Traffic Optimization with Iterative Weighted Scheduling Method

Timo Hämäläinen, Jyrki Joutsensalo, Kimmo Kaario, and Ari Viinikainen
Department of Mathematical Information Technology
University of Jyväskylä
40014 Jyväskylä, FINLAND

Abstract—In the future most communication networks (wireless and wired) will be required to provide support to a number of different types of traffic, each with its own particular characteristics and quality of service parameters including e.g. guaranteed bandwidth, jitter, and latency. The customers of different classes pay different prices to the service provider, who must share resources in a plausible way. Differentiation can be implemented by using a multiqueue system, where each queue corresponds to one service class. In this paper, an adaptive Weighted Fair Queue based algorithm for traffic allocation is presented and studied in the single node case. The weights in the fixed point type fast adaptive WFQ algorithm are updated using revenue as a target function. Due to the adaptive nature of the algorithm, it can operate in the nonstationary environments. In addition, it is nonparametric and deterministic in the sense that any assumptions about call density functions or duration distributions are not made.

I. INTRODUCTION

Integrated packet switched service networks must carry a wide range of different traffic types being still able to provide performance guarantees to realtime traffic such as Voice over IP (VoIP), Video-on-Demand (VoD), or videoconferencing and at the same time give some capacity to the best effort traffic. Service differentiation and management are issues which need careful Quality of Service (QoS) design in modern networks. In QoS design, different demands of different types of traffic (VoIP, VoD etc.) and different subclasses with different prices (gold, silver, bronze etc.) must be taken account for giving plausible and fair service.

Service differentiation research can be divided into two different concepts. The absolute service differentiation model can be implemented with a state-less core architecture that uses QoS parameter carried by packet header to provide fair-queuing [9] or guaranteed delay [10]. In the relative service differentiation model [1], [2], the traffic flows are grouped into different service classes which are ordered, such that class \( m \) is better (or at least no worse) than class \( i - m \) in terms of per-hop performance metrics.

Queuing policy is one basic principle for allocating service. Most popular queuing algorithms are priority queue [7] and weighted fair queue (WFQ) [8]. Priority queue absolutely prefers classes with higher priority, but it is nonadaptive and unfair, i.e. the delay in the low priority queue may increase unreasonable large. In contrast, WFQ gives weights for different classes in such a way that the performance of the low priority queues is guaranteed. An another QoS based queuing method is class based queuing (CBQ). By using CBQ and link-sharing mechanisms the link bandwidth can be divided to a number of different classes, and the traffic can be isolated between these classes. If individual traffic sources within a class are to be given a service guarantee, an additional acceptance procedure must be deployed. This procedure will provide a probabilistic, and less conservative, delay bound than that provided by sorted priority algorithms [3].

In this paper we extend our previous pricing and QoS research [5], [6], to take into account queuing issues. An adaptive WFQ based algorithm for traffic allocation is presented and studied. The weights in gradient type WFQ algorithm are adapted using revenue as a target function to be maximized. Adaptive WFQ can also be used as an admission control mechanism, but this is the topic in the future research. The close research is [4], where adaptive WFQ algorithm was investigated, but revenue criterion did not used. Our algorithm uses the following input parameters: (1) minimum processing time of the classifier for transmitting data from one queue to the output; (2) number of customers in the queues; (3) delay parameters such as insertion delay, transmission delay etc. Due to the adaptive nature of the algorithm, it can operate in the nonstationary environments. In addition, it is nonparametric in the sense that any call densities or duration distributions are assumed. The performance of the algorithm is investigated in the three service class case.

II. ALGORITHM

Here the algorithm is presented in the simplified form. Let \( d_o \) be the minimum processing time of the classifier for transmitting data from one queue to the output in Fig. 1. For simplicity an assumption that data packets have the same size. The number of service classes is denoted by \( m \). In WFQ, the real processing time (delay) is \( d = N_i d_o / w_i \), where \( w_i(t) = w_i \), \( i = 1, \ldots, m \) are weights allotted for each class, and \( N_i(t) = N_i \) is a number of customers in the \( i \)th queue. Here time index \( t \) has been dropped for convenience. The constraint for the weights are

\[
w_i > 0
\]
and
\[
\sum_{i=1}^{m} w_i = 1. \quad (2)
\]
If some weight is \( w_i = 1 \), then the other weights are \( w_j = 0, j \neq i, \) and class \( i \) is served by minimum processing time \( d_0 \), if \( N_i = 1 \). For each service class, a revenue or **pricing function**
\[
r_i(d) = r_i(N_i d_0 / w_i + c_i) \quad \text{(3)}
\]
(euros/minute) is decreasing with respect to the delay \( d \). Here \( c_i(t) = c_i \) includes insertion delay, transmission delay etc., and here it is assumed to be constant. A goal is to maximize revenue criterion
\[
F(w_1, \ldots, w_m) = \sum_{i=1}^{m} N_i r_i (N_i d_0 / w_i + c_i) \quad \text{(4)}
\]
under the weight constraint 1 and 2. As a special case, consider **linear revenue model**.

