

# Optimal dynamic scheduling in a multiclass fluid model of Internet servers with transient overload

Junxia Chang  
Hayriye Ayhan  
Jim Dai

School of Industrial and Systems Engineering  
Georgia Institute of Technology  
Atlanta, GA 30332-0205 U.S.A  
junxiach@isye.gatech.edu  
hayhan@isye.gatech.edu  
dai@isye.gatech.edu

Zhen Liu  
Mark S. Squillante  
Cathy H. Xia

IBM T. J. Watson Research Center  
Department of Electrical Engineering  
Yorktown Heights, NY 10598, USA  
zhenl@us.ibm.com  
mss@us.ibm.com  
cathyx@us.ibm.com

**Abstract**—We consider the optimal dynamic scheduling of different requests of service in a multiclass stochastic fluid model that is motivated by recent and emerging computing paradigms for Internet services and applications. Our primary focus is on environments with specific performance guarantees for each class under a profit model in which revenues are gained when performance guarantees are satisfied and penalties are incurred otherwise. Within the context of the corresponding fluid model, we explore the dynamic scheduling of different classes of service under conditions where the workload of certain classes may be overloaded for a transient period of time. In particular, we consider the case with two fluid classes and a single server whose capacity can be shared arbitrarily among the two classes. Under the assumptions that the class 1 arrival rate varies with time and the class 1 fluid can more efficiently reduce the holding cost, we determine the optimal server allocation policy that minimizes the holding cost in the fluid model when the arrival rate function for class 1 is known. Using the key insights gained from this deterministic case, we also develop heuristic policies for the stochastic fluid system when the arrival rate function for class 1 is random.

## I. INTRODUCTION

Internet services and other recent emerging applications have created new computing and networking paradigms in which a set of e-commerce businesses contract with a common hosting provider of Internet applications and services for their respective customers. A key characteristic of such environments is the diverse requirements of the various e-commerce businesses and customers. In order to address these diverse requirements and leverage potential economies of scale, the hosting service provider will often deploy a cluster of servers to effectively share the computing and networking resources required to support the desired Internet applications and services. A number of computer industry companies are already providing such hosting services, e.g., HP, IBM and Intel, and it appears that more companies will be doing so in the future.

The diverse requirements of e-commerce businesses and customers motivate the definition of different service classes. These service classes typically have distinct levels of importance to the hosting service provider, the businesses and

their customers. Moreover, many of these service classes require specific Quality-of-Service (QoS) performance guarantees, because failures to deliver such levels of QoS can have a significant impact on the e-commerce businesses and customers. For example, customers will easily lose patience and discontinue using the service if its responsiveness is perceived to be too long. Thus, as part of the contract between the service provider and each business, the hosting service provider agrees to guarantee a certain level of QoS for each class of service, and in return each e-commerce business agrees to pay the service provider for satisfying these QoS performance guarantees. These contracts are based on a Service-Level-Agreement (SLA), between each business and the service provider, that defines the QoS performance guarantees for the classes of service, and the anticipated level of per-class workload from the customers of the business.

A critical issue for the hosting service provider concerns the control of server resource allocation to optimize performance and profit measures in cluster-based computing environments with SLA contracts containing QoS performance guarantees. This is also a fundamental issue for the continued growth and success of Internet services and applications. We therefore focus herein on a particularly important class of dynamic scheduling problems that arise in these computing environments. It is important to note, however, that while our analysis and results are motivated by such environments, they apply more generally to a wide variety of emerging computing environments with SLA-based QoS performance guarantees.

Most previous studies have considered QoS performance guarantees based on throughput or mean response time measures. However, a crucial issue for Internet applications and services concerns the per-request efficiency with which the differentiated services are handled, because delays experienced by customers can result in lost revenue and customers for a business as noted above. Furthermore, such QoS performance guarantees may not be fully captured by the more standard performance metrics such as throughput and mean response time. To address these issues, we consider

a general class of SLAs in which a threshold is defined for each class of service such that the hosting service provider gains revenues when the QoS level experienced by the class stays at or below the threshold, but the service provider pays penalties to the corresponding businesses when this threshold is exceeded. The optimal control problem is then concerned with allocating server resources in order to maximize the profit of hosting the collection of e-commerce sites under these SLA constraints.

Another important aspect of the problem concerns the diverse workloads of different e-commerce businesses and their variation over time. In particular, it is quite common in the computing environments of interest to have the workload of certain classes in each e-commerce site alternate between a period during which the arriving workload exceeds the allocated capacity, and a period during which the arriving workload is less than this capacity, even though the average load is within the allocated capacity; e.g., refer to [2]. These periods of transient overload can have a significant impact on the performance experienced by the different classes of service. This in turn can have a critical impact on the penalties that the hosting service provider must pay each e-commerce business according to the SLA contract between them. It is therefore crucial to include these important workload characteristics in the analysis of the optimal control problem.

