisson. Wein finds the optimal policy for the associated diffusion model and gives an intuitive translation of the policy to the original network.

The fluid analysis is complicated by the fact that the average throughput constraint does not carry over to the fluid. We show that the standard fluid scaling gives a model without input. This model, and two ad hoc fluid models with input, are used to synthesize policies for the original network. For standard MQNETs, the

fluid model can be solved using Potryagin's minimization principle [3], perturbation analysis [24], [26], [22], or separated continuous linear programming [15], [27]. We solve the fluid model without input control for two examples and use a *workload relaxation* (see [13] for workload formulations and [18] for fluid workload relaxations) of the fluid model with input control. The fluid policy is translated back to the original network and the translation "tuned" by setting two or three parameters. Finally, the diffusion, tuned fluid, and optimal policies are compared numerically.

The diffusion and fluid-based policies both perform fairly well in the examples. The diffusion provides more insight into the input policy, while the fluid provides more insight into the sequencing policy. In the balanced six-class example studied by Wein, both the fluid policy and the dynamic programming solution show that switching curves are unimportant. The optimal sequencing priorities are static and agree with a greedy rule on all but one of the regions analyzed. For this example the diffusion policy does remarkable well–within 1% of optimal. In an unbalanced three-class example, the fluid policy is within 4% of optimal and outperforms the diffusion policy. One reason for the diffusion to be less accurate on this example is that one of the stations is not in heavy traffic. However, the fluid switching curves for the sequencing control are more prominant and have more cost impact. These examples suggest that the relative performance of the diffusion and fluid-based policies depends on whether the fluid control switches when some buffers are nonempty (in this case we say there are internal switching surfaces; such a policy is not greedy). One can check for these switching curves to see if the fluid policy has the potential to do well. The importance of the fluid switching curves is also supported by their relationship with the switching curves in the original model [22] and other numerical results [21].

For benchmarking, we optimize a six-class balanced network with a traffic intensity of 0.9 using dynamic programming. The dynamic program is made more efficient by finding a fairly small set of states that contain the recurrent class under the optimal control policy. Input control effectively limits the buffer sizes, making optimization of larger examples feasible.

Fluid models of queueing network control problems have received considerable attention recently for policy synthesis. Their fundamental justification is that the original network under a given policy converges to the fluid model with a similar policy when time and queue lengths are scaled by the same factor. This concept of fluid approximation holds in a very strict sense for homogeneous (Jackson) networks [9] and more loosely for MQNETs [8], where fluid limits exist but may not be unique and the relationship between the policies is less straightforward. The optimal relative value function of the original network is also related to the optimal fluid cost function; see [17] and Meyn's earlier papers referenced therein. Using a fluid solution in the original network requires, at a minimum, buffering methods to compensate for the movement of fluid through empty buffers. Other adjustments to the fluid policy may also be desirable when translating it to the stochastic network. Target-tracking translation methods are presented in [4] and [16]. Some numerical examples where a translated fluid policy performs well are given in [12] and [21]. A greedy fluid policy is tested in [2]. Alternative fluid input control problems are formulated in [20] and [22], but are not entirely satisfactory. [20] constrains the instantaneous output rate, inducing a smoothing of output over time that is not desirable in the original network. Average throughput until draining is constrained in [22]; our fluid analysis refines this approach, as discussed in Section 3.

The paper is organized as follows. The input control problem is formulated as a dynamic program in Section 2. Fluid formulations are developed and analyzed in Section 3. Sections 4 and 5 each present an example, including its fluid solution and numerical comparison of policies. The fluid solutions for the examples are derived in Appendix A and the diffusion policy from [23] is summarized in Appendix B.

2 A Two-Station Input Control Problem

As in [23], consider an open multiclass network with controllable inputs, such as Figure 3. There are n FIFO queues, or classes of customers, indexed by $i = 1, \ldots, n$, and two

stations, numbered 1 and 2. Class i is served by station $s(i)$. Let $\mathcal{P}$ be the (transient) routing matrix, so that after service completion at class i a customer is routed to class j with probability p_{ij} We assume Poisson arrivals and i.i.d. exponential service times for each class; however, more general processes also have fluid and heavy traffic limits. The uniformized discrete-time Markov chain has state $\Phi_k \in Z_+^n$, where $[\Phi_k]_i$ denotes the number of class i customers in the system after k transitions. Exogenous arrivals have a long-run average probability q_i of arriving to class i. Wein assumes that the arrival sequence cycles deterministically through the classes, but we assume independent arrivals to simplify the state space. In Wein's formulation, the time of each arrival is specified by the control. We approximate this impulse control in the MDP by a large potential arrival rate λ and a scalar input control $\widetilde{v}(k) = 1$ if input is accepted and 0 otherwise. The tilda denotes the stochastic model. Let μ_i be the service rate for class i customers and $0 \leq \widetilde{u}_i(k) \leq 1$ the fraction of time server $s(i)$ allocates to class i. The transition probabilities are

$$P(\Phi_{k+1} = x + e_i \mid \Phi_k = x) = \lambda q_i \widetilde{v}(k)/\Lambda \tag{1}$$

$$P(\Phi_{k+1} = x + e_j - e_i \mid \Phi_k = x) = \mu_i p_{ij} \widetilde{u}_i(k)/\Lambda, \tag{2}$$

where $\Lambda = \lambda + \sum_{i=1}^n \mu_i$ is the uniformization constant, e_i is the ith standard basis vector, and we use the class 0 and the convention $e_0 = 0$ to represent departures. In state $\Phi_k = x$ the controls must satisfy the constraints

$$\liminf_{T\to\infty} \frac{1}{T} E_x \sum_{k=0}^{T-1} \lambda \widetilde{v}(k) \geq \overline{\lambda} \tag{3}$$

$$\sum_{i:s(i)=j} \widetilde{u}_i(k) \leq 1 \quad \text{for } j = 1, 2 \tag{4}$$

$$\widetilde{u}_i(k) = 0 \ \text{ if } \ x_i = 0 \tag{5}$$

$$\widetilde{u}_i(k) \geq 0 \tag{6}$$

$$0 \leq \widetilde{v}(k) \leq 1, \tag{7}$$

where E_x denotes expectation given $\Phi_0 = x$ under some policy. Constraint (3) requires average throughput to be at least $\overline{\lambda}$ and (4) states that a server's allocations cannot exceed one. There is a linear holding cost $c'x$ in state x, with $c > 0$. The objective is to minimize average cost

$$\widetilde{J}(x,u) = \limsup_{T\to\infty} \frac{1}{T} E_x \sum_{k=0}^{T-1} c'\Phi_k. \tag{8}$$

Assuming a stable policy exists, an optimal policy will achieve the minimum average cost J^* independent of x. The appropriate policy will use a stationary feedback control $u = (\widetilde{v}(x), \widetilde{u}(x))$.

Instead of solving this constrained MDP, we will solve the Lagrangian problem that replaces (3) with a reward $r > 0$ per unit input accepted, minimizing

$$\widetilde{J}(x,u,r) = \limsup_{T\to\infty} \frac{1}{T} E_x \sum_{k=0}^{T-1} \left[c'\Phi_k - \lambda r\widetilde{v}(k)\right]. \tag{9}$$

As r is varied, the (deterministic) optimal policy for the Lagrangian problem will achieve some throughput $\lambda^*(r)$ and will be optimal for the constrained problem with $\overline{\lambda} = \lambda^*(r)$. The function λ^* is piece-wise constant and nondecreasing. Optimal policies for values of $\overline{\lambda}$ not in the range of λ^* can be constructed by randomizing the Lagrangian policy in a single state [1]; however, the discontinuities in λ^* are small in the examples, so randomization is not used.

Let $q = (q_1, \ldots, q_n)'$,

$$R = (I - \mathcal{P}')\text{diag}(\mu_1, \ldots, \mu_n), \tag{10}$$

and C be the $2 \times n$ constituency matrix with $c_{ji} = 1$ if $s(i) = j$ and 0 otherwise. Assuming $\mathcal{P}$ is irreducible, R^{-1} will exist and the vector of traffic intensities at each

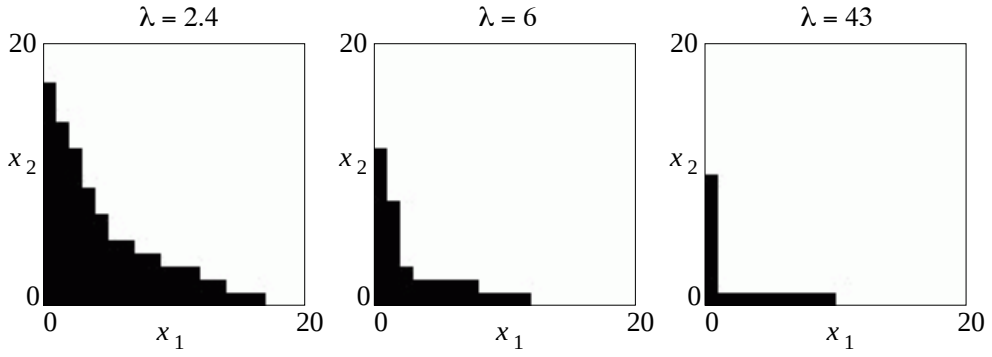

Figure 1: Optimal input control for two-class example. Input is accepted in the shaded region.

station needed to meet the throughput constraint is

$$\rho = CR^{-1}q\overline{\lambda}. \tag{11}$$

Define the system load $\rho_\bullet = \max\{\rho_1, \rho_2\}$. A stabilizing policy that achieves the throughput $\overline{\lambda}$ will exist if the usual load condition $\rho_\bullet < 1$ is met.

