Delay Guarantee and Bandwidth Allocation for Network Services

Jyrki Joutsensalo[1], Timo Hämäläinen[1], Jarmo Siltanen[2], and Kari Luostarinen[3].
Department of Mathematical Information Technology
University of Jyväskylä
40014 University of Jyväskylä, FINLAND
Telephone: +358 14 2603292
[2]Jyväskylä Polytechnic
Jyväskylä
FINLAND
[3] Metso Paper Inc., P.O. Box 587
FIN-40101 Jyväskylä
FINLAND
Emails: jyrkij@mit.jyu.fi, timoh@mit.jyu.fi, jarmo.siltanen@jypoly.fi, and kari.luostarinen@metso.com

Abstract— **This paper presents a packet scheduling scheme for ensuring delay and bandwidth as a Quality of Service (QoS) requirement. For customers, rightful service is given while optimizing revenue of the network service provider. A gradient and fixed point type algorithms for updating the weights of a packet scheduler are derived from a revenue-based optimization problem. In the linear pricing scenario, algorithms are simple to implement. We compared algorithms with optimal brute-force method. Especially fixed point algorithm converges very fast to the optimal solution, typically in one iteration and about 40 operations, when number of classes is three. The weight updating procedures are independent on the assumption of the connections' statistical behavior, and therefore they are robust against erroneous estimates of statistics. Also, a Call Admission Control (CAC) is implemented in context of our scenario.**

Index Terms— **Pricing, revenue optimization, delay, bandwidth, Quality of Service (QoS).**

I. Introduction

The next generation networks will support real-time service, best-effort service, and other different classes of services, which have different QoS requirements in terms of bandwidth, delay, jitter, and packet loss. Due to the diversity of performance requirements and traffic characteristics, different classes of traffic should be treated separately according to their respective QoS requirements. Packet schedulers should isolate different classes of traffic and maintain fairness among them.

Several methods have been proposed in the literature for bandwidth allocation. A scheduling algorithm, called Relaxed-order Fair Queueing is presented in [1]. They demonstrate that the fairness may be achieved even when the order of packet serviced in packet switched networks is unconstrained or slightly constrained to the order of packet serviced in the GPS fluid model, i.e., the conditions to achieve fairness are relaxed.

A rate based protocol, called Adaptive Bandwidth Scheduling (ABS) is proposed in [2][3]. It has adaptive priority that is based on a combination of four priority types: space priority, time priority, loss priority and rate regulation. With the combination of different priorities in one scheme, this protocol can satisfy various desirable QoS properties. Thus, the serving order of packets from different flows can be adjusted according to the momentary needs of the individual flows.

An adaptive bandwidth allocation algorithm for determining the bandwidth and ensuring the defined QoS policy by using queue status and priority assignment is presented in [4]. The algorithm, called adaptive weighted fair queue (AWFQ) scheduling, distributes the bandwidth to efficiently utilize the network's resources.

Reference [5] presents a scheduling scheme, which is based on weighted packet scheduling policies, that adaptively change the scheduling weights of behavior aggregates. The proposed scheme is shown to achieve low loss rate, low delay and delay jitter for the premium service in the differentiated service (DiffServ) [6] architecture.

The maximization of the revenue has also been studied. Reference [7] proposes a configuration scheme for the worst case of network congestion to guarantee maximum revenue for the service provider in networks with WFQ schedulers. The knowledge of the topology is shown to be sufficient for feasible optimum network configuration.

This paper extends our previous studies [8], [9], where only delay was considered as QoS parameter. We present adaptive algorithms for optimizing the network operator revenue and for ensuring QoS requirements. In this paper, we add bandwidth as another QoS parameter to be observed.

The rest of the paper is organized as follows. In section II, packet scheduler as well as delays and bandwidth formulation are presented. In section, III, pricing models and revenue maximization is discussed. In that section, gradient and fixed point algorithms are introduced for fair service and revenue optimization. Section IV consists of experimental part while the next section discusses properties of the algorithms, experiments, implementation, and computational complexity.

Final section concludes the study.

II. THE PACKET SCHEDULER, DELAYS, AND BANDWIDTH

In this section, we formulate expressions for delays and bandwidth (bit rate) of the data traffic. Consider the packet scheduler, which has for example two queues. Gold class connections pay most of money while getting best service, and silver class connections pay least of money.