**Definition:** The function
\[
r_i(t) = -r_i t + k_i, \quad i = 1, \ldots, m,
\]
\[
r_i > 0, \quad \text{(5)}
\]
\[
k_i > 0, \quad \text{(6)}
\]
is called linear pricing function.

**Theorem 1:** Consider the linear pricing function (5) and the corresponding revenue function
\[
F = F(w_1, \ldots, w_m, N_1, \ldots, N_m) = \sum_{i=1}^{m} N_i (-r_i N_i / w_i + k_i), \quad \text{(8)}
\]
where \( d_0 = 1 \) and \( c_i = 0 \) for convenience. Then upper bounds for buffer sizes are
\[
qu_i = \left\lfloor \frac{k_i}{2 r_i} \right\rfloor, \quad i = 1, \ldots, m, \quad \text{(9)}
\]
where \( y = \lfloor x \rfloor \) denotes maximum integer \( y \) satisfying \( y \leq x \).

**Proof:** The optimal number of users for fixed weights is obtained as follows:
\[
\frac{\partial F}{\partial N_i} = -2 \frac{r_i k_i}{w_i} N_i + k_i = 0.
\]
Therefore
\[
N_i = \frac{1}{2} \frac{w_i k_i}{r_i}, \quad i = 1, \ldots, m. \quad \text{(11)}
\]
The second derivative is
\[
\frac{\partial^2 F}{\partial N_i^2} = -2 \frac{r_i}{w_i} < 0,
\]
because \( r_i > 0 \) and \( w_i \geq 0 \). Therefore \( F \) is strictly concave with respect to \( N_i, i = 1, \ldots, m \) having one and only one global maximum, which is satisfied by Eq. (11). Because \( w_i \leq 1, i = 1, \ldots, m, \) then
\[
N_i \leq \frac{1}{2} \frac{k_i}{r_i},
\]
for which Eq. (9) follows. This completes proof. **Q.E.D.**
The solution (11) is plausible and easy to interpret:
- When \( w_i \) is large, then it gives large weights to those buffers, where the number of customers is large to prevent too large delay to those buffers.
- Positive \( k_i \) increases revenue. It is simply positive constant vertical shift. Thus, larger \( k_i \) is, larger number of customers bring larger revenue.
- Negative \(-r_i \) in Eq. (8) has an opposite effect than \( k_i \).
  Thus, number of customers is inversely proportional to \( r_i \). The coefficient is a kind of a penalty term.

Upper bound for revenue is stated as follows:

**Theorem 2:** In the case of linear pricing model (5), upper bound for revenue is
\[
F \leq \frac{1}{4} \sum_{i=1}^{m} \frac{k_i^2}{r_i} \quad \text{(14)}
\]

**Proof:** Select optimal value for \( N_i \) in Eq. (11), and substitute it in Eq. (8). Then
\[
F = \sum_{i=1}^{m} \frac{1}{2} \frac{w_i k_i}{r_i} \left(-r_i \frac{1}{2} \frac{w_i k_i}{r_i} + k_i \right) = \frac{1}{4} \sum_{i=1}^{m} \frac{w_i k_i^2}{r_i} \quad \text{(15)}
\]
Due to the condition \( w_i \leq 1 \), Eq. (14) follows. **Q.E.D.**

Interpretation of (14) is quite obvious: \( k_i \) increases upper limit, while \( r_i \) decreases it.

As a special case, when all buffers are full according to the rule (9), we get the following result:

**Theorem 3:** When
\[
N_i = \frac{1}{2} \frac{k_i}{r_i},
\]
revenue is
\[
F = \frac{1}{2} \sum_{i=1}^{m} \frac{k_i^2}{r_i} (1 - \frac{m}{2}). \quad \text{(17)}
\]
Proof is omitted. It is clear that in practice the buffer sizes must be selected smaller than in Eq. (9). As a special case, when the there is only one class, i.e. \( m = 1 \), upper bound...
(14) can be achieved, but not for the other values of $m$, if the buffers are full.