In Wein's formulation, input is only accepted when at least one entering class (a class i with $q_i > 0$) is empty; this is also true in (3)-(8) when λ is large. To investigate how large a value is needed, we solved a symmetric two-class example with $\mu = (1, 1)$, $c = (1, 1)$, $q = (0.5, 0.5)$, and $\overline{\lambda} = 1.8$. There is no routing; the two queues are parallel. Figure 1 shows that the optimal policy takes the correct form (accept input only when $x_1 x_2 = 0$) for $\lambda = 43$, which is about an order of magnitude greater than the combined service rate.

3 Fluid and Workload Formulations

A fluid model is obtained when all transitions are replaced by their mean rates and a continuous state is used; $x_i(t)$ is the length of the class i queue at time t, with

$x(t) \in R_+^n$. We present three fluid models, differing in their treatment of the average throughput constraint (3).

The first model is obtained by taking the fluid limit of the Lagrangian problem (4) - (7) and (9). The reward for input vanishes in the limit, leaving a fluid model with "zero input" as in [20]:

$$\begin{aligned} \text{(ZI)} \quad \min & \int_0^\infty c'x(t)dt \\ & \dot{x}(t) = -Ru(t) \\ & Cu(t) \leq 1 \\ & x(0) = x \\ & x(t) \geq 0, \quad u(t) \geq 0. \end{aligned}$$

Here $\dot{x}$ can be interpreted as a derivative to the right at nondifferentiable points. Problem (ZI) is deterministic and transient; for each initial state we can choose a time horizon T such that $x(t) = 0$ for all $t \geq T$ for a suitable class of policies. The sequel assumes T is set large enough so that this is the case.

Intuitively, r vanishes because it is not optimal to accept input when queue lengths are large. More formally, consider a sequence of processes

$$\Phi^n(t) = \frac{\Phi(nt)}{n}$$

and initial states $\Phi(0) = nx$ indexed by n and extend Φ, etc. to R_+ as piece-wise constant functions. Let

$$\widetilde{J}(x, u, r; T) = E_x \int_0^T \left[c'\Phi(t) - \lambda r\widetilde{v}(t)\right] dt$$

be the expected cost in $(0, T]$. Then the expected cost for the scaled process is

$$\begin{aligned} J^n(x,u,r;T) &= \frac{1}{n^2}\widetilde{J}(nx,u,r;nT) \\ &= E_x \int_0^T \left[c'\Phi^n(t) - \frac{1}{n}\lambda r \widetilde{v}(nt) \right] dt. \end{aligned}$$

The last term vanishes as $n \to \infty$, leaving

$$\limsup_{n\to\infty} J^n(x,u,r;T) = \limsup_{n\to\infty} E_x \int_0^T c'\Phi^n(t)dt,$$

i.e., the fluid limit is the same as for the sequencing control problem with no input. For sequencing control problems, it is shown in [17] that this limiting cost under the optimal policy u^* is equal to the optimal cost in (ZI):

$$\limsup_{n\to\infty} J^n(x,u^*,r;T) = \int_0^T c'x^*(t)dt, \tag{12}$$

where $x^*(t)$ is optimal for (ZI). Because of (12) we call (ZI) the fluid model associated with the original network. It will be used to construct a sequencing policy.

The second fluid model addresses input control by scaling the reward for input so that it does not vanish in the fluid limit. To make the total cost bounded, the minimum cost rate for any control such that $\dot{x} = 0$ must be zero. Define $v^* = \min\{\overline{\lambda}/(\lambda\rho_\bullet), 1\}$; λv^* is the largest stable input rate. Call the system *subcritical* if $\overline{\lambda}/(\lambda\rho_\bullet) > 1$ and *critical* otherwise; i.e., a system is subcritical if it can drain while accepting input at the maximum rate. The "Lagrangian" fluid model is

$$\begin{aligned} \text{(FLGR)} \quad V(x) = \min \int_0^\infty & [c'x(t) + \lambda r(v^* - v(t)]dt \\ & \dot{x}(t) = -Ru(t) + q\lambda v(t) \\ & Cu(t) \le 1 \\ & x(0) = x \\ & x(t) \ge 0, \;\; u(t) \ge 0, \;\; 0 \le v(t) \le 1. \end{aligned}$$

For a suitable class of policies and a fixed x, the system drains by some T, then $x(t) = 0$ and $v(t) = v^*$ for all $t \geq T$, so that total cost is finite. The parameters λ and r must be chosen. For the numerical examples, we set $\lambda = \overline{\lambda}$. This value just allows the throughput $\overline{\lambda}$ to be achieved after the system drains and makes any system with $\rho_\bullet < 1$ subcritical.

Like (ZI), optimal trajectories for (FLGR) can be computed using separated continuous linear programming. In particular, [15] allows both the state and the control in the objective function. To apply [27], one can fix T and convert state costs to control costs. However, analytic solutions are more difficult than for (ZI). We present results for simple systems and then consider approximations.

Proposition 1 *Consider (FLGR) for a single class and station. The optimal policy is nonidling, has feedback form, $v(x) = v(x(t))$, and uses a threshold*

$$v(x) = \begin{cases} v^*, & x \leq z \\ 0, & \text{otherwise.} \end{cases}$$

In the subcritical case $v^* = 1$ and $z = (\mu - \lambda)r/c$. In the critical case, $z = 0$.

Proposition 2 *Consider (FLGR) for the two-class parallel queue. The optimal policy is nonidling and has feedback form. In the subcritical case, $v(x) = 1$ if*

$$\frac{q_1 c_1}{\mu_1 - \lambda q_1} x_1 + \frac{q_2 c_2}{\mu_2 - \lambda q_2} x_2 \leq r$$

and $v(x) = 0$ otherwise. In the critical case, $v(0) = v^* = \max\{\mu_i/(\lambda q_i)\}$ and $v(x) = 0$ for $x \neq 0$.

See Appendix A.1 for proofs.

We have not solved larger examples to obtain an optimal sequencing policy, but note that when the optimal input region is absorbing, as in the examples above, (FLGR) is equivalent to a system with uncontrolled arrivals in this region. Outside of the input region, the optimal sequencing policy may be different than for (ZI)

because of the different terminal costs when hitting the input region. However, as $|x| \to \infty$, these differences become negligible, so that the "scaled" optimal (FLGR) control, $\widehat{u}(x) = \lim_{n\to\infty} u^*(nx)$, must agree with the optimal (ZI) control, except for x where the optimal (ZI) control is not unique. These similarities motivate our using (ZI) for sequencing.

Next we introduce a workload formulation to approximate (FLGR). It is similar to that occurring in diffusion models [13] where the state space collapses to a workload process; however, in the fluid model there is no time-scale separation and the workload version is a relaxation, introduced in [18]. It is shown there that fluid models (without input control) converge in a certain sense to their workload relaxations as $\rho_\bullet \to 1$. Let $M = CR^{-1}$. In light of (11), M_{ji} is the expected time station j must devote to a class i customer before it exits the network. The workload vector

$$w(t) = Mx(t) \tag{13}$$

represents the total amount of work in the network for each station. Let $W = \{Mx : x \geq 0\}$. In terms of workload, the dynamics of (FLGR) reduce to

$$\dot{w}(t) = -Cu(t) + (\lambda/\overline{\lambda})\rho v(t).$$

Define the effective cost

$$\bar{c}(w) = \min\{c'x : w = Mx, \;\; x \geq 0\}, \tag{14}$$

i.e., $\bar{c}(w)$ is the minimum cost of any state with workload w. If increasing a component of w always increases $\bar{c}(w)$, then $\bar{c}$ is said to be *monotone* and the optimal policy is nonidling. The workload model allows the station utilizations $Cu(t)$ to be any value up to one when $w_j(t) > 0$, i.e., there is no starvation. Denoting these utilizations $\zeta(t)$

and setting $\lambda = \overline{\lambda}$ as discussed above, the workload relaxation of (FLGR) is

$$\begin{aligned} \text{(WR)} \quad \widehat{V}(w) = \min & \int_0^\infty [\bar{c}(w(t)) + \lambda r(1 - v(t)]dt \\ & \dot{w}(t) = -\zeta(t) + \rho v(t) \\ & w(0) = w \\ & w(t) \in W, \;\; 0 \le \zeta(t) \le 1, \;\; 0 \le v(t) \le 1. \end{aligned}$$

Problem (WR) is a relaxation of (FLGR) because $\bar{c}(Mx) \le c'x$ and, for any (u, v) feasible at x for (FLGR), $\zeta = Cu$ and v are feasible for (WR). We note that [18] further simplifies (WR) by allowing $w_j(t)$ to increase instantaneously, an approximation that does not appear useful for input control.

For the two-class parallel queue, $\bar{c}$ is linear and $W = R_+^2$, so (WR) is equivalent to (FLGR). When $\bar{c}$ is not monotone, call the monotone region $W^+ = \{w \in W : \partial\bar{c}/\partial w_j \ge 0\}$. In W^+ the optimal control is nonidling. In $W \backslash W^+$ the optimal control may idle. Such a control is analyzed for one example in Section 5 and Appendix A.4.