Parameter Δt_i denotes time which passes when data is transferred through the queue i to the output in the switch, when $w_i = 1$. If the queue is almost empty, delay is small, and when the buffer is full, it is large. Variable w_i is the weight allocated for class i. Constraint for weights w_i is

$$\sum_{i=1}^{m} w_i = 1, \quad w_i > 0. \tag{1}$$

Variables w_i give weights, how long time queues i are served per total time. Therefore, delay d_i in the queue i is actually

$$d_i = \frac{\Delta t_i}{w_i}. \tag{2}$$

Without loss of generality, only non-empty queues are considered, and therefore

$$w_i \neq 0, \quad i = 1, \ldots, m, \tag{3}$$

where m is number of service classes. When one queue becomes empty, $m \to m-1$. Parameter N_i denotes the number of such connections in the ith service class.

Bandwidth or bit rate is formulated as follows. Let the processing time of the data be T [seconds/bit] in the packet scheduler. There are N_i connections or packets in the class i. Let us denote the packet size b_{ij} [bits] or [kbytes] in the class $i = 1, \ldots, m$ and the connection $j = 1, \ldots, N_i$. It is easy to see that bandwidth of the packet (i, j) is

- linearly proportional to the packet size b_{ij},
- linearly proportional to the weight w_i,
- inversely proportional to the processing time T, and
- inversely proportional to the total sum of the packet lengths b_{ij}, $j = 1, \ldots, N_i$, because other packets occupy the same band in a time-divided manner.

Therefore, the expression for the bandwidth is

$$B_{ij}[bits/s] = \frac{b_{ij} w_i}{N_i E(b_i) T} = \frac{b_{ij} w_i}{N_i E(b_i)} = \frac{b_{ij} w_i}{\sum_{l=1}^{N_l} b_{il}} \tag{4}$$

where the processing time T can be scaled $T = 1$, without loss of generality. Here

$$E(b_i) = \frac{1}{N_i} \sum_{j=1}^{N_i} b_{ij} \tag{5}$$

is mean packet length in the class i.

III. PRICING MODELS AND REVENUE MAXIMIZATION

We concentrate on the pricing and fair resource allocation from the point of view of the customers. On the other hand, from the point of view of the service provider, we try to maximize revenue. First, we introduce the concept of *pricing functions*. In the scope of our study, there are two QoS parameters, namely delay and bandwidth. Therefore, two separate pricing functions are defined.

A. Delay

For delay, pricing functions are denoted by $f_i(d)$, where d is the delay, and f_i is decreasing with respect to d. In addition, $f_i(d)$ is (strictly) convex with respect to d for ensuring revenue more than $-\infty$. Revenue obtained for one user in the class i is just $f_i\left(\frac{\Delta t_i}{w_i}\right)$. Because there are N_i connections in the class i with all having the same delay, revenue corresponding to the delays in the class i is

$$R_i^{delay}(w_i) = N_i f_i\left(\frac{\Delta t_i}{w_i}\right). \tag{6}$$

In our study, we use *linear* pricing function for delays. Then

$$f_i(x) = -r_i x + k_i, \quad k_i > 0, \tag{7}$$

$$f_i'(x) = -r_i, \tag{8}$$

$$f_i''(x) = 0. \tag{9}$$

Here $k_i > 0$ guarantees positive revenue with minimum delay. For the classes that have better service, factors r_i are larger compared with those classes, that have service of lower priority. In addition, for classes having better service, k_i are larger. Notice that the pricing function may be even negative, when the delay is too large. However, Call Admission Control (CAC) mechanism takes care that this situation is prevented.

B. Bandwidth

For bandwidth, pricing functions are denoted by $g_i(B)$, where B is the bandwidth, and g_i is positive, increasing, and concave with respect to B. Concavity is natural feature; for example, when images are transferred, the price may be twice compared to the price when voice is transferred, while the bandwidth ratio - image bandwidth/voice bandwidth - is much more than two.