Our problem falls within the general class of optimal resource control problems, but based on the foregoing non-conventional performance metrics and workload characteristics. Some research studies have considered the issue of workloads with transient overload, but these studies have focused on single-class workloads and specific scheduling strategies, such as admission control (e.g., [6]) and direct modifications to the Internet server scheduling mechanism (e.g., [2], [5]). In contrast, our focus in this paper is on the optimal dynamic scheduling of a multiclass system with transient overload. Furthermore, very little research has even attempted to consider the issue of maximizing profit in these computing paradigms under non-conventional performance metrics. The primary exception is the study in [10], which develops queueing-theoretic bounds and approximations to formulate the resource control optimization problem and then exploits efficient algorithms for computing the optimal solution. This study is the most closely related to our research, but it differs from the present study in several important respects. Our focus in this paper is on deriving the optimal dynamic scheduling policy and gaining insights into its fundamental properties, as opposed to computing the steady-state solution, and to do so under a workload with transient overload, which is not considered in [10].

The objective of this paper is to investigate the optimal server resource control problem described above as a dynamic scheduling problem. Our approach is based on

formulating the problem as a multiclass stochastic fluid model and exploiting optimal control theory [11], [12] to obtain the optimal control policy that maximizes the total revenue over a fixed time horizon. Recent studies of a similar spirit for different dynamic scheduling problems include [1], [3]. To the best of our knowledge, however, no optimal scheduling policy is known for the general problem considered herein. We therefore focus on minimizing the penalty of the hosting service provider by dynamically scheduling its server resources among the fluid classes in a system that can be overloaded for a transient period. To capture the QoS performance guarantees in the SLA contracts, a threshold value is introduced for each fluid class such that a holding cost is incurred only if the amount of fluid of a certain class exceeds its threshold value. We consider the specific case of two fluid classes and a single server whose capacity can be shared arbitrarily among the two classes. Under the assumptions that the class 1 arrival rate changes with time and the class 1 fluid can more efficiently reduce the holding cost, we determine the optimal server resource allocation policy that minimizes the holding cost in the corresponding fluid model when the arrival rate function for class 1 is known. Using the key insights gained from this deterministic case, we also develop heuristic policies for the stochastic fluid system when the arrival rate function for class 1 is random. Preliminary numerical examples demonstrate that these heuristic policies yield good results in terms of minimizing the expected holding cost.

The remainder of this paper is organized as follows. We define our multiclass fluid model in §II. Deterministic and stochastic instances of the model are analyzed in §III and §IV, respectively. Our concluding remarks are provided in §V.

## II. THE STOCHASTIC FLUID MODEL

We consider the following stochastic fluid system that serves two classes of fluid. For each class, fluid continuously arrives at its buffer whose capacity is assumed to be infinite. Fluid in both classes is served by a single server whose service capacity can be shared arbitrarily among the two classes. When the server devotes full effort to class  $i$ , it processes class  $i$  fluid at rate  $\mu_i$ ,  $i = 1, 2$ .

Class 2 fluid arrives at the system at a constant rate  $\lambda_2$  throughout the time horizon under consideration. Class 1 fluid has a high arrival rate  $\lambda_1^h$  during the first part of the time interval. In the rest of the time interval, it has a low arrival rate  $\lambda_1^l$ . Naturally, we assume  $\lambda_1^l \leq \lambda_1^h$ . The durations of the first and second time intervals are denoted by  $H$  and  $L$ , respectively. We assume that  $H$  is a random variable with a known distribution and  $L$  is infinitely long or long enough such that at the end of each cycle everything will be cleared and the system can start over again in the next cycle. This is a reasonable assumption in practice because if the underload

period is not sufficiently long to satisfy these conditions, then the original system approximated by the fluid model will be unstable. We call the time interval  $[0, H)$  the high load period and the time interval  $[H, H + L)$  the low load period.

We use  $Z_i(t)$  to denote the fluid level in class  $i$  at time  $t$ , and  $u_i(t)$  to denote the fraction of capacity at time  $t$  that the server spends on class  $i$  fluid,  $0 \leq u_i(t) \leq 1$ ,  $i = 1, 2$ . The dynamics of our fluid system is given by the following equation, for  $i = 1, 2$

$$Z_i(t) = Z_i(0) + \int_0^t \lambda_i(s) - \mu_i u_i(s) ds, \quad t \in [0, H + L), \quad (1)$$

where  $\lambda_i(s)$  is the arrival rate to class  $i$  at time  $s$ . Note that the class 1 arrival rate function  $\lambda_1(s) = \lambda_1^h \cdot 1(s \leq H) + \lambda_1^l \cdot 1(H < s \leq H + L)$  is random due to the random length of the high load period. Consequently, the fluid level process  $Z$  is also random. The allocation process  $U = \{(u_1(t), u_2(t)), t \geq 0\}$  reflects how the server spends its service capacity among two classes. It is called a scheduling or service policy.