Finally, we introduce a third fluid model which is less tractable but offers additional insight into the input control policy. For the fluid, the average throughput constraint is

$$\int_0^T \sum_{i=1}^n \mu_i p_{i0} u_i(t) \ge \overline{\lambda} T. \tag{15}$$

Again setting $\lambda = \overline{\lambda}$, the throughput will be $\overline{\lambda}$ after draining, so (15) enforces an average throughput until draining of $\overline{\lambda}$. However, (15) has the undesirable effect that if the initial customers have high work content, input is needed to reduce the average work content of customers. To avoid this behavior, we consider the fluid input control

problem

$$
\begin{aligned}
\text{(FIC)} \quad \min \quad & \int_0^T c'x(t)dt \\
& \dot{x}(t) = -Ru(t) + q\lambda v(t) \\
& Cu(t) \leq 1 \\
& \int_0^T \sum_{i=1}^n \frac{u_i(t)}{\rho_{s(i)}} dt \geq 2T \\
& x(0) = x \\
& x(t) \geq 0, \quad u(t) \geq 0, \quad 0 \leq v(t) \leq 1.
\end{aligned}
$$

Note that if each station j is busy for a total time $\rho_j T$, as needed to achieve the throughput $\overline{\lambda}$ in an empty system, the integral constraint will be tight. Because of the integral constraint, stationary feedback policies will not be optimal for (FIC).

We have only solved (FIC) for the two-class parallel queue. A connection with larger examples is suggested by again analyzing a workload relaxation. The dynamics are as in (WR). Using the flow balance equation

$$
\sum_{s(i)=j} \int_0^T u_i(t)dt = w_j + \rho_j \int_0^T v(t)dt
$$

at station $j = 1, 2$ to rewrite the integral constraint, the workload relaxation is

$$
\begin{aligned}
\text{(WRIC)} \quad \min & \int_0^T \bar{c}(w(t))dt \\
& \dot{w}(t) = -\zeta(t) + \rho v(t) \\
& \frac{w_1}{\rho_1} + \frac{w_2}{\rho_2} + 2\int_0^T v(t)dt \geq 2T \\
& w(0) = w \\
& w(t) \in W, \;\; 0 \leq \zeta(t) \leq 1, \;\; 0 \leq v(t) \leq 1.
\end{aligned}
$$

If $\bar{c}$ is monotone, the optimal control is nonidling and has the same structure as the two-class parallel queue (FIC), but on the state space W.

Proposition 3 *If $\bar{c}$ is monotone, the optimal policy for (WRIC) is nonidling. From initial states $w_1 \geq w_2$, $v(t) = 0$ for $t < T_0 - y$ and $v(t) = 1$ otherwise, where the draining time T_0 and y satisfy*

$$T_0 = w_1 + \rho_1 y \tag{16}$$

$$y = \left[\frac{[(2\rho_1 - 1)/\rho_1]w_1 - w_2/\rho_2}{2(1-\rho_1)} \right]^+ . \tag{17}$$

Input only occurs before draining if $y > 0$, i.e., if $\rho_1 > 1/2$ and $\frac{\rho_2}{\rho_1}(2\rho_1 - 1) \leq w_2/w_1 \leq 1$. Symmetric results hold for $w_1 < w_2$. Furthermore, input is only accepted on the boundary of W.

See Appendix A.1 for a proof. The structure of this policy is illustrated in Figure 2. For a given trajectory, input is accepted during the last y (possibly zero) time units before draining. No input is accepted before draining for trajectories starting in the shaded region; only the lower trajectory shown accepts input before draining.

If $\bar{c}$ is not monotone, the optimal policy uses idleness in parts of $W \backslash W^+$. The input policy is more complex, but still does not accept input in the interior of W^+.

4 A Six-class "Greedy" Example

The example from [23] is shown in Figure 3. The optimal fluid policy is nearly greedy in the sense described in Appendix A.2, allowing the diffusion policy in [23] to perform very well. Type A customers follow the routing $1 \to 2$ and type B are routed $3 \to 4 \to 5 \to 6$ The data are $\mu = (1/4,\ 1,\ 1/8,\ 1/6,\ 1/2,\ 1/7)$, $c = (1,\ 1,\ 1,\ 1,\ 1,\ 1)$, $\overline{\lambda} = 0.127$, and $q_1 = q_3 = 0.5$. The load is $\rho_1 = \rho_2 = 0.9$. We chose a potential input rate of $\lambda = 4$. Wein calculates the diffusion policy (see Appendix B for equations):

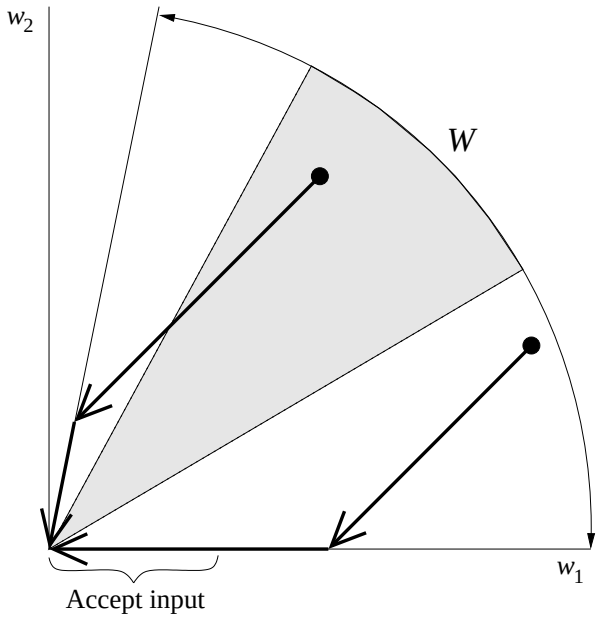

Figure 2: Optimal trajectories for (WRIC), $\bar{c}$ monotone.

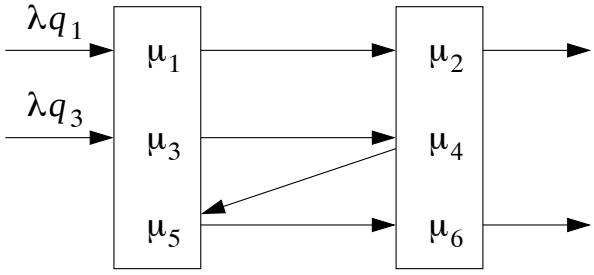

Figure 3: Wein's six-class example.

Accept input when

$$w_1 < 19 \text{ and } \; w_2 - \frac{1}{4} w_1 \leq 0 \;\; \text{or} \tag{18}$$

$$w_2 < 62 \;\; \text{and} \;\; w_1 - \frac{2}{13} w_2 \leq 0, \tag{19}$$

where the workload mapping $w = Mx$ is

$$w_1 = 4x_1 + 10x_3 + 2x_4 + 2x_5$$
$$w_2 = x_1 + x_2 + 13x_3 + 13x_4 + 7x_5 + 7x_6.$$

This policy can be understood by visualizing x_i in the workload space (Figure 4).

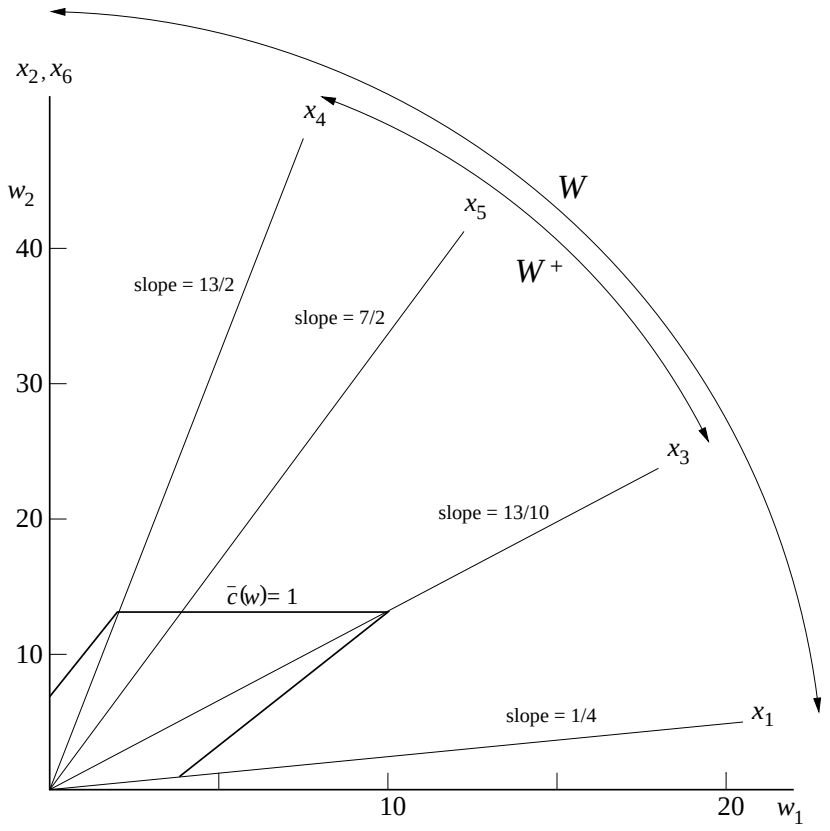

Figure 4: Mapping queues into workloads: Six-class example.