Revenue corresponding to the bandwidths in the class i is

$$R_i^{bandwidth}(w_i) = \sum_{j=1}^{N_i} g_i\left(\frac{b_{ij} w_i}{\sum_{l=1}^{N_i} b_{il}}\right). \tag{10}$$

In this study, we consider two kinds of pricing functions for bandwidth, namely *linear* and *polynomial* pricing functions. In the linear function,

$$g_i(x) = e_i x, \tag{11}$$

where e_i is positive and larger for the classes having higher priority. Allthogh this kind of the function is not *strictly* concave with respect to x, parameter e can be selected in such a manner that it is in "discrete" way concave with respect to the product $e_i x$, $i = 1 \ldots, m$. By "discrete concavity" we mean that

$$e_i - e_{i+1} < e_{i+1} - e_{i+2}, \quad e_i > e_{i+1}. \tag{12}$$

Polynomial pricing function has the form

$$g_i(x) = s_i x^p, \quad 0 < p < 1, \tag{13}$$

$$g_i'(x) = p s_i x^{p-1}, \tag{14}$$

$$g_i''(x) = p(p-1) s_i x^{p-2}. \tag{15}$$

Condition $0 < p < 1$ implies concavity of the pricing function. Factor s_i is larger in the classes that have better service.

It is worth to mention that our approach is general in the sense that the pricing functions can be changed to other ones.

C. Revenue maximization algorithms

We consider two kinds of approaches for maxizing revenue. Both the methods use Lagrangian constraint approach. The first one is a gradient algorithm, while the second one is a fixed point type algorithm. In the Lagrangian approach, we add two subrevenues with penalty factor, and total revenue has the form

$$\begin{aligned} R &= \sum_{i=1}^{m} [R_i^{delay}(w_i) + R_i^{bandwidth}(w_i)] + \lambda(1 - \sum_{i=1}^{m} w_i) \\ &= \sum_{i=1}^{m} N_i f_i \left(\frac{\Delta t_i}{w_i} \right) \\ &+ \sum_{i=1}^{m} \sum_{j=1}^{N_i} g_i \left(\frac{b_{ij} w_i}{\sum_{l=1}^{N_i} b_{il}} \right) \\ &+ \lambda(1 - \sum_{i=1}^{m} w_i). \end{aligned} \tag{16}$$

Taking derivative with respect to R, we get

$$\begin{aligned} \frac{\partial R}{\partial w_i} &= -N_i f_i' \left(\frac{\Delta t_i}{w_i} \right) \frac{\Delta t_i}{w_i^2} \\ &+ \sum_{j=1}^{N_i} g_i' \left(\frac{b_{ij} w_i}{\sum_{l=1}^{N_i} b_{il}} \right) \frac{b_{ij}}{\sum_{l=1}^{N_i} b_{il}} \\ &- \lambda = 0. \end{aligned} \tag{17}$$

Derivative with respect to the Lagrangian penalty term

$$\frac{\partial R}{\partial \lambda} = 0 \tag{18}$$

implies

$$\sum_{i=1}^{m} w_i = 1. \tag{19}$$

Then

$$\begin{aligned} \lambda &= \lambda \sum_{i=1}^{m} w_i = \sum_{i=1}^{m} \lambda w_i \\ &= -\sum_{i=1}^{m} N_i f_i' \left(\frac{\Delta t_i}{w_i} \right) \frac{\Delta t_i}{w_i} \\ &+ \sum_{i=1}^{m} \sum_{j=1}^{N_i} g_i' \left(\frac{b_{ij} w_i}{\sum_{l=1}^{N_i} b_{il}} \right) \frac{b_{ij} w_i}{\sum_{l=1}^{N_i} b_{il}}. \end{aligned} \tag{20}$$

Therefore derivative has the form

$$\begin{aligned} \frac{\partial R}{\partial w_i} &= -N_i f_i' \left(\frac{\Delta t_i}{w_i} \right) \frac{\Delta t_i}{w_i^2} \\ &+ \sum_{j=1}^{N_i} g_i' \left(\frac{b_{ij} w_i}{\sum_{l=1}^{N_i} b_{il}} \right) \frac{b_{ij}}{\sum_{l=1}^{N_i} b_{il}} \\ &+ \sum_{k=1}^{m} N_k f_k' \left(\frac{\Delta t_k}{w_k} \right) \frac{\Delta t_k}{w_k} \\ &- \sum_{k=1}^{m} \sum_{j=1}^{N_k} g_k' \left(\frac{b_{kj} w_k}{\sum_{l=1}^{N_k} b_{kl}} \right) \frac{b_{kj} w_k}{\sum_{l=1}^{N_k} b_{kl}}. \end{aligned} \tag{21}$$