The following theorem gives sufficient condition for achieving non-negative revenue as well as an other upper bound for revenue:

**Theorem 4:** If weights are selected by

$$w_i = \frac{N_i r_i / k_i}{\sum_{j=1}^{m} N_j r_j / k_j},$$  

(18)

and a constraint

$$\sum_{i=1}^{m} \frac{N_i r_i}{k_i} < 1,$$  

(19)

is used in the call admission control mechanism, then

$$0 \leq F \leq \sum_{i=1}^{m} N_i k_i.$$  

(20)

**Proof:** Define

$$a = \sum_{i=1}^{m} N_i r_i / k_i.$$  

(21)

Revenue is

$$F = \sum_{i=1}^{m} \left( -\frac{r_i N_i^2}{k_i w_i} \frac{k_a}{N_i r_i} + N_i k_i \right)$$

$$= \sum_{i=1}^{m} (N_i k_i a + N_i k_i)$$

$$= \sum_{i=1}^{m} N_i k_i (1 - a)$$

$$= \sum_{i=1}^{m} N_i k_i \left( 1 - \sum_{i=1}^{m} \frac{N_i r_i}{k_i} \right) \geq 0,$$  

(22)

when constraint (19) is satisfied. Because $N_i \geq 0$, $r_i > 0$, $k_i > 0$, then $0 \leq a < 1$. Then it follows that

$$F = \sum_{i=1}^{m} N_i k_i (1 - a) < \sum_{i=1}^{m} N_i k_i.$$  

(23)

This completes proof. Q.E.D.

Next we derive an adaptive algorithm for updating optimal weights using Lagrange type approach. We define the revenue as follows:

$$F(w_1, \ldots, w_m) = \sum_{i=1}^{m} N_i r_i \left( \frac{N_i}{w_i} \right) + \lambda (1 - \sum_{i=1}^{m} w_i^2).$$  

(24)

Here we must emphasize that adaptive WFQ weights are $v = w^2$ instead of $w$. For linear pricing functions, we propose the following theorem:

**Theorem 5:** For linear pricing functions, $F$ has one maximum.

**Proof:** In the linear case, Eq. (24) has the form

$$F(w_1, \ldots, w_m) = \sum_{i=1}^{m} N_i (r_i \frac{N_i}{w_i} + k_i) + \lambda (1 - \sum_{i=1}^{m} w_i^2)$$  

(25)

Then

$$\frac{\partial F}{\partial w_i} = -r_i N_i^2 (-2) w_i^{-3} - 2 \lambda w_i = 0.$$  

(26)

It follows that

$$r_i N_i^2 w_i^3 = \lambda w_i.$$  

(27)

On the other hand,

$$\frac{\partial F}{\partial \lambda} = 1 - \sum_{i=1}^{m} w_i^2 = 0,$$  

(28)

and

$$\sum_{i=1}^{m} w_i^2 = 1.$$  

(29)

So

$$\sum_{i=1}^{m} w_i \lambda w_i = \sum_{i=1}^{m} w_i^2 \lambda w_i = \sum_{i=1}^{m} \frac{r_i N_i^2}{w_i^3}.$$  

(30)

Therefore

$$\frac{\partial F}{\partial w_i} = 2 \frac{r_i N_i^2}{w_i^2} - 2 \lambda w_i \sum_{i=1}^{m} \frac{r_i N_i^2}{w_i^3}$$  

(31)

The second derivative is

$$\frac{\partial^2 F}{\partial w_i^2} = -6 \frac{r_i N_i^2}{w_i^4} - 2 \sum_{i=1}^{m} \frac{r_i N_i^2}{w_i^2} + 4 \frac{r_i N_i^2}{w_i^3}$$

$$= r_i N_i^2 \left( \frac{4}{w_i^2} - \frac{6}{w_i^4} \right) - 2 \sum_{i=1}^{m} \frac{r_i N_i^2}{w_i^3} < 0,$$  

(32)

because $r_i > 0$ and

$$\frac{4}{w_i^2} - \frac{6}{w_i^4} = \frac{4 w_i^2 - 6}{w_i^2} < 0$$  

(33)

So $F$ is strictly concave, having one maximum. Q.E.D.