When only $x_i > 0$, the workload lies on a line through the origin with slope M_{2i}/M_{1i}; these lines are labeled x_i in Figure 4. The policy accepts input when on or above the line for class 4, or when on the line for class 1. Inequality (18) holds only on the lower boundary of the workload space, where only class 1 is held. Because of the time-scale separation that occurs in heavy traffic scaling, workload can be instantaneously shifted between classes in the diffusion model. Withholding input allows $w_1(t)$ and $w_2(t)$ to be instantaneously *simultaneously* reduced, but not rebalanced. The *workload imbalance,* $w_1(t) - w_2(t)$, is the only independent quantity that cannot be immediately eliminated. It will be held in the class with the smallest cost per unit of workload imbalance by giving this class lowest priority. When $w_1 >(<) w_2$, only class 1(4) is held. This strategy explains the slopes in (18)-(19) and gives some of the sequencing priorities. The rest of the priorities are not specified by the diffusion. Table 1 shows the priorities resulting from Wein's workload balancing rule. Two other changes were made to translate the diffusion solution to the original network. As in [23], the 0's in (18)-(19) were replaced by $\varepsilon > 0$ to allow for the discreteness of customers ($\varepsilon = 1$ in the numerical results). Secondly, we scaled the thresholds 19 and 62 until the throughput requirement was exactly met.

Now we turn to the fluid policy. For sequencing we use (ZI). The optimal policy is greedy in all but one of the regions analyzed in Appendix A.2. A greedy policy minimizes $c'\dot{x}(t)$. In our examples $c = \mathbf{1}$ and the greedy policy maximizes the current rate at which fluid leaves the system. However, the greedy criterion does not fully specify the policy. For example, when $x_2 > 0$ greedy requires $u_2 = 1$ but station 1 may idle. Table 3 in Appendix A.2 supplements greedy by using the priorities 1, 5, 3 (also, not shown in Table 3, when only type B customers are present, last buffer first served is used). The station 1 controls not specified by the greedy rule are underlined. These priorities match the priorities 2, 6, 4 of the station 2 classes that these classes feed; it is shown in Appendix A.2 that they are also optimal in most of the regions. This "supplemented greedy" sequencing policy has nearly static priorities of 1, 5, 3 and 2, 6, 4, changing only when $x_2 = x_6 = 0$:

Case 6. $x_5 > 0$: station 1 splits between 1 and 5; station 2 splits between 2 and 6.

Case 7. $x_4 > 0$ and $x_5 = 0$: station 1 splits between 1 and 5; station 2 splits between 2, 4 and 6.

In these cases effort is split between classes after a buffer empties to avoid starvation. We translate this fluid behavior to the original network by switching between two classes when a safety stock level is reached. Thus, we use the priorities

1, 5, 3 unless $x_2 + x_6 \leq k$, then use 5, 1, 3

2, 6, 4 unless $\{x_2 + x_6 \leq k_2$ and $x_5 > 0\}$, then use 6, 2, 4
or $\{x_2 + x_6 \leq k_2$ and $x_5 = 0\}$, then use 4, 6, 2

For input control, (WRIC) suggests that input should only be accepted outside of W^+. Figure 4 shows that W^+ is bounded by the directions associated with x_3 and x_4. Recognizing that the immediate effect of input is to prevent starvation at station 1, we focus on the upper part of $W \backslash W^+$, where typically x_1, x_3, and x_5 are small ($x_1 = x_3 = x_4 = x_5 = 0$ at $w_1 = 0$) and accept input when

$$x_1 + x_3 + x_5 \leq c \quad \text{and} \quad w_2 \leq w_2^*. \tag{20}$$

The second condition limits the total work in the system. The parameter w_2^* is set to achieve the required throughput. Although the parameters k, k_2, and c must be chosen, all represent "boundary thicknesses" and will be small integers. In our numerical tests $k = 1$, $k_2 = c = 0$ gave the lowest cost.

Average cost (8) is compared in Table 1 for the optimal, diffusion, and fluid policy. The optimal policy was computed using dynamic programming value iteration. An accurate state space truncation was found by adjusting the truncation until the recurrent states did not touch the upper bound for any class. We found that the

recurrent states satisfy $x \leq (2, 42, 1, 6, 6, 9)$. The algorithm converged to four significant figures within 4000 iterations (less if "warmed up" from a similar run) and ran in about 10 minutes on a 1.4 gigahertz Linux workstation. Cost of the other policies was computed using value iteration without minimizing, again checking the truncation, so that there is no simulation error. The throughput of $\bar{\lambda} = 0.127$ was achieved with parameter values of $w_2^* = 70$ for the diffusion and 62.3 for the fluid. The diffusion parameter of 70 is close to the predicted value of 62 in (19), suggesting that the diffusion gives fairly accurate performance evaluation. Examination of the the diffusion input policy shows that it agrees with the optimal input policy in most states. The small set of recurrent states may seem surprising. In this example the immediate effect of accepting input is to avoid starvation at station 1 and it is used primarily when $x_1 = x_3 = 0$.

Table 1. Priorities and average cost for six-class example

Policy	Optimal	Diffusion	Fluid
Average cost (% suboptimal)	4.03	(1%)	(24%)
Station 1 priorities	dynamic	5, 3, 1	1, 5, 3*
Station 2 priorities	2, 6, 4	2, 6, 4	2, 6, 4*

*Fluid is nearly static priority; the exceptions are listed above.

The fluid heuristic does not perform as well, despite the three parameters used. However, the sequencing priorities of the optimal policy appear similar to the fluid. For example, station 1 switches from class 1 to class 5 when x_6 approaches 0. We conclude that the diffusion policy is effective because there is balanced heavy traffic and the optimal sequencing priorities are nearly static.

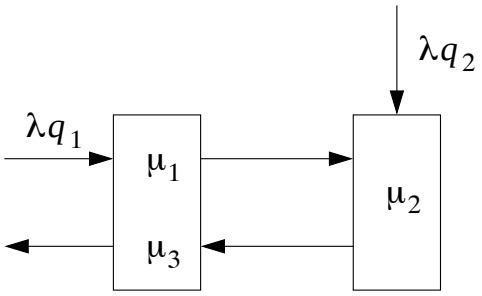

Figure 5: A three-class network.

5 A Three-class Example with Fluid Thresholds

An example in which sequencing control thresholds, or switching curves, will play a more important role is shown in Figure 5. The data are $\mu = (1/4,\ 1/9,\ 1/3)$, $c = (1,\ 1,\ 1)$, $\overline{\lambda} = 0.105$, $q = (0.8, 0.2, 0)$, and $\lambda = 10$. The load is $\rho_1 = q_1\overline{\lambda}/\mu_1 + \overline{\lambda}/\mu_3 = 0.651$ and $\rho_2 = \overline{\lambda}/\mu_2 = 0.945$. We computed the diffusion policy (see Appendix B for equations). Work is held in class 3(2) when $w_1 >(<)\ w_2$. Input is accepted when

$$w_1 < 38 \text{ and } \ w_2 \leq 0 \ \text{ or} \tag{21}$$

$$w_2 < 51 \ \text{ and } \ w_1 - \frac{1}{3}w_2 \leq 0, \tag{22}$$

where

$$w_1 = 7x_1 + 3x_2 + 3x_3$$
$$w_2 = 9x_1 + 9x_2.$$

Wein's workload balancing rule gives priority to class 1 at station 1. As in the first example, the 0's are replaced by $\varepsilon > 0$ and the thresholds of 38 and 51 are scaled to meet the throughput requirement.

The fluid sequencing control for (ZI) uses the switching curve $x_3 = 2/7x_1$ when $x_2 = 0$ (see Appendix A.3). Input control and variability both have the general effect

of shifting the switching curve in the original network. Thus, we add shifts to (33): Give priority to class 3 if

$$x_2 > b \quad \text{or } x_3 > \frac{2}{7}x_1 + c. \tag{23}$$

Both adjustments can be thought of as buffering against variability. The buffering $x_2 > 1$ is also found to be effective for a re-entrant line variant of this problem with similar parameter values in [12]. We tuned the policy by searching for the best parameter values, resulting in $b = 1$ and $c = 8$. For input control, we use the upper boundary in Figure 7, which is optimal for (WR), and extend it to larger w_1, where the optimal control for (WR) is not known. Thus, (38) or (40) must hold to accept input. Recognizing that in the original network starvation can be avoided by immediately accepting input, we also require the starvation condition

$$(x_1 + x_3)x_2 = 0 \tag{24}$$

to accept input. Setting the parameter $r = 491$ achieves the required throughput and corresponds to the bound $w_2 \leq 27$ in (38), while in the diffusion policy the bound (22) was scaled to $w_2 \leq 36$.

Table 2. Priorities and average cost for three-class example

Policy	Optimal	Diffusion	Fluid
Average cost (% suboptimal)	4.15	(15%)	(4%)
Station 1 priorities	dynamic	1, 3	dynamic

Table 2 shows that the diffusion policy is outperformed by the fluid heuristic. The fluid and diffusion costs were interpolated to the throughput of 0.1058 achieved by the optimal policy Performance of the fluid heuristic is sensitive to the switching curve (23). The optimal switching curve for station 1, shown in Figure 6, exhibits a slope close to the 2/7 predicted by the fluid but with a large offset from the x_3 boundary. In contrast, the diffusion policy uses static priorities, so that the control only switches on the boundary. Presumably the imbalanced load also contributes to

the poor performance of the diffusion policy. Figure 6 also shows the optimal input control. It is difficult to assess similarity with (38) and (40), but there is a clear similarity with (24).