Because gradient shows the direction of largest increase of the function R with respect to the variables w_i, the weights are updated according to the gradient algorithm. During one iteration step, weights are updated according to the rule

1) Update weights

$$v_i(t) = w_i(t) + \mu \left. \frac{\partial R(N_1(t), \ldots, N_m(t), w_i)}{\partial w_i} \right|_{w_i = w_i(t)}. \tag{22}$$

2) Perform scaling

$$w_i(t+1) = \frac{v_i(t)}{\sum_{j=1}^{m} v_i(t)}. \tag{23}$$

Here t denotes time, and $N_i(t)$ shows that the number of connections vary as a function of time. Parameter μ is a forgetting factor, which is either constant or depend inversely on the norm of the gradient. It ensures time-varying nature of the weights due to the nonstationary data traffic.

Let us next examine the concavity of R. If the second order derivative is negative, it is concave, having unique maximum. In that condition, algorithm converges to the global optimum. Second order derivative is examined in the interval $0 < w_i \leq 1$:

$$\frac{\partial^2 R}{\partial w_i^2} = \sum_{k=1}^{7} a_{ik}, \tag{24}$$

where

$$a_{i1} = N_i f_i'' \left(\frac{\Delta t_i}{w_i} \right) \frac{\Delta t_i^2}{w_i^4}, \tag{25}$$

$$a_{i2} = 2 N_i f_i' \left(\frac{\Delta t_i}{w_i} \right) \frac{\Delta t_i}{w_i^3} < 0, \tag{26}$$

$$a_{i3} = \sum_{j=1}^{N_i} g_i'' \left(\frac{b_{ij} w_i}{\sum_{l=1}^{N_i} b_{il}} \right) \frac{b_{ij}^2}{\left(\sum_{l=1}^{N_i} b_{il} \right)^2} < 0, \tag{27}$$

$$a_{i4} = -a_{i1} w_i, \tag{28}$$

$$a_{i5} = -a_{i2} w_i, \tag{29}$$

$$a_{i6} = -a_{i3} w_i, \tag{30}$$

$$a_{i7} = -\sum_{j=1}^{N_i} g_i' \left(\frac{b_{ij} w_i}{\sum_{l=1}^{N_i} b_{il}} \right) \frac{b_{ij}}{\sum_{l=1}^{N_i} b_{il}} < 0 \tag{31}$$

Notice that

$$a_{i2} + a_{i5} = a_{i2}(1 - w_i) < 0 \tag{32}$$

and

$$a_{i3} + a_{i6} = a_{i3}(1 - w_i) < 0. \tag{33}$$

From Eqs. (24)-(33) it is seen that it is possible to achieve necessary conditions for concavity for many kinds of functions, because almost all terms a_{ik} or their sums are negative. However, usually $a_{i1} \geq 0$, and therefore is is difficult so obtain simple rule guaranteeing concavity. The factor a_{i4} compensates positivity of a_{i1}, but $|a_{i4}| < |a_{i1}|$. However, general selections for functions are

$$f_i'(x) < 0, \tag{34}$$

$$f_i''(x) \geq 0, \tag{35}$$

$$g_i'(x) > 0, \tag{36}$$

and

$$g_i''(x) < 0. \tag{37}$$

If auxiliary condition

$$\frac{|f_i'(x)|}{x} \geq f_i''(x), \quad (a_{i1} + a_{i2} \leq 0), \tag{38}$$

is used, then R is strictly concave. For example, conveniently chosen logarithmic type function obeys this condition for positive variable. If

$$f_i(x) = -a_i ln(b_i x + c_i), \quad a_i, b_i, c_i > 0, \tag{39}$$

then Eq. (38) holds. In addition, $f_i'(x) < 0$ and $f_i''(x) > 0$.

Consider next two special cases:

1) f_i and g_i are linear.
2) f_i is linear, and g_i is polynomial.

1) Linear functions: When both functions are linear, revenue is

$$R = \sum_{i=1}^{m} -\frac{N_i r_i \Delta t_i}{w_i} + N_i k_i + e_i w_i + \lambda(1 - \sum_{i=1}^{m} w_i), \tag{40}$$

and derivative reduces to

$$\begin{aligned} \frac{\partial R}{\partial w_i} &= \frac{N_i r_i \Delta t_i}{w_i^2} \\ &+ e_i \\ &- \sum_{k=1}^{m} \left(\frac{N_k r_k \Delta t_k}{w_k} + e_k w_k \right). \end{aligned} \tag{41}$$

Uniqueness of the optimal solution in the interval $0 < w_i \leq 1$ is seen from the second order derivative

$$\begin{aligned} \frac{\partial^2 R}{\partial w_i^2} &= -\frac{2N_i r_i \Delta t_i}{w_i^3} \\ &+ \frac{N_i r_i \Delta t_i}{w_i^2} \\ &< 0, \end{aligned} \tag{42}$$

when $0 < w_i \leq 1$.