Let  $h_i > 0$  and  $\theta_i \geq 0$  be constants,  $i = 1, 2$ . For a real number  $x$ ,  $x^+ = \max(x, 0)$ . Consider the following integral

$$\int_0^{H+L} \sum_{i=1}^2 h_i (Z_i(t) - \theta_i)^+ dt, \quad (2)$$

which is called the total cost of the system and is denoted by  $C$ . One interprets  $h_i$  as the holding cost per unit of time when the fluid level in class  $i$  exceeds  $\theta_i$ . One further interprets  $\theta_i$  as the threshold value we introduce for fluid class  $i$  to capture the corresponding QoS performance guarantee in the SLA contracts. When the fluid level in class  $i$  is below  $\theta_i$ , the fluid does not accumulate cost for the system. Obviously, the cost depends on initial fluid level  $z = Z(0)$ , and allocation  $U$  employed. When such an explicit dependence is needed, we use  $C_U(z)$  to denote such a cost. The focus of this paper is to find an allocation  $U$  to minimize the expected total cost  $\mathbb{E}[C_U(z)]$  for each initial point  $z$ . We assume that working on class 1 can more efficiently reduce holding costs. Namely,  $h_1 \mu_1 > h_2 \mu_2$ .

When  $\theta_i = 0$  for  $i = 1, 2$ , the optimal policy is given by the well-known  $c\mu$  rule [13], [7], [8]. Namely, the server gives priority to class  $i$  with the highest  $h_i \mu_i$ . To the best of our knowledge, there is no optimal policy known for our general problem. In the special case when  $H$  is deterministic and is known at the beginning of the time horizon, we will present an optimal policy. Using this policy, we will construct heuristic policies for the case that  $H$  is random. We will present numerical experiments showing that these heuristic policies perform well.

For future reference, we define the traffic intensities of the system. The system load per unit of time contributed by class 1 fluid is  $\rho_1^h = \lambda_1^h / \mu_1 > 0$  for the high load period and  $\rho_1^l = \lambda_1^l / \mu_1$  for the low load period. The system

load per unit of time contributed by class 2 fluid is constant given by  $\rho_2 = \lambda_2 / \mu_2 > 0$ . The overall system load is  $\rho^h = \rho_1^h + \rho_2$  for the high load period and  $\rho^l = \rho_1^l + \rho_2$  for the low load period. When  $\rho^h > 1$  and  $\rho^l < 1$ , the total system work increases in the high load period and decreases in the low load period. In this case, the high load period is also called the overload period. Thus, when  $\rho^h > 1$  and  $\rho^l < 1$  the system experiences an overload period followed by an underload period, a phenomenon known as transient overload in literature [2]. While understanding the transient overload case is a primary motivation of this paper, and thus we typically consider  $\rho^h > 1$ , it is important to note that our results generally do not require this condition, unless explicitly stated otherwise.

### III. OPTIMAL POLICIES IN THE DETERMINISTIC CASE

In this section, we present the optimal policy when the lengths of the high period and the low period are known. The proof of the optimality of this policy is given in [4]. Specifically, we assume that the duration of the overload period  $H$  is deterministic and known at the beginning of the time horizon. Thus the arrival rate function  $\lambda_1(s) = \lambda_1^h 1(s \leq H) + \lambda_1^l 1(H < s \leq H + L)$  is known. The system starts with initial fluid level  $Z(0) = (Z_1(0), Z_2(0))$ . We also assume that  $L$  is infinitely long or long enough that all the loads can be decreased to below their thresholds under non-idling policies. (In fact as long as  $H + L \geq t_2$ , where  $t_2$  is specified later in this section, we would call  $L$  is long enough.) For convenience, we first define the policy for the low period, i.e., when  $H < t \leq H + L$ .

**Definition 1.** *The following policy, referred to as the Low-period-policy, is implemented in the low period.*

- If  $Z_1(t) > \theta_1$ , full capacity is given to class 1, i.e.  $u_1(t) = 1, u_2(t) = 0$ , until class 1 fluid level is decreased to its threshold  $\theta_1$ .
- If  $Z_1(t) = \theta_1$ ,  $Z_2(t) > \theta_2$ , class 1 fluid is kept at its threshold value  $\theta_1$ , while the remaining capacity is used to serve class 2, i.e.  $u_1(t) = \rho_1^l$ ,  $u_2(t) = 1 - \rho_1^l$  until class 2 fluid level is decreased to its threshold  $\theta_2$ .
- If  $Z_1(t) \leq \theta_1$ ,  $Z_2(t) \leq \theta_2$ , then the policy is not unique and  $u_1(t)$  and  $u_2(t)$  can be chosen such that  $u_1(t) \geq \rho_1^l$ ,  $u_2(t) \geq \rho_2$  and  $u_1(t) + u_2(t) = 1$ .