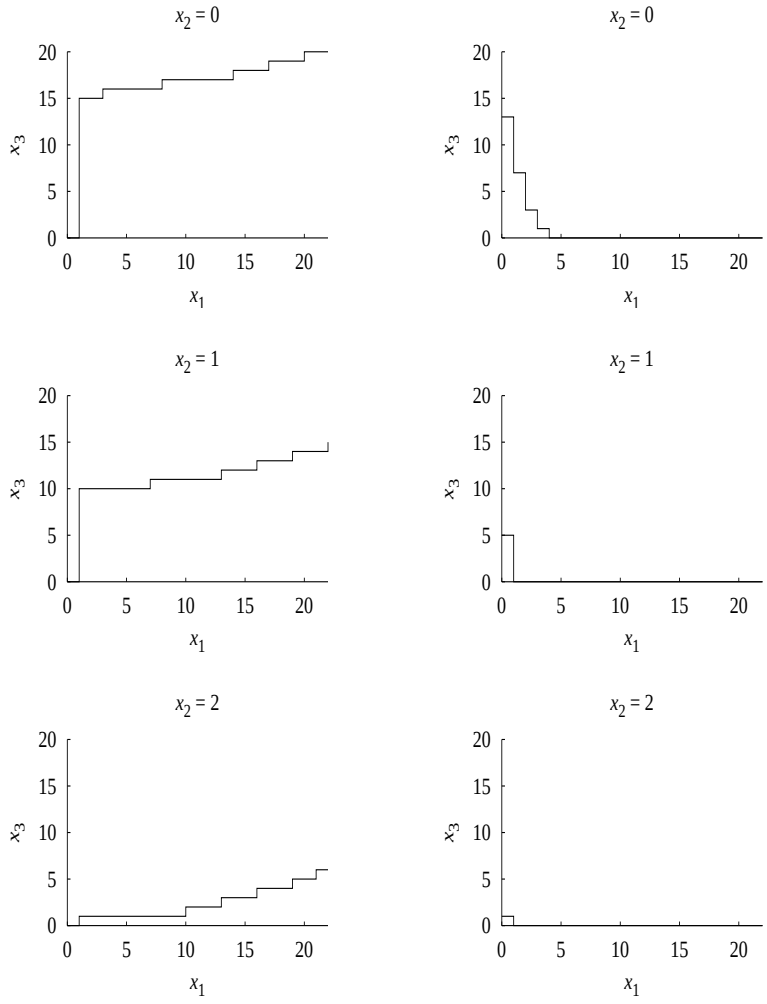

Figure 6: Optimal policy for three-class example. Class 1 is served below the lines in (i). Input is accepted below the lines in (ii).

6 Conclusion

We have demonstrated that fluid analysis of a two-station network with controllable inputs and sequencing control can provide useful information for constructing heuristic policies. Input vanishes in the fluid limit of the Lagrangian formulation, leaving a fluid control problem with no input (ZI) that is used to find a sequencing policy. Ad hoc fluid models were used to study the input policy, with the following results.

- In a two-class model with reward for accepting input (FLGR) and no sequencing control, input is accepted below a linear switching curve. For larger networks, a two-dimensional workload relaxation (WR) can be analyzed to find more complex input regions.
- In a workload model with a constraint on average station utilization (WRIC), input is accepted when the ratio of initial workloads at the two stations is sufficiently far from one.
- The workload models give some sequencing information via the concept of a minimum cost state for a given workload.

Our numerical results suggest that the diffusion policy performs well if the fluid model is greedy, but not as well if the fluid control switches when some buffers are nonempty (in this case we say there are interior switching curves; such a policy is not greedy). For example, the fluid version of Wein's six-class example is nearly greedy and the diffusion policy is within 1% of optimal. Developing a good heuristic policy requires considering each example individually; however, an important criterion for classifying them is to check whether the fluid sequencing policy is greedy. If it is greedy, there are no internal switching curves and there is little opportunity for the fluid policy to improve on the diffusion policy. If there are internal switching curves, particularly in important parts of the state space, the fluid sequencing policy may be valuable. For input control, the fluid-based policy performed reasonably well in the two examples but was somewhat ad hoc, making it difficult to generalize.

Appendix A Fluid Analysis for the Examples

A.1 Proof of Propositions 1-3

To state Bellman's equation for (FLGR), let

$$V(x,t) = \int_t^\infty [c'x(s) + \lambda r(v^* - v(s))]ds$$

under the optimal policy. Recalling that the system drains by T and $v(t) = v^*$ for all $t \geq T$, we have that $\partial V/\partial t = 0$ and $V(x,t) = V(x)$. The optimal policy and value function V satisfy Bellman's equation

$$\min_{u,v} \{c'x + \lambda r(v^* - v) + \nabla V(x)(-Ru + q\lambda v)\} = 0. \tag{25}$$

Proof of Proposition 1: For a single class and station, dropping the subscripts, (25) is

$$\min_{u,v} \{cx + \lambda r(v^* - v) + V'(x)(-\mu u + \lambda v)\} = 0, \tag{26}$$

which shows that input is accepted if $V'(x) \leq r$. If $\lambda < \mu$ the system is subcritical and $v^* = 1$. Assuming a threshold policy, say at $x = z$,

$$V(x) = \tfrac{1}{2}c\frac{x^2}{\mu - \lambda}, \quad x \leq z.$$

Setting $V'(z) = r$ (indifference in Bellman's equation) gives

$$z = \frac{r}{c}(\mu - \lambda).$$

For this policy we also have

$$V(x) = \tfrac{1}{2}c\frac{x^2 - z^2}{\mu} + \frac{\lambda r(x - z)}{\mu} + \tfrac{1}{2}c\frac{z^2}{\mu - \lambda}, \quad x > z.$$

It is easily verified that $v(x) = 0$, $x > z$, $v(x) = 1$, $x \leq z$, u nonidling, and V satisfy (26).

If $\lambda \geq \mu$ the system is critical, $v^* = \mu/\lambda$, and

$$V(x) = \tfrac{1}{2} c \frac{x^2}{\mu} + rx$$

satisfies (26) with $v(x) = 0$, $x > 0$, $v(0) = v^*$ and u nonidling. In other words, the input threshold is 0.

Proof of Proposition 2: For the two-class parallel queue system, (25) is

$$\min_{u,v} \left\{ c'x + \lambda r(v^* - v) + \sum_{i=1}^{2} \frac{\partial V}{\partial x_i}(-\mu_i u_i + q_i \lambda v) \right\} = 0 \tag{27}$$

which shows that input is accepted if

$$\sum_{i=1}^{2} q_i \frac{\partial V}{\partial x_i} \leq r. \tag{28}$$

In the subcritical case $\lambda q_i < \mu_i$ for $i = 1, 2$. If input is accepted at all t from some initial x and u is nonidling, then

$$V(x) = \tfrac{1}{2} \sum_{i=1}^{2} \frac{c_i x_i^2}{\mu_i - \lambda q_i}. \tag{29}$$

Using (29) in (28) gives

$$\frac{q_1 c_1 x_1}{\mu_1 - \lambda q_1} + \frac{q_2 c_2 x_2}{\mu_2 - \lambda q_2} \leq r. \tag{30}$$

Any trajectory that starts in the region defined by (30) remains there, verifying (29). Also, (27) holds in this region with $v = 1$ and u nonidling. It can be shown that V is convex, implying that outside of this region (28) is false and (27) holds with $v = 0$ and u nonidling. Hence, the optimal policy accepts input in region (30) and is nonidling.

In the critical case, consider the policy $v(x) = 0$, $x \neq 0$, $v(0) = v^*$, and u nonidling. For $x_1/\mu_1 \geq x_2/\mu_2$, let

$$V(x) = \tfrac{1}{2}\sum_{i=1}^{2} \frac{c_i x_i^2}{\mu_i} + \frac{\lambda r v^* x_1}{\mu_1}$$

and symmetrically for $x_1/\mu_1 < x_2/\mu_2$. This policy and V satisfy (27).

A similar result is that if station 1 is critically loaded but station 2 is not, i.e., $\overline{\lambda}/(\lambda\rho_1) \leq 1 < \overline{\lambda}/(\lambda\rho_2)$, the optimal policy accepts input when

$$x_1 = 0 \quad \text{and} \quad \frac{q_2 c_2}{\mu_2 - \lambda q_2} x_2 \leq r.$$

Proof of Proposition 3: Consider (WRIC) with $\bar{c}$ monotone. A standard interchange argument shows that the optimal policy is nonidling. Then the draining time satisfies

$$T_0 = \max\{w_1 + \rho_1 y, w_2 + \rho_2 y\}$$
$$y = \int_0^{T_0} v(t)dt.$$

After draining, $v(t) = 1$ is optimal. Hence, the integral constraint holds with T_0 in place of T, namely

$$\sum_{j=1}^{2} \frac{w_j + \rho_j y}{\rho_j} \geq 2\max\{w_1 + \rho_1 y, w_2 + \rho_2 y\}. \tag{31}$$

The objective function is increasing in y, as is the surplus in (31), implying that the optimal y is either zero or makes (31) tight.

From an initial state with $w_1 \geq w_2$, the optimal y is small enough that the maximum in (31) is achieved by the first term, i.e., there is no idleness at station 1 before T_0 and (16) holds. Under these conditions, solving (31) for y gives (17). We see from (17) that $y = 0$ if $w_2/w_1 \geq (\rho_2/\rho_1)(2\rho_1 - 1)$, which always holds if $\rho_1 \leq 1/2$.