As shown earlier, gradient algorithm is a straightforward approach for optimizing the weights. The other one is faster. It is fixed point type algorithm, which equates

$$w_i = F(w_i). \tag{43}$$

From derivative (41) we see that

$$w_i = \sqrt{\frac{N_i r_i \Delta t_i}{\sum_{k=1}^{m} \left(\frac{N_k r_k \Delta t_k}{w_k} + e_k w_k \right) - e_i}}. \tag{44}$$

It is seen that

- in the numenator, larger the number of connections N_i is, larger is the weight,
- in the numenator, larger the "penalty" factor r_i is, larger is the weight,
- in the numenator, larger the delay Δt_i is, larger is the weight,
- in the denominator, larger the gain factor e_i is, larger is the weight.

All of these are very plausible results. Iteration is as follows:

1) Update the weight

$$v_i(t) = F[N_1(t), \ldots, N_m(t), w_1(t), \ldots, w_m(t)]. \tag{45}$$

2) Perform scaling

$$w_i(t+1) = \frac{v_i(t)}{\sum_{k=1}^{m} v_k(t)}. \tag{46}$$

2) Linear and polynomial functions: Consider next the case where f_i is linear and g_i is polynomial. Now gradient is

$$\begin{aligned} \frac{\partial R}{\partial w_i} &= \frac{N_i r_i \Delta t_i}{w_i^2} \\ &+ \frac{s_i}{2\sqrt{w_i}} \sum_{j=1}^{N_i} \sqrt{\frac{b_{ij}}{\sum_{l=1}^{N_i} b_{il}}} \\ &- \sum_{k=1}^{m} \frac{N_k r_k \Delta t_k}{w_k} \\ &- \sum_{k=1}^{m} \frac{s_k \sqrt{w_k}}{2} \sum_{j=1}^{N_k} \sqrt{\frac{b_{kj}}{\sum_{l=1}^{N_k} b_{kl}}}, \end{aligned} \tag{47}$$

and the second order derivative

$$\begin{aligned} \frac{\partial^2 R}{\partial w_i^2} &= -\frac{2N_i r_i \Delta t_i}{w_i^3} \\ &- \frac{s_i}{4w_i^{\frac{3}{2}}} \sum_{j=1}^{N_i} \sqrt{\frac{b_{ij}}{\sum_{l=1}^{N_i} b_{il}}} \\ &+ \frac{N_i r_i \Delta t_i}{w_i^2} \\ &- \frac{s_i}{4\sqrt{w_i}} \sum_{j=1}^{N_i} \sqrt{\frac{b_{ij}}{\sum_{l=1}^{N_i} b_{il}}} < 0 \end{aligned} \tag{48}$$

guarantees uniqueness of the solution. In this case, fixed point iteration uses the formula

$$w_i = \sqrt{\frac{N_i r_i \Delta t_i}{\sum_{k=1}^{m}\left(\frac{N_k r_k \Delta t_k}{w_k} + \frac{s_k\sqrt{w_k}}{2}\sum_{j=1}^{N_k}\sqrt{\frac{b_{kj}}{\sum_{l=1}^{N_k} b_{kl}}}\right) - \sigma_i}}, \quad (49)$$

where

$$\sigma_i = \frac{s_i}{2\sqrt{w_i}}\sum_{j=1}^{N_i}\sqrt{\frac{b_{ij}}{\sum_{l=1}^{N_i} b_{il}}}. \quad (50)$$