The optimal policy depends on the load conditions. In the next three subsections, we will describe the optimal policy under all possible load conditions. In the first case, it is assumed that  $\rho_1^h > 1, \rho^l < 1$ ; in the second case, it is assumed that  $\rho^h > 1, \rho_1^h < 1, \rho^l < 1$ ; and in the last case, it is assumed that  $\rho^h < 1, \rho^l < 1$ .

### A. The case $\rho_1^h > 1, \rho^l < 1$

Throughout this section, we assume that  $\rho_1^h > 1, \rho^l < 1$ . Then the optimal policy has the following structure:

**(OPT)**

$$\begin{aligned} \forall t \in (0, s_1) : & \quad u_2(t) = 1, u_1(t) = 0; \\ \forall t \in (s_1, s_2) : & \quad u_2(t) = u_2, u_1(t) = u_1, u_1 + u_2 = 1; \\ \forall t \in (s_2, H) : & \quad u_2(t) = 0, u_1(t) = 1; \\ \forall t \in (H, H + L) : & \quad \text{Low-Period-Policy.} \end{aligned}$$

Thus, the optimal policy gives fixed priority to class 2 in the interval 0 to  $s_1$ , employs processor sharing in the interval  $s_1$  to  $s_2$  and gives fixed priority to class 1 in the interval  $s_2$  to  $H$ . Specific values of  $s_1, s_2, u_1$ , and  $u_2$  depend on the initial fluid levels and the length of the high period. Before discussing the computation of  $s_1, s_2, u_1$  and  $u_2$  for all possible cases, we introduce the notation used in our developments:

$$\begin{aligned} d_1 &:= \theta_1 - Z_1(0), \quad \psi_1 := \frac{d_1/\mu_1}{\rho_1^h - 1}, \quad \tilde{\psi}_1 := \frac{d_1/\mu_1}{\rho_1^h}; \\ d_2 &:= \theta_2 - Z_2(0), \quad \psi_2 := \frac{d_2/\mu_2}{\rho_2}, \quad \tilde{\psi}_2 := \frac{-d_2/\mu_2}{1 - \rho_2}. \end{aligned}$$

The quantities  $\psi_1, \psi_2, \tilde{\psi}_1$  and  $\tilde{\psi}_2$  have the following interpretations. Quantity  $\psi_1$  is the time that class 1 increases to its threshold  $\theta_1$  under the policy that gives fixed priority to class 1 if the initial fluid level of class 1 is below  $\theta_1$  and if the high period is long enough. Quantity  $\tilde{\psi}_1$  is the time class 1 increases to its threshold  $\theta_1$  under the policy that gives fixed priority to class 2 if the initial fluid level of class 1 is below  $\theta_1$  and if the high period is long enough. Quantity  $\psi_2$  is the time class 2 increases to its threshold  $\theta_2$  under the policy that gives fixed priority to class 1 if the initial fluid level of class 2 is below  $\theta_2$ . Finally,  $\tilde{\psi}_2$  is the time class 2 decreases to its threshold  $\theta_2$  under the policy that gives fixed priority to class 2 if the initial fluid level of class 2 is above  $\theta_2$ . Clearly,  $d_1$  and  $d_2$  denote the initial deviation of the fluid levels from the desired thresholds for classes 1 and 2, respectively.

We also define

$$\begin{aligned} a_1 &:= \frac{d_1/\mu_1 + d_2/\mu_2}{\rho_1^h + \rho_2 - 1}, \quad a_2 := \frac{1 - \eta\xi}{1 - \eta} \psi_1^+ - \frac{\eta(1 - \xi)}{1 - \eta} \psi_2^+, \\ B &= \frac{1 - \eta\xi}{1 - \eta} \psi_1^+ - \frac{(1 - \rho_1^l)[1 + \eta(\rho_1^h - 1)] + (1 - \eta)(\rho_1^h - 1)}{(\rho_1^h - 1)(\rho_1^h - \rho_1^l)(1 - \eta)} \tilde{\psi}_2^+, \end{aligned}$$

where  $\eta = h_2\mu_2/h_1\mu_1$  and  $\xi = (\rho_1^h - 1)/(\rho_1^h - \rho_1^l)$ . Quantities  $a_1, a_2$  and  $B$  have the following interpretations. Quantity  $a_1$  is the critical value such that if the high period is longer than  $a_1$  then under any policy either class 1 fluid level will exceed its threshold  $\theta_1$  or class 2 fluid level will exceed its threshold  $\theta_2$ . Quantity  $a_2$  is the critical value such that if the high period is longer than  $a_2$ , and the low period is long enough to reduce the fluid level of class 1 to its threshold  $\theta_1$  then fixed priority to class 1 is the optimal policy in the high period. Finally,  $B$  is the critical value such that if the high period is longer than  $B$  and the low period is long enough

to reduce the fluid level of class 1 to its threshold  $\theta_1$  then the optimal policy never uses processor sharing in the high period.

We are now ready to provide a more detailed description of the optimal policy.