The same argument applies if $w_1 < w_2$.

Because $\bar{c}(w(t))$ is decreasing, it is optimal to time the input as late as possible, namely, $v(t) = 0$, $t < T_0 - y$ and $v(t) = 1$ otherwise. To show that input is only accepted on the boundary of W, write the integral constraint as

$$\sum_{j=1}^{2} \int_0^{T_0} \frac{\zeta_j(t)}{\rho_j} dt \geq 2T_0.$$

Observe that $\zeta_j(t) = 1$ when $w(t)$ is in the interior of W and $\zeta_j(t) \geq \rho_j$ when $v(t) = 1$. Setting y equal to the time spent on the boundary of W before T_0, so that input is accepted exactly when on the boundary, the integrand is at least one and the constraint holds. Hence, the optimal y is no larger than this value.

A.2 Sequencing Control for the Six-Class Example (ZI)

We will construct an optimal policy for (ZI) that is greedy in the six-class example of Section 4. First we make some general observations about greedy policies. A *greedy* control in state x is one that minimizes $c'\dot{x}$. It depends on x only through the set of nonempty buffers $N = \{i : x_i > 0\}$. Using $\dot{x} = -Ru$, for each N a greedy control is a solution of

$$\begin{aligned} \text{(LP)} \quad \max \lambda(u) = \ & \mathbf{c}'Ru \\ \text{s.t.} \quad & Cu(t) \leq \mathbf{1} \\ & -[Ru]_i \geq 0, \quad i \notin N \\ & u \geq 0. \end{aligned}$$

For networks with $c = \mathbf{1}$ and deterministic routing, the following reformulation is helpful. Let $M^{(N)}$ denote the columns of M in the set N and similarly for $x^{(N)}$.

$$\begin{aligned} \text{(LPX)} \quad \max \; & -\mathbf{1}'\dot{x}^{(N)} \\ \text{s.t.} \; & -M^{(N)}\dot{x}^{(N)} \leq \mathbf{1} \\ & -\dot{x}^{(N)} \geq 0. \end{aligned}$$

Proposition 4 *Consider (ZI) with $c = \mathbf{1}$ and deterministic routing. For each N, (LPX) has an optimal solution. If $\dot{x}^{(N)}$ is optimal for (LPX) then, setting the other components of $\dot{x}$ to zero, $u = -R^{-1}\dot{x}$ is a greedy control.*

Proof: Letting $c = \mathbf{1}$ and $\dot{x} = -Ru$, $Cu = CR^{-1}Ru = -M\dot{x}$ and (LP) becomes

$$\begin{aligned} \max \; & -\mathbf{1}'\dot{x} \\ \text{s.t.} \; & -M\dot{x} \leq \mathbf{1} \\ & \dot{x}_i \geq 0, \quad i \notin N \\ & -R^{-1}\dot{x} \geq 0. \end{aligned}$$

Any feasible solution with some $\dot{x}_i > 0$ can be converted to a solution $\dot{x} \leq 0$ that has the same objective function value as follows. Reduce service to the predecessor of each class with $\dot{x}_i > 0$; if a predecessor class now has $\dot{x}_i > 0$, repeat the process until $\dot{x} \leq 0$. Deterministic routing implies $R^{-1} \geq 0$. Adding the constraint $\dot{x} \leq 0$ gives (LPX). Since $M \geq 0$ it is clear that (LPX) has an optimal solution. □

Returning to the six-class example, the greedy policy, supplemented by the (non-idling) priorities 1, 5, 3 at station 1, is shown in Table 3; "·" denotes an x_i that can be zero or positive. Controls that are not uniquely determined by the greedy criterion are underlined. The control depends only on which x_i are zero. We use this supplemented greedy policy in Section 4 to construct a heuristic.

To show the similarity with the optimal policy, we further divide the cases in

Table 3. When only type A customers (classes 1 and 2) are present, there is no sequencing control and the nonidling policy in Table 3 is optimal.

When $x_1 = 0$ (cases 2, 3, and 4 plus the case where only type B are present, not shown in Table 4) station 2 can be treated in isolation. To see this, assume that station 1 uses priorities 5, 3. Then because $1/\mu_5 < 1/\mu_6$ and $1/\mu_3 + 1/\mu_5 < 1/\mu_4 + 1/\mu_6$, station 2 is never starved. The optimal fluid solution for a single station with feedback (known as Klimov's problem) is greedy; see [10]. Hence, the greedy solution for station 2 in isolation–namely, priorities 2, 6, 4, combined with priorities 5, 3 at station 1–is optimal. Note that when only type B are present, this policy is last buffer first served (LBFS).

Table 3. Greedy policy with 1, 5, 3 priorities for (ZI), six-class example

Case	x	u	Departure rate
1	$(+,+,\cdot,\cdot,\cdot,\cdot)$	$(\underline{1},1,0,0,0,0)$	1
2	$(0,+,\cdot,\cdot,+,\cdot)$	$(0,1,0,0,\underline{1},0)$	1
3	$(0,+,+,\cdot,0,\cdot)$	$(0,1,\underline{1},0,0,0)$	1
4	$(0,+,0,\cdot,0,\cdot)$	$(0,1,0,0,0,0)$	1
5	$(+,0,\cdot,\cdot,\cdot,+)$	$(1,1/4,0,0,0,3/4)$	5/14
6	$(+,0,\cdot,\cdot,+,0)$	$(20/26,5/26,0,0,6/26,21/26)$	4/13
7	$(+,0,\cdot,+,0,0)$	$(22/25,11/50,0,18/50,3/25,21/50)$	7/25
8	$(+,0,\cdot,0,0,0)$	$(1,1/4,0,0,0,0)$	1/4

Next consider cases 1 and 5-7 when $w_2/w_1 \geq 13/10$ (above the x_3 direction in Figure 4). Transitions can occur only to increasing case numbers, e.g., from case 7 to case 8, or to states where only type B are present. Further, cases 2-4 cannot transition to cases 5-8. Without entering case 8, station 2 does not idle, implying that $w_2(t)/w_1(t) \geq 13/10$ for all t before entering case 8. But case 8 has $w_2/w_1 < 13/10$, so it cannot be entered. The trajectories use the available departure rates in decreasing order, station 2 is never starved, and station 1 is never starved until $x_1 = 0$ (after

which station 2 can be treated in isolation). Hence, they minimize the time to drain as well, suggesting that they are (pathwise) optimal. A case-by-case proof of optimality appears possible but tedious.

In case 8 the optimal control has a switching curve:

$$u(x) = \begin{cases} u^1 = (1, 1/4, 0, 0, 0, 0), & x_3 \le 3x_1 \\ u^{1,3} = (2/7, 1/14, 4/7, 6/14, 1/7, 7/14), & \text{otherwise.} \end{cases} \tag{32}$$

Control u^1 is greedy and drains from buffer 1 while $u^{1,3}$ drains buffers 1 and 3 without idleness. Since $\dot{x}(u^1) = -\frac{1}{4}e_1$, any trajectory in case 8 with $x_3 > 0$ will eventually use $u^{1,3}$, then empty buffer 1 before buffer 3. To derive (32) we first establish that one of these two controls must be optimal.

Consider a trajectory on which x_4 or x_6 is positive at some time. Modify the trajectory to keep $x_4(t) = x_6(t) = 0$ by delaying class 3 or 5 effort and serving class 1 earlier. Station 2 need not be starved when type B are present, so the modified trajectory is feasible with the same $u_6(t)$ and the same cost. Similarly, on a trajectory with x_5 positive at some time, class 3 and 4 effort can be delayed to keep $x_4(t) = x_5(t) = 0$ with the same (or reduced) cost. Hence, only controls with $\dot{x}_4 = \dot{x}_5 = \dot{x}_6 = 0$ need be considered. Further restricting to nonidling corner point controls leaves u^1, $u^{1,3}$, and $u^3 = (3/13, 0, 8/13, 6/13, 2/13, 7/13)$; u^3 drains buffer 3 and has the smallest departure rate. If a trajectory uses u^3 then x_2 is positive at some time. A second trajectory can be constructed using only u^1, $u^{1,3}$, and draining class 3 after $x_1 = 0$ that dominates the first in the sense that cumulative departures up to each t are at least as large. We omit the details.

Now assume that u^1 or $u^{1,3}$ is used while $x_1 > 0$, then buffer 3 is drained. The velocities are $\dot{x}^1 = (-1/4, 0, 0, 0, 0, 0)$, $\dot{x}^{1,3} = (-1/14, 0, -1/16, 0, 0, 0)$, and $(0, 0, -1/13, 0, 0, 0)$, respectively. From an initial state with $x_3 \ge x_1$, assume $u^{1,3}$

is used until $x_1 = 0$. The total cost is

$$V(x) = \tfrac{13}{2}x_1^2 + x_1x_3 + \tfrac{13}{2}x_3^2.$$

The switching curve must satisfy the first order condition

$$\nabla V \cdot (\dot{x}^1 - \dot{x}^{1,3}) = (13x_1 + x_3, 0, x_1 + 13x_3, 0, 0, 0) \cdot (-5/28, 0, 1/14, 0, 0, 0) = 0,$$

which yields $x_3 = 3x_1$. Convexity of total cost under an optimal policy implies that this is the unique switching curve and (32) is optimal.