Adaptive algorithm can be derived by setting

$$\frac{\partial F}{\partial w_i} = 0$$  

(34)

in Eq. (31). After some algebraic manipulations, Eq. (31) can then be represented in the form

$$v_i = \frac{\sqrt{\sum_{i=1}^{m} r_i N_i^2 / k_i}}{\sum_{i=1}^{m} r_i N_i^2 / k_i},$$  

(35)

$$v_i = w_i^2.$$  

(36)

Thus adaptive rule for updating weights can be realized directly by using Eq. (35) with known number $N_i(t)$ ($t$ is time index) of customers, and using normalization

$$v_i (t + 1) = v_i(t) / \sum_i v_i(t)$$  

(37)

after each iteration. The algorithm is conjectured to be a fixed point rule, but here we have not an analytic proof of that claim.
III. SIMULATIONS

In the simulation, three service classes \((m = 3)\) have pricing functions

\[ r_1(t) = -5t \quad (38) \]

for gold class,

\[ r_2(t) = -2t \quad (39) \]

for silver class, and

\[ r_3(t) = -t \quad (40) \]

for bronze class. Pricing (revenue) functions are shown in Fig. 2. Without loss of generality, we have selected \(k_i = 0\), because it has no effect on the algorithm; \(k_i\) has only the vertical shifting effect on the revenue (and thus should be positive in practical application: largest for highest-priority class, and smallest for lowest-priority class). To guarantee fair queue, revenues may be even negative, i.e. service provider must pay money to the customer, if the delay is too large. In the simulations, our purpose is to demonstrate the convergence speed of the algorithm. Therefore, we have selected time-invariant traffic profile for giving the intuition behind the curves. In the practical situation, traffic tends to be highly nonstationary with both the arrival rates as well as durations. However, it is emphasized that the algorithm updates adaptively and time-dependently weights, when number of customers \(N_i(t)\) varies with time \(t\). We selected the number of customers as follows: \(N_1 = 100\) for gold class, \(N_2 = 500\) for silver class, and \(N_3 = 1000\) for bronze class. To demonstrate convergence as well as Theorem 5, we gave random initial guesses to the weights. 100 simulations were performed.

Figures 3, 4, and 5 show that the weights converge from all 100 random initial guesses to the same solution, thus experimentally showing one global optimum. Figures 6, 7, and 8 show convergence of the weights in the logarithmic scale as a function of iteration index \(t\), i.e. \(10 \log_{10}(|v_3(t+1) - v_3(t)|)\). Here it is illustrated that the convergence tends to be linear in the logarithmic scale (and thus quite fast). In the final
Figure 6, revenue is plotted in the logarithmic scale. It is not surprising that it achieves global optimum due to the Theorem 5.

IV. DISCUSSION

Here, we present conclusions from our approach as well as experiments. Also some future topics as discussed. The conclusions are as follows:

- We have analytically shown that in the case of linear pricing scenario, revenue has one maximum. Proof was based on the concavity of the revenue function.
- Fixed point type algorithm was derived for maximizing the revenue. Experiments clearly justify the convergence of the algorithm. However, our goal is to give analytic convergence proof in the future.
- The algorithm is deterministic and non-parametric, ie. it uses only the information about the number of customers, not about statistics of arrival rates or duration distributions.
- The algorithm tends to have oscillative behavior, as shown in those figures representing the evolution of the weights. However, there are potentially lots of more effective fixed point rules, which can be derived from the solution of the first order derivative of the Lagrangian-based revenue criterion.
- The algorithm seems to converge linearly in the logarithmic scale.
- Also, Call Admission Control (CAC) mechanism can be used in the context of the algorithm. It is based on the hypothesis testing. This is one of our future research topics.
- Here, we investigated only the single node case. Multinode case is a much more challenging problem.
- The algorithm used the same packet sizes. However, it is quite straightforward to develop the version, which can handle different packet sizes.
General conclusion is that the linear pricing scenario is quite simple, perhaps nonpractical. However, we believe that practical pricing scheme should be based on piecewise linear model. Studies to that direction are made.

V. CONCLUSIONS

In this paper, we introduced an adaptive WFQ algorithm, which was derived from revenue target function. The experiments demonstrated the revenue maximization ability of the adaptive WFQ algorithm, while still allocating delays in a fair way.

In the future work, revenue criterion is used as an admission control mechanism. In admission control, call is accepted/rejected by hypothesis test, where revenue increase/decrease is estimated, when call is observed. Also multinode case is investigated in the future. It is important to develop such a distributed approximation, which does not suffer the curse of dimensionality and computational complexity of the optimal global approach. Others than strictly linear pricing models are studied.

REFERENCES