General form of the fixed point type iterative algorithm is as follows. Let

$$\frac{\partial R}{\partial w_i} = 0. \quad (51)$$

We obtain

$$\frac{c_i[w_1(t), \ldots, w_m(t)]}{w_i(t+1)^2} = h_i[w_1(t), \ldots, w_m(t)], \quad (52)$$

or

$$w_i(t+1) = \sqrt{\frac{c_i[w_1(t), \ldots, w_m(t)]}{h_i[w_1(t), \ldots, w_m(t)]}}, \quad (53)$$

where

$$c_i = N_i f_i'\left(\frac{\Delta t_i}{w_i}\right)\Delta t_i \quad (54)$$

$$\begin{aligned} h_i &= \sum_{j=1}^{N_i} g_i'\left(\frac{b_{ij} w_i}{\sum_{l=1}^{N_i} b_{il}}\right)\frac{b_{ij}}{\sum_{l=1}^{N_i} b_{il}} \\ &+ \sum_{k=1}^{m} N_k f_k'\left(\frac{\Delta t_k}{w_k}\right)\frac{\Delta t_k}{w_k} \\ &- \sum_{k=1}^{m}\sum_{j=1}^{N_k} g_k'\left(\frac{b_{kj} w_k}{\sum_{l=1}^{N_k} b_{kl}}\right)\frac{b_{kj} w_k}{\sum_{l=1}^{N_k} b_{kl}}. \end{aligned} \quad (55)$$

IV. Experiments

A. Experiment 1

In the first experiment, we examine the performance of the fixed point algorithm in the case, where both the pricing functions are linear, and the data traffic is static. By this experiment, we show how fixed point algorithm converges fast from arbitrary initial guess of the weights $w_i(0)$, $i = 1, 2, 3$ to the unique solution. In the first case, we select the parameters as follows:

- number of connections are $N_1 = 10$, $N_2 = 20$, $N_3 = 50$;
- penalty factors for delays are $r_1 = 10$, $r_2 = 5$, $r_3 = 2$;
- delays are $\Delta t_1 = 5$, $\Delta t_2 = 15$, $\Delta t_3 = 30$;
- shifting factors are $k_1 = 500$, $k_2 = 300$, $k_3 = 100$;
- gain factors for bandwidths are $e_1 = 20$, $e_2 = 17$, $e_3 = 10$.

100 simulations were performed using random initial guesses for weights. In all the cases, weights converge to the same solution *in one iteration step*. Of course, revenue curves also converge in the optimal way. Weights have the final values $(w_1, w_2, w_3) = (0.193, 0.334, 0.472)$. Revenue has the final value $R = 2590$.

We made also such an experiment, where brute-force method tested revenue in the grid $(w_1, w_2, 1 - w_1 - w_2)$, where $w_1, w_2 = 0.001, 0.002, 0.003, \ldots, 0.998$, $\sum_{i=1}^{3} w_i = 1$, $w_3 = 1 - w_1 - w_2$. Brute-force method produced revenue $R = 2590$, which is the same as the revenue obtained by fixed point algorithm.

Notice the for obtaining the same revenue for fixed-point and brute-force algorithms, only one iteration is needed in the fixed-point rule. Thus, total number of operations in three class case is as follows: 24 multiplications and 9 additions.

B. Experiment 2

In the second case, we changed the parameters to be

- number of connections are $N_1 = 15$, $N_2 = 25$, $N_3 = 70$;
- penalty factors for delays are $r_1 = 30$, $r_2 = 15$; $r_3 = 5$.
- delays are $\Delta t_1 = 10$, $\Delta t_2 = 25$; $\Delta t_3 = 50$.
- shifting factors are $k_1 = 10000$, $k_2 = 5000$, $k_3 = 3000$;
- gain factors for bandwidths are $e_1 = 30$, $e_2 = 25$, $e_3 = 22$.

Again, 100 simulations were performed. Again in all the cases, weights converge to the same solution in one iteration step. Weights have the final values $(w_1, w_2, w_3) = (0.227, 0.327, 0.447)$, and revenue is $R = 397290$. Brute-force method produced revenue $R = 397290$, which is the same as the revenue produced by the fixed point method in one iteration.

C. Experiment 3

In this experiment, we vary $N_i = exp(1), exp(2), \ldots, exp(20)$, $\Delta t_i = exp(1), exp(2), \ldots, exp(20)$. We compare fixed point method with one iteration step with brute-force method. The result is that the mean difference between two revenues is

$$|R_{fixedpoint} - R_{bruteforce}|/R_{bruteforce} = 0.07\%. \quad (56)$$

Conclusions made from this experiments are that *fixed point algorithm produces optimal weights in fast a way*.

V. Discussion

Here we discuss the properties of the algorithms as well as conclusions made by the experiments. We also converse implementation and computational complexity.