Finally, consider cases 1 and 5-7 when $w_2/w_1 < 13/10$. Some of these trajectories will enter case 8 with $x_3 > 0$; for these trajectories a greedy policy is not optimal. It is conceivable that the optimal control on these trajectories differs from Table 3 before reaching case 8, due to the terminal cost upon reaching case 8. However, after checking some of the cases, we conjecture that Table 3 is optimal for these cases. For example, in case 1 class 2 has priority over class 6 because it has a larger departure rate and, as exit classes, they do not influence starvation of other classes.

Although not every case has been verified, we suspect that the supplemented greedy policy is optimal except when $x_3 > 3x_1 > 0$ and all other $x_i = 0$ (part of case 8). We expect that the supplemented greedy policy will perform very similar to the optimal fluid policy when used to construct a heuristic for the original network.

A.3 Sequencing Control for the Three-Class Example

Consider the three-class example of Section 5. The fluid policy for (ZI) is given in [25] with a proof of optimality using duality. Adding uncontrolled arrivals at class 1, the network becomes a re-entrant line; Weiss gives the fluid policy in [24] and points out that it can be found by perturbation analysis. Our example is more general in that there are arrivals at class 2. For uncontrolled arrivals and parameters

$1/\mu_2 > 1/\mu_1 + 1/\mu_3$, the fluid policy is nonidling and has a switching curve. Station 1 gives priority to class 3 if

$$x_2 > 0 \quad \text{or} \quad \frac{x_3}{x_1} > \frac{1}{1-\lambda_1/\mu_2 - \lambda_2/\mu_2} \left(\frac{1/\mu_2 - 1/\mu_1 - 1/\mu_3}{1/\mu_1 + 1/(\mu_3 - \lambda_2/\mu_2)} \right), \tag{33}$$

otherwise it serves class 1 just enough to avoid starving station 2, i.e., $u_1 = (\mu_2 - \lambda_2)/\mu_1$, $u_2 = 1$, $u_3 = 1 - u_1$ if $x_3 > 0$ and $u_3 = \mu_2/\mu_3$ if $x_3 = 0$. To derive (33) using perturbation analysis, we first write the fluid cost (with uncontrolled arrivals) assuming the latter control is used:

$$J(x) = \frac{1}{2} \left[\frac{1}{\mu_2 - \lambda_1 - \lambda_2} x_1^2 + \frac{1}{\mu_2 \mu_3 (1/\mu_2 - 1/\mu_1 - (1-\lambda_2/\mu_2)/\mu_3} x_3^2 \right]. \tag{34}$$

If switching at state x is optimal, then an infinitesimal delay in switching must not affect cost, so

$$\frac{\partial J}{\partial x_1} \partial x_1 + \frac{\partial J}{\partial x_3} \partial x_3 = 0, \tag{35}$$

where $\partial x_1 = -(\ \mu_2 - \lambda_2)\partial t$ and $\partial x_3 = [\mu_2 + (\ \mu_2 - \lambda_2)\mu_3/\mu_1]\partial t$. The result (33) follows after some algebra. For our parameter values and uncontrolled arrivals at the rate $\overline{\lambda}$, the switching curve is $x_3 > 3.33 x_1$. However, for the fluid heuristic, we used the switching curve for (ZI), which is obtained by setting $\lambda_1 = \lambda_3 = 0$ in (33), giving $x_3 > (2/7)x_1$. A general rationale for using (ZI) is given in Section 3.

A.4 Input Control for the Three-class Example (WR)

We will use (WR) to approximate the input policy in (FLGR) for the three-class example of Section 5. Set $\lambda = \overline{\lambda} = 0.105$. The effective cost is found by solving the LP (14) in the directions corresponding to x_1, x_2, and x_3 in Figure 7; it is linear between these directions. The result is

$$\bar{c}(w) = \begin{cases} \bar{c}_2^+ w_2, & w \in W^+ \\ \bar{c}_1 w_1 + \bar{c}_2 w_2 & w \in W \backslash W^+, \end{cases}$$

where $\bar{c}_2^+ = 1/9$, $\bar{c}_1 = 1/3$, $\bar{c}_2 = -4/27$, and W^+ lies above the direction of x_1, which is $w_2 = 9/7w_1$. The other data used are $\rho = (0.651, 0.945)$. Bellman's equation is similar to (27):

$$\min_{u,v} \left\{ \bar{c}(w) + \lambda r(1-v) + \sum_{i=1}^{2} \frac{\partial \hat{V}}{\partial w_i}(-u_i + \rho_i v) \right\} = 0 \tag{36}$$

which shows that $v = 1$ if

$$\sum_{i=1}^{2} \rho_i \frac{\partial \hat{V}}{\partial w_i} \leq \lambda r. \tag{37}$$

First we find the region where $v = 1$ in W^+. From $w \in W^+$, the optimal trajectory remains in W^+ and there is no idling. If input is accepted at all t, then

$$\hat{V}(w) = \tfrac{1}{2} \frac{\bar{c}_2^+ w_2^2}{1 - \rho_2}$$

and (37) yields

$$w_2 \leq \frac{\lambda r}{\bar{c}_2^+} \left(\frac{1 - \rho_2}{\rho_2} \right) = 0.055r. \tag{38}$$

One can check that $u = 1$, $v = 1$, and $\hat{V}$ satisfy (36) in the part of W^+ given by (38). In fact, in W^+ $\bar{c}$ depends only on w_2, and $\dot{w}_2 = -1 + \rho_2 v$, $w_2 > 0$. Since this is equivalent to the one-dimensional problem in Appendix A.1, that analysis shows that accepting input in W^+ when (38) holds is optimal.

Now consider a trajectory from $w \in W \backslash W^+$. Assuming nonidling, that input is always accepted and that station 2 drains last, so that $u_2(t) = 1$ until draining, the cost is

$$\hat{V}(w) = \tfrac{1}{2}(\bar{c}_1 w_1 + \bar{c}_2 w_2 + \bar{c}_2^+ w_2^+)t^+ + \tfrac{1}{2} \frac{\bar{c}_2^+ (w_2^+)^2}{1 - \rho_2} \tag{39}$$

where

$$t^+ = \frac{9/7w_1 - w_2}{9/7(1-\rho_1) + (1-\rho_2)} = 3.266w_1 - 2.540w_2$$

is the time to enter W^+ and

$$w_2^+ = w_2(t^+) = w_2 - (1-\rho_2)t^+ = -0.1796w_1 + 1.1397w_2.$$

The two terms in (39) are the cost incurred in $W \backslash W^+$ and in W^+. Proceeding numerically,

$$\frac{\partial \hat{V}}{\partial w_1} = 1.0884w_1 - 0.8467w_2$$
$$\frac{\partial \hat{V}}{\partial w_2} = -0.8467w_1 + 2.6787w_2$$

and substitution into (37) yields

$$w_2 \leq 0.04625w_1 + 0.05302r. \tag{40}$$

Note that (40) is consistent with (38), meeting at the boundary $w_2 = 9/7w_1$ of W^+, as shown in Figure 7. However, (40) assumes nonidling, when in fact station 2 idles in part of $W \backslash W^+$ because $\bar{c}_2 < 0$. Thus, it must be intersected with a nonidling region to give a valid input region.

For simplicity, we only find the station 2 idling curve for trajectories where input is always accepted. From (36), it is optimal for station 2 to work if $\partial \hat{V}/\partial w_2 \geq 0$. Again using (39), this condition yields

$$w_2 \geq 0.316w_1. \tag{41}$$

Recall that (40) assumes nonidling, i.e., (41), and (41) assumes input, i.e., (40). In fact, (36) is satisfied by $u = 1$, $v = 1$, and $\hat{V}$ in the part of $W \backslash W^+$ satisfying (40) and (41). For these trajectories station 2 drains last, as assumed. Furthermore, we claim that $u = 1$, $v = 1$ is not optimal outside of this region because a first order condition is met on the boundary of the region and $\hat{V}$ is convex.

In summary, accepting input and not idling is optimal in the shaded region in

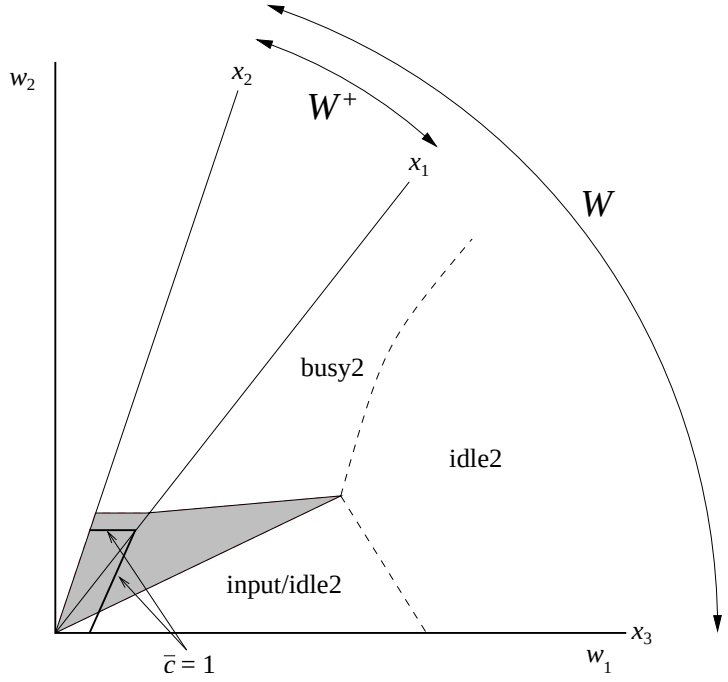

Figure 7: Workload space for three-class example. The shaded region accepts input and idles station 2. The dashed curves are hypothetical.