- Case 1:  $Z_1(0) \geq \theta_1$ . In this case, the optimal policy is given by **(OPT)** with  $s_1 = s_2 = 0$ .
- Case 2:  $Z_1(0) < \theta_1, Z_2(0) > \theta_2$ . Computation of  $s_1, s_2, u_1$  and  $u_2$  depends on the length of the high period.
  - Case 2.1: If

$$a_1 < H \leq B, \quad (3)$$

then  $s_1, s_2, u_1$  and  $u_2$  are computed by solving

$$Z_2(0) + (\lambda_2 - \mu_2)s_1 = \theta_2; \quad (4)$$

$$Z_1(0) + \lambda_1^h s_1 = Z_1(s_1); \quad (5)$$

$$Z_2(s_1) + (\lambda_2 - \mu_2 u_2)(s_2 - s_1) = \theta_2; \quad (6)$$

$$Z_1(s_1) + (\lambda_1^h - \mu_1 u_1)(s_2 - s_1) = Z_1(s_2); \quad (7)$$

$$u_1 + u_2 = 1; \quad (8)$$

$$Z_1(s_2) + (\lambda_1^h - \mu_1)(t_1 - s_2) = \theta_1; \quad (9)$$

$$Z_1(t_1) + (\lambda_1^h - \mu_1)(H - t_1) = Z_1(H); \quad (10)$$

$$Z_1(H) + (\lambda_1^l - \mu_1)(t_2 - H) = \theta_1; \quad (11)$$

$$\mu_1 h_1(t_2 - t_1) = \mu_2 h_2(t_2 - s_2). \quad (12)$$

Note that equations (4) to (11) describe the evolution of the fluid levels of class 1 and class 2 from time 0 to  $t_2$  under the optimal policy, where  $t_2$  represents the time epoch at which the class 1 fluid level in the low period reaches its threshold value as indicated in equation (11).

- Case 2.2: If

$$\max\{B, \tilde{\psi}_1\} < H \leq a_2,$$

then we set  $s_1 = s_2$  and solve the equations (5) and (9)–(12) for  $s_2, t_1$  and  $t_2$ .

- Case 2.3: If

$$H \leq \max\{a_1, \tilde{\psi}_1\},$$

then the optimal policy is given by **(OPT)** with  $s_1 = \min\{\tilde{\psi}_2, H\}, s_2 = H, u_2 = \rho_2$ , and  $u_1 = 1 - \rho_2$ .

- Case 2.4: If

$$H > a_2,$$

then the optimal policy is given by **(OPT)** with  $s_1 = s_2 = 0$ .

**Remark 2.** It can be readily verified that  $\tilde{\psi}_1 \geq \tilde{\psi}_2$  implies  $B \geq a_1 \geq \tilde{\psi}_1$ , and that  $\tilde{\psi}_1 \leq \tilde{\psi}_2$  implies  $B \leq a_1 \leq \tilde{\psi}_1$ .

- Case 3:  $Z_1(0) < \theta_1, Z_2(0) \leq \theta_2, \psi_1 \leq \psi_2$ . In this case, then the optimal policy is given by **(OPT)** with  $s_1 = s_2 = 0$ .

- Case 4:  $Z_1(0) < \theta_1$ ,  $Z_2(0) \leq \theta_2$ ,  $\psi_1 \geq \psi_2$ . In this case,  $s_1 = 0$ . However, the computation of  $s_2$ ,  $u_1$  and  $u_2$  depends on the length of the high and the low periods as discussed below.

- Case 4.1: If

$$a_1 \leq H \leq a_2,$$

then  $s_2$ ,  $u_1$ ,  $u_2$ ,  $t_1$  and  $t_2$  are computed by solving equations (6)–(12) with  $s_1 = 0$ .

- Case 4.2: If

$$H \leq a_1,$$

then the optimal policy is given by **(OPT)** upon setting  $s_1 = 0$ ,  $s_2 = H$ , selecting  $u_2$  as any value in the interval  $[(\rho_2 - \frac{d_2/\mu_2}{H})^+, \frac{d_1/\mu_1}{H} - (\rho_1^h - 1)]$  and setting  $u_1 = 1 - u_2$ .

- Case 4.3: If

$$H \geq a_2,$$

then the optimal policy is given by **(OPT)** with  $s_1 = 0$  and  $s_2 = 0$ .

The following corollary immediately follows from the description of the optimal policy.

**Corollary 3.** *If*

- (i)  $Z_1(0) \geq \theta_1$  or,
  - (ii)  $Z_1(0) \leq \theta_1$ ,  $Z_1(0) \leq \theta_2$  and  $0 \leq \psi_1 \leq \psi_2$ ,
- then the policy with

$$\begin{aligned} \forall t \in (0, H) & \quad u_1(t) = 1, u_2(t) = 0, \\ \forall t \in (H, H + L) & \quad \text{Low-Period-Policy,} \end{aligned}$$

is optimal for all  $H \geq 0$  and  $L \geq 0$ .