A. Algorithms and experiments

We make the following conclusions shown by algorithms and experiments:

- In the pricing scenario, we have derived analytic form to the revenue and gradient as well as fixed point algorithms for updating the weights w_i, which allocate data traffic to the connections of different service classes.
- The updating procedure is deterministic and nonparametric i.e. it does not make any assumptions of the statistical behavior of the traffic and connections. Thus it is robust against the errors that may occur from the wrong models.

- Algorithms are unique and optimal, which have been proved by Lagrangian optimization method.
- Fixed point solution makes algorithm quite simple, especially in the case, where both pricing functions are linear.
- Because all penalty and gain factors are positive, all classes obtain service in fair way.

B. Computational complexity

Consider first the general expression of the gradient (21). We see that the dominating role is to compute the triple sum, when $N_i \gg m$, $i = 1, \ldots, m$.We know that the number of classes is usually quite small - say $m = 3$ or $m = 4$. In addition, that triple sum is same for all w_i. Therefore, computational complexity per iteration is $O(\sum_{i=1}^{m} N_i^2)$.

Consider next the epression of the gradient (41) when both pricing functions are linear. Now the dominating role is to calculate the single sum. Again, it is the same for all weights. Then the number of computations per iteration reduces in both the gradient and fixed point algorithms to $O(m)$, if m is large. However, if m is small, then the role of the other terms increases. Total number of multiplications and additions is $8m$ and $3m$, respectively.

When f_i is linear and g_i is polynomial, there is double sum in the gradient (47). Thus computational complexity per iteration is $O(m \sum_{i=1}^{m} N_i)$.

VI. CONCLUSIONS

In this paper, we presented adaptive algorithms for optimizing the network operator revenue and for ensuring delay and bandwidth as QoS requirements. We derived the algorithms from a revenue-based optimization problem. The proposed fixed point weight updating algorithm was found to be computationally inexpensive in our scope of study. In the experiments we simulated the operation of the adaptive algorithm and compared it with a brute-force method. The obtained results show that by using the adaptive weights the revenue is the same than with the brute-force method. Our algorithms are deterministic and non-parametric, and thus we believe that in practical environments it is a competitive candidate due to its robustness. One task in the future is to prove analytically convergence of the fixed point algorithm, as well as to derive the speed of convergence.

REFERENCES

[1] H. Liao and W. Zhu, "Enable fair queueing with decoupled bandwidth-delay guarantees by relaxing the packet service order," in *ICC 2003 - IEEE International Conference on Communications*, 2003.

[2] D. H. Hoang and D. Reschke, "Adaptive bandwidth scheduling for broadband networks," in *Proc. IWK'98*, TU Ilmenau, Sept. 1998.

[3] D. H. Hoang and D. Reschke, "Integrated concept and simulation model for adaptive bandwidth scheduling scheme in ATM networks," in *Proc. IWK'98*, TU Ilmenau, Sept. 1998.

[4] M.-F. Horng, W.-T. Lee, K.-R. Lee, and Y.-H. Kuo, "An adaptive approach to Weighted Fair Queue with QoS enhanced on IP network," in *Proc. IEEE Region 10 International Conference on Electrical and Electronic Technology (TENCON'01)*, vol. 31, 2001, pp. 181–186.

[5] H. Wang, C. Shen, and K. G. Shin, "Adaptive-weighted packet scheduling for premium service," in *Proc. IEEE International Conference on Communications (ICC'01)*, vol. 6, 2001, pp. 1846 – 1850.

[6] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services," IETF RFC 2475, Dec. 1998.

[7] E. Magaña, D. Morató, and P. Varaiya, "Router scheduling configuration based on the maximization of benefit and carried best effort traffic," *Telecommunication Systems*, vol. 23, no. 2-4, p. 275–292, Oct.-Dec. 2003.

[8] J. Joutsensalo, T. Hämäläinen, M. Pääkkönen, and A. Sayenko, "QoS- and revenue aware adaptive scheduling algorithm," *Journal of Communications and Networks*, vol. 6, no. 1, pp. 68–77, Mar. 2004.

[9] J. Joutsensalo, T. Hämäläinen, K. Luostarinen, and J. Siltanen, "Adaptive scheduling method for maximizing revenue in flat pricing scenario," *International Journal of Electronics and Communications*, 2005.