Figure 7, where (38) or (40) holds in addition to (41). Input is rejected just above the region and station 2 idles just below the region. Experience with similar problems suggests that other boundaries of the control regions are defined by deflective switching curves that a trajectory crosses at most once. Hypothetical deflective curves are shown. We also checked the behavior at large w, where input can be neglected, and found that for the problem without input, it is optimal for station 2 to work if $w_2 \geq 1.0037 w_1$. Thus, the asymptotic slope of the station 2 switching curve should be 1.0037.

Appendix B The Diffusion Policy

For convenience, the diffusion policy derived in [23] and [14] is summarized here. Renumber the classes so that the quantity

$$\frac{\rho_2 M_{1i} - \rho_1 M_{2i}}{c_i}$$

is maximized by class $i = 1$ and minimized by class $i = 2$. These are the classes in which the workload imbalance $\rho_2 w_1 - \rho_1 w_2$ is held when it is positive or negative, respectively. Let $\beta = R^{-1} q \overline{\lambda}$ be the traffic intensity by class, $\alpha_i = \beta_i / \rho_{s(i)}$ the fraction of busy time at station $s(i)$ spent on class i, and s_i^2 the variance of the class i service time. For exponential distributions, $s_i^2 = 1/\mu_i^2$. The uncontrolled queue length process is modeled by a Brownian motion with covariance Σ, where

$$\Sigma_{ij} = \sum_{k=1}^{n} \left[\alpha_k \mu_k p_{ki} (1_{\{i=j\}} - p_{kj}) + \alpha_k \mu_k s_k^2 R_{ik} R_{jk} \right].$$

Mapping into the workload imbalance, the variance is

$$\sigma^2 = \begin{bmatrix} \rho_2 & -\rho_1 \end{bmatrix} M \Sigma M' \begin{bmatrix} \rho_2 \\ -\rho_1 \end{bmatrix}.$$

The workload imbalance is controlled to stay within the interval $[a, b]$, where

$$\begin{aligned}
a &= \frac{1}{\nu} \ln \left(\frac{(h_1 + h_2)\rho_2(1 - \rho_1)}{h_1 \rho_2 (1 - \rho_1) + h_2 \rho_1 (1 - \rho_2)} \right) < 0 \\
b &= \frac{1}{\nu} \ln \left(\frac{(h_1 + h_2)\rho_1(1 - \rho_2)}{h_1 \rho_2 (1 - \rho_1) + h_2 \rho_1 (1 - \rho_2)} \right) > 0 \\
h_1 &= \frac{c_2}{\rho_1 M_{22} - \rho_2 M_{12}} \\
h_2 &= \frac{c_1}{\rho_2 M_{11} - \rho_1 M_{21}} \\
\nu &= \frac{2\sqrt{n}(\rho_1 - \rho_2)}{\sigma^2}.
\end{aligned}$$

Here n is a scaling parameter used by convention in the exposition of diffusion limits (a and b are actually limits of a *scaled* workload imbalance); it does not affect the results. As explained in Section 4, the controlled process lives on the x_1 and x_2 directions. Mapping the workload imbalance interval $[a, b]$ onto these directions (and unscaling) gives the control limits

$$w_1^* = \frac{\sqrt{n} M_{11} b}{\rho_2 M_{11} - \rho_1 M_{21}}$$
$$w_2^* = \frac{\sqrt{n} M_{22} a}{\rho_2 M_{12} - \rho_1 M_{22}}.$$

The policy accepts input when

$$w_1 < w_1^* \text{ and } \frac{w_2}{w_1} \leq \frac{M_{21}}{M_{11}} \text{ or}$$
$$w_2 < w_2^* \text{ and } \frac{w_2}{w_1} \geq \frac{M_{22}}{M_{12}}.$$

The constraints on the right keep the workload process on the x_1 and x_2 directions.

References

[1] E. Altman, *Constrained Markov Decision Processes*, Boca Raton, FL: Chapman and Hall/CRC Press 1999.

[2] D. Atkins and H. Chen, Performance evaluation of scheduling control of queueing networks: Fluid model heuristics, Queueing Systems 21 (1995) 391-413.

[3] F. Avram, D. Bertsimas, and M. Ricard, Fluid models of sequencing problems in open queueing networks: An optimal control approach, in: *Stochastic Networks,* vol. 71 of the Proceedings of the IMA, eds. F. Kelly and R. Williams, Springer-Verlag, New York, 1995 pp. 199-234.

[4] N. Bäuerle, Asymptotic optimality of tracking policies in stochastic networks, Ann. Appl. Prob. 10 (2000) 1065-1083.

[5] D. Bertsimas, D. Gamarnik, and J. Sethuraman, From fluid relaxations to practical algorithms for high-multiplicity job-shop scheduling: The holding cost objective, Oper. Res. 51 (2003) 798-813.

[6] T. Boudoukh, M. Penn, and G. Weiss, Scheduling jobshops with some identical or similar jobs, J. of Scheduling 4 (2001) 177-199.

[7] D. Bertsimas, I.Ch. Paschaladis, and J.N. Tsitsiklis, Optimization of multiclass queueing networks: Polyhedral and nonlinear characterizations of achievable performance, Ann. Appl. Prob. 4 (1994), no. 1, 43-75.

[8] H. Chen, Fluid approximation and stability of multiclass queueing networks: Work-conserving disciplines, Ann. Appl. Prob. 5 (1995) 637-665.

[9] H. Chen and A. Mandelbaum, Discrete flow networks: Bottleneck analysis and fluid approximations, Math. Oper. Res. 16 (1991) 408-446.

[10] H. Chen and D.D. Yao, Dynamic scheduling of multiclass fluid networks, Oper. Res. 41 (1993) 1104-1115.

[11] J.G. Dai, On the positive Harris recurrence for multiclass queueing networks: A unified approach via fluid limit models, Ann. Appl. Prob. 5 (1995) 49-77.

[12] D. Eng, J. Humphrey, and S.P. Meyn, Fluid network models: Linear programs for control and performance bounds, in: *Proceedings of the 13th IFAC World Congress*, eds. J. Cruz, J. Gertler and M. Peshkin, San Francisco, CA, 1996, vol. B, pp. 19-24.

[13] J.M. Harrison, Balanced fluid models of multiclass queueing networks: A heavy traffic conjecture, in *Stochastic Networks*, vol. 71 of the Proceedings of the IMA, eds. F. Kelly and R. Williams, Springer-Verlag, New York, 1995, pp. 1-20.

[14] J.M. Harrison and L.M. Wein, Scheduling networks of queues: Heavy traffic analysis of a two-station closed network, Oper. Res. 38 (1990) 1052-1064.

[15] X. Luo and D. Bertsimas, A new algorithm for state-constrained separated continuous linear programming, SIAM J. Control and Optim., 37 (1998) 177-210.

[16] C. Maglaras, Discrete-review policies for scheduling stochastic networks: Fluid-scale asymptotic optimality, Ann. Appl. Prob. 10 (2000) 897-929.

[17] S.P. Meyn, Sequencing and routing in multiclass queueing networks. Part I: Feedback regulation, *2000 IEEE International Symposium on Information Theory*, Sorrento, Italy, June 25 - June 30, 2000, and SIAM J. Control and Optimization 40 (2001) 741-776.

[18] S.P. Meyn, Sequencing and routing in multiclass queueing networks. Part II: Workload relaxations, SIAM J. Control and Optimization, 42 (2003) 178-217.

[19] I.Ch. Paschaladis, C. Su, and M.C. Caramanis, Target-pursuing scheduling and routing policies for multiclass queueing networks, IEEE Trans. Automat. Contr. 49 (2004) 1709-1722.

[20] M.J. Ricard, Optimization of queueing networks: An optimal control approach, Ph.D. thesis, MIT Operations Research Center, Cambridge MA, 1995.

[21] M.H. Veatch, Fluid analysis of arrival routing, IEEE Trans. Automat. Contr. 46 (2001) 1254-1257.

[22] M.H. Veatch, Using fluid solutions in dynamic scheduling, in: *Analysis and modeling of manufacturing systems*, eds. S.B. Gershwin, Y. Dallery, C. Papadopoulos and J.M. Smith, Kluwer, New York, 2002, pp. 399-426.

[23] L.M. Wein, Scheduling networks of queues: Heavy traffic analysis of a two-station network with controllable input, Oper. Res. 38 (1990)1065-1078.

[24] G. Weiss, On optimal draining of re-entrant fluid lines, in: *Stochastic Networks,* vol. 71 of the Proceedings of the IMA, eds. F. Kelly and R. Williams, Springer-Verlag, New York, 1995, pp. 91-103.

[25] G. Weiss, Optimal draining of fluid re-entrant lines: Some solved examples, in: *Stochastic Networks: Theory and Applications*, eds. F.P. Kelly, I. Ziedins and S. Zachary, Oxford Univ. Press, 1996.

[26] G. Weiss, Algorithm for minimum wait draining of two station fluid re-entrant line, Annals of Operations Research 92 (1999) 65-86.

[27] G. Weiss, A simplex based algorithm to solve separated continuous linear programs, to appear Math. Programming, 2002.