Note that if the initial fluid levels satisfy the conditions in (i) or (ii), the policy described in Corollary 3 is optimal even when the length of the high period and the length of the low period are random variables.

**B. The case  $\rho^h > 1$ ,  $\rho_1^h < 1$ ,  $\rho^l < 1$**

In this section, we assume that  $\rho^h > 1$ ,  $\rho_1^h < 1$ ,  $\rho^l < 1$ . Then the optimal policy has the following structure:

$$\begin{aligned} \forall t \in (0, s_1) : & \quad u_2(t) = 1, u_1(t) = 0, \\ \forall t \in (s_1, s_2) : & \quad u_2(t) = \rho_2 - \frac{(\theta_2 - Z_2(s_1))/\mu_2}{a_1(s_1)}, \\ & \quad u_1(t) = 1 - u_2(t), \\ \forall t \in (s_2, s_3) : & \quad u_2(t) = 0, u_1(t) = 1, \\ \forall t \in (s_3, H) : & \quad u_2(t) = 1 - \rho_1^h, u_1(t) = \rho_1^h, \\ \forall t \in (H, H + L) : & \quad \text{Low-Period-Policy,} \end{aligned}$$

where

$$a_1(s_1) = \frac{(\theta_1 - Z_1(s_1))/\mu_1 + (\theta_2 - Z_2(s_1))/\mu_2}{\rho_1^h + \rho_2 - 1},$$

and  $s_1, s_2, s_3$  are given as

$$\begin{aligned} s_1 &= \max\{t : 0 \leq t \leq H, Z_2(t) \geq \theta_2, Z_1(t) \leq \theta_1\} \vee 0, \\ s_2 &= \max\{t : s_1 \leq t \leq H, Z_1(t) \leq \theta_1\} \vee s_1, \\ s_3 &= \max\{t : s_2 \leq t \leq H, Z_1(t) \geq \theta_1\} \vee s_2. \end{aligned}$$

**C. The case  $\rho^h < 1, \rho^l < 1$**

In this section, we assume  $\rho^h < 1, \rho^l < 1$ . Then the optimal policy is

$$\begin{aligned} \forall t \in (0, H) & \quad \text{Low-Period-Policy with } \rho_1^h \text{ replacing } \rho_1^l, \\ \forall t \in (H, H + L) & \quad \text{Low-Period-Policy.} \end{aligned}$$

**Remark 4.** *The policies described in subsections III-B and III-C can be implemented without knowing the length of the high and the low periods. Hence, these policies are also optimal when the length of the high period and the length of the low period are random variables.*

#### IV. RANDOM HIGH PERIOD

In this section, we consider the case when the duration of the high period  $H$  is a random variable with a known distribution. Thus the arrival rate function  $\lambda_1(s)$  is also random. We assume that the low period is long enough, hence at the end of each high-low cycle, everything will be cleared and the system will start over again in the next cycle. Recall that the system starts with a high period, followed by a low period. We want to find the policy which minimizes the expected cost within a high-low cycle. In particular, using the policy described in Section III, we will develop some heuristic policies. Recall that, for a given policy  $U$ , our objective is to minimize the expected total cost given by

$$\mathbb{E}_U \left[ \int_0^{H+L} \sum_{i=1}^2 h_i (Z_i(t) - \theta_i)^+ dt \right]. \quad (13)$$

Recall further that for case 1 and case 3 in Section 3.1 and for cases in Sections 3.2 and 3.3, we know the optimal policy when  $H$  is random. Thus, it suffices to specify the heuristic policies for cases 2 and 4 in Section 3.1.

Ideally, once the exact length of the high period  $H$  is known, one can follow the optimal policy in the deterministic case described in Section III. Let  $C(H)$  denote the total holding cost under the optimal policy when the length of the high period is known. Since one can not observe the true length of the high period until it ends, such a policy is not achievable. However, the quantity in (13) is always bounded below by  $\mathbb{E}[C(H)]$ . This lower bound (which will be referred as LB) can be used as a guideline to evaluate the performance of a policy. The details on the computation of this lower bound can be found in [4].

We will now consider a set of policies called the  $\{\pi^x\}$  policies. The policy  $\pi^x$  is obtained based on the optimal

policy in the deterministic case by setting  $H = x$  and making the necessary modifications once the length of the high period is realized. Note that the predetermined policy can be modified as follows, upon observing if the high period is longer or shorter than  $x$ . A sudden drop in the arrival rate of class 1 customers indicates the end of the high period. Let  $H_0$  denote the actual length of the high period. Then we can employ the following policy:

$$\begin{aligned}
\forall t \in (0, s_1(x) \wedge H_0) : & \quad u_2(t) = 1, u_1(t) = 0, \\
\forall t \in (s_1(x) \wedge H_0, s_2(x) \wedge H_0) : & \quad u_2(t) = u_2, u_1(t) = u_1 \\
& \quad u_1 + u_2 = 1, \\
\forall t \in (s_2(x) \wedge H_0, x \wedge H_0) : & \quad u_2(t) = 0, u_1(t) = 1, \\
\forall t \in (x \wedge H_0, H_0) : & \quad u_2(t) = 0, u_1(t) = 1, \\
\forall t \in (H_0, H_0 + L) : & \quad \text{Low-Period-Policy},
\end{aligned}$$

where  $s_i(x)$  is computed according to the optimal policy given in Section III by setting  $H = x$ .

Clearly, the value of  $x$  determines the performance of a  $\pi^x$  policy. We consider three different alternatives for the choice of  $x$ . The first option sets  $x = \mathbb{E}[H]$ . Note that this is a very simple policy which only requires the first moment of  $H$ . Another option is to use  $x^*$  which minimizes the expected holding cost in the set of  $\pi^x$  policies. In order to compute  $x^*$ , we first obtain a closed form expression for the expected holding cost as a function of  $x$  and then solve for  $x$  that minimizes this expression. Finally, we consider  $x'$  which gives the best policy depending on the region that  $\mathbb{E}[H]$  belongs to. Recall that in the deterministic case, the policy structure differs according to the interval that  $H$  is in. Thus, we choose  $x$  that minimizes the expected cost in the region that  $\mathbb{E}[H]$  belongs to. This optimal value of  $x$  is denoted by  $x'$ . We use MAPLE in the computation of both  $x^*$  and  $x'$ . However, since the computation of  $x'$  requires the holding cost expression only for a specific region, it is easier than the computation of  $x^*$ . We will denote these three policies by  $\pi^{\mathbb{E}[H]}$ ,  $\pi^{x^*}$  and  $\pi^{x'}$ .

### A. Numerical Results

We have carried out a set of numerical experiments to evaluate the performance of the  $\{\pi^x\}$  policies proposed above. We assume that the system starts empty and the high period  $H$  is an exponential random variable with mean  $\mathbb{E}[H]$ . We consider four different parameter sets as listed in Table I.

We compute the expected cost under  $\pi^{\mathbb{E}[H]}$ ,  $\pi^{x^*}$  and  $\pi^{x'}$  policies and compare them with the lower bound, LB. In particular, we compute the relative difference, defined as,  $\frac{C_\pi - \text{LB}}{\text{LB}}$ . Figure 1 provides a graph of this relative difference with respect to  $\mathbb{E}[H]$ .

Parameters	$\rho_1^h$	$\rho_1^l$	$\rho_2$	$\theta_1$	$\theta_2$	$\eta$	$\mu_1$	$\mu_2$
Setting 1	2	0.5	0.45	10	1	0.9	1	1
Setting 2	3	0.2	0.4	15	2	0.5	1	1
Setting 3	5	0.1	0.4	40	2	0.4	1	1
Setting 4	2	0.1	0.2	50	5	0.6	1	1

TABLE I  
FOUR DIFFERENT PARAMETER SETTINGS.

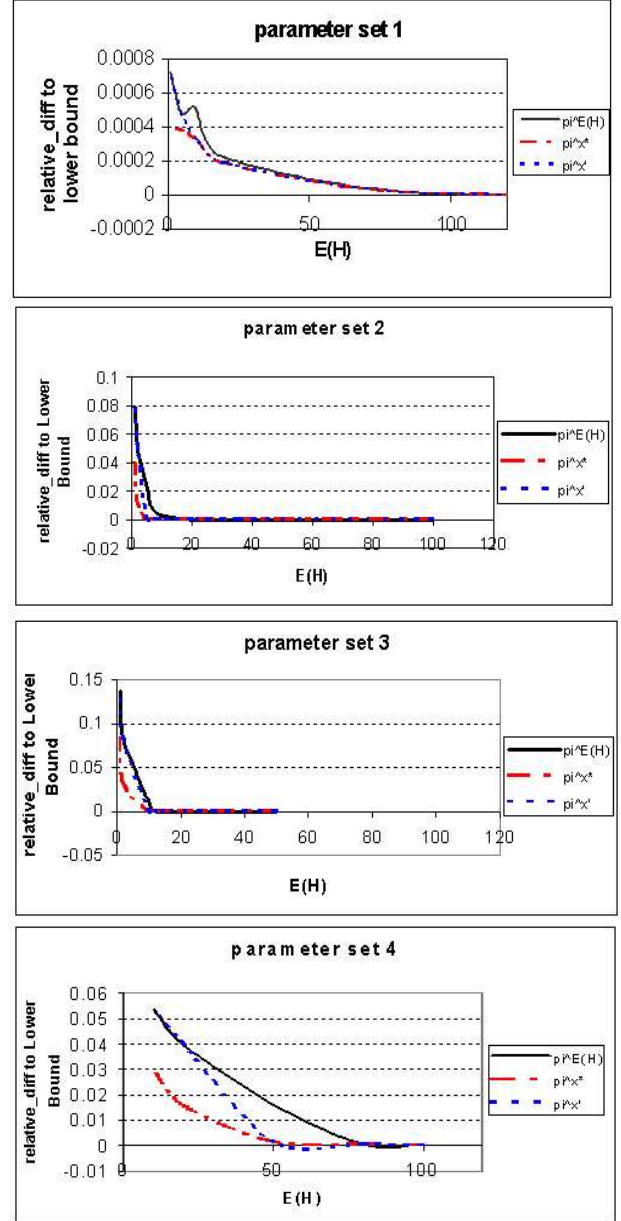


Fig. 1. Performance comparison under different parameter settings

From Figure 1, we conclude that the relative differences (to the lower bound) under the three policies are all quite small. When  $\mathbb{E}[H]$  is small, the relative difference to the

lower bound gets bigger. As expected,  $\pi^{x^*}$  gives the best performance, and  $\pi^{x'}$  gives better performance than  $\pi^{\mathbb{E}[H]}$ . However, the performance of  $\pi^{\mathbb{E}[H]}$  is not much worse than the performances of the other two policies. Therefore, one might prefer to use the  $\pi^{\mathbb{E}[H]}$  policy since the computation of  $x^*$  and  $x'$  is much harder and since the implementation of the  $\pi^{\mathbb{E}[H]}$  policy requires only the first moment of  $H$ . Note that these observations are valid under the assumption that  $H$  is an exponential random variable. Moreover, additional numerical experiments demonstrate similar trends under the assumption that  $H$  follows a uniform or hyperexponential distribution.

## V. SUMMARY AND CONCLUSIONS

We presented a study of the dynamic scheduling of different classes of service in a fluid model of computing paradigms for Internet services that may be overloaded for a transient period. In particular, we focused on minimizing the penalty of the hosting service provider by scheduling its server resources among various e-commerce sites under Service-Level-Agreement (SLA) contracts with specific Quality-of-Service (QoS) performance guarantees for each class of service.

We considered the case with two fluid classes and a single server whose capacity can be shared arbitrarily among the two classes. In order to capture the QoS performance guarantees in the SLA contracts, we introduced a threshold value for each fluid class such that a holding cost is incurred only if the amount of fluid of a certain class exceeds its threshold value. Under the assumptions that the class 1 arrival rate changes with time and the class 1 fluid can more efficiently reduce the holding cost, we specified the optimal server allocation policy that minimizes the holding cost in the corresponding fluid model when the arrival rate function for class 1 is known. Using the key insights gained from this deterministic case, we also developed heuristic policies for the stochastic fluid system when the arrival rate function for class 1 is random. Preliminary numerical examples demonstrated that these heuristic policies provide good performance in terms of the expected holding cost.

## VI. REFERENCES

- [1] AVRAM F., BERTSIMAS D. AND RICARD M.(1995). Fluid Models of Sequencing Problems in open Queueing Networks: an Optimal Control Approach, in *Stochastic Networks*, eds. F.P. Kelly and R.J. Williams 199-234.
- [2] N. Bansal and M. Harchol-Balder. Scheduling solutions for coping with transient overload. In, Technical Report *CMU-CS-01-134*, Department of Computer Science, Carnegie Mellon University, 2001.
- [3] N. Bauerle and U. Rieder. Optimal control of single-server fluid networks. *Queueing Systems - Theory and Applications*, 35, 185-200, 2000.
- [4] J. Chang, H. Ayhan, J.G. Dai and C.H. Xia. Dynamic scheduling of a multiclass fluid model with transient overload, under review, 2003.
- [5] H. Chen and P. Mohapatra. Session-based overload control in QoS-aware Web servers. In *INFOCOM 2002*.
- [6] A. Iyengar, E. MacNair and T. Nguyen. An analysis of web server performance. In *Proceedings of GLOBECOM'97*, 1997.
- [7] G.P. Klimov. Time sharing service systems I. *Theory of Probability and Its Applications*, 19(3):532–551, 1974.
- [8] G.P. Klimov. Time sharing service systems II. *Theory of Probability and Its Applications*, 23(2):314–321, 1978.
- [9] Z. Liu, N. Niclausse, and C. Jalpa-Villanueva. Traffic model and performance evaluation of Web servers. *Performance Evaluation*, 46:77–100, October 2001.
- [10] Z. Liu, M.S. Squillante, and J.L. Wolf. Optimal control of resource allocation in e-business environments with strict quality-of-service performance guarantees. In *Proceedings of the IEEE Conference on Decision and Control*, 2002.
- [11] L.S. Pontryagin, V.G. Boltyanskii, R.V. Gamkrelidze and E.F. Mishchenko. *The Mathematical Theory of Optimal Processes*, Interscience Publishers, New York, 1962.
- [12] A. Seierstad, and K. Sydsater. *Optimal Control Theory with Economic Applications*, North-Holland, Amsterdam (1987).
- [13] W. E. Smith. Various optimizers for single-stage production. *Naval Res. Logist. Quart.*, 3:59–66, 1956.