3-20-2007

An Optimal Bandwidth Allocation and Data Droppage Scheme for Differentiated Services in a Wireless Network

Waseem Sheikh
Purdue University, waseem@purdue.edu

Arif Ghafoor
Purdue University, ghafoor@purdue.edu

An Optimal Bandwidth Allocation and Data Droppage Scheme for Differentiated Services in a Wireless Network

Waseem Sheikh, *Member, IEEE*, Arif Ghafoor, *Fellow, IEEE*

Abstract— We present an optimal proportional bandwidth allocation and data droppage scheme to provide differentiated services (DiffServ) for streaming multimedia data in a single-hop wireless network. Our resource allocation scheme finds the optimal bandwidth allocation and data drop rates under minimum quality-of-service (QoS) constraints. It combines the desirable attributes of relative DiffServ and absolute DiffServ approaches. In contrast to relative DiffServ approach, our scheme guarantees the minimum amount of bandwidth provided to each user without dropping any data at the base-station, when the network has sufficient resources. If the network does not have sufficient resources to provide minimum bandwidth guarantees without data droppage to all users, the proportional data dropper finds the optimal data drop rates within acceptable levels of QoS and thus avoids the inflexibility of absolute DiffServ approach. We formulate the optimal bandwidth allocation and data droppage as constrained non-linear optimization problems and solve these using efficient techniques. We demonstrate through simulations that our scheme exhibits the desirable features of the absolute and relative DiffServ approaches.

Index Terms—Bandwidth allocation, data droppage, quality-of-service (QoS), differentiated services (DiffServ), wireless network.

I. Introduction

Today's wireless networks must be able to provide different levels of quality-of-service (QoS) to mobile users. This diversity in the service provided to mobile users stems from the fact that users have subscribed to different service classes, have different portable devices with different playout capabilities, and are accessing various types of multimedia data such as audio, video and images. Providing differentiated services to mobile users in a wireless network is a challenging problem which needs to take into account the constraints and heterogeneities of the wireless network along with the diverse QoS requirements of users. The random time-varying fading effect of the wireless channel makes the provision of diverse services more complex by requiring that the resource allocation scheme for service differentiation should be able to dynamically adjust to the time varying nature of the wireless channel. In this paper we provide a cross-layer optimized optimal bandwidth allocation and data droppage scheme, implemented at the base-station, to provide differentiated services for multimedia data in a single-hop mobile wireless network. Our scheme adapts to the random variations in channel capacity and in the user's bandwidth requirements and finds the optimal bandwidth allocation and data droppage rate for differentiated services under minimum QoS constraints. Our service differentiation scheme combines the desirable features of the absolute and relative DiffServ approaches. Our scheme finds the optimal separation between the bandwidth allocations to the users under minimum bandwidth constraints, whenever the network has sufficient resources to provide minimum bandwidth guarantees, without any data droppage, at the base-station to all the users. If the minimum bandwidth requirement of users can not be provided, we provide an optimal mechanism for dropping the data at the base station according to the DiffServ class weights. The data droppage scheme provides a mechanism for graceful degradation of video quality by finding the optimal values of the data drop rates according to the various service classes.

We formulate the optimal bandwidth allocation and data droppage problems as non-linear optimization problems, discuss their efficient solution techniques and evaluate our scheme against weighted proportional fair scheduling (WPFS) with simulations.

We make the following practical assumptions in this paper. We assume that all of the multimedia data is available at the base-station before its playout deadline. This is realistic because the wired backhaul network has sufficient resources to transmit multimedia data to the base-station before its playout deadline at the mobile terminal. We assume that the mobile users can tolerate a fixed maximum playout delay. Since for streaming multimedia data the playout delay depends on how much bandwidth is allocated to the user, we assume that the users provide a minimum bandwidth request corresponding to their

Waseem Sheikh is a Ph.D. Candidate in School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: waseem@purdue.edu).

Arif Ghafoor is with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: ghafoor@ecn.purdue.edu).

maximum playout deadline and buffer capacity. We do not consider queuing delay at the base-station, since it is reasonable to assume that the base-station has sufficient memory to store all the multimedia data.

The proposed resource allocation with service differentiation scheme has the following novel features and advantages over the existing schemes:

First, the proposed scheme combines the desirable features of the absolute and relative DiffServ schemes as follows. Under high load conditions, a relative DiffServ scheme keeps the service classes separated by the same relative distance even if the minimum acceptable QoS requirements of certain connections are being violated. This is especially undesirable for streaming multimedia applications which require minimum bandwidth guarantees for acceptable quality-of-presentation. On the other hand an absolute DiffServ scheme is also undesirable since it may be infeasible to provide absolute service guarantees especially in a wireless network with limited and time-varying resources. In addition, absolute DiffServ provides an inflexible admission control mechanism. Under absolute DiffServ, a request may be denied because the system can not provide absolute service guarantees, however, if the distances between the service classes are readjusted, new users may be admitted to the system. Our scheme provides optimal service class separation while ensuring minimum QoS guarantees to all the users.

Second, the proposed scheme provides a flexible admission control mechanism. A relative service differentiation scheme does not provide an admission control mechanism whereas the absolute service differentiation scheme provides an inflexible admission control. Our scheme allows the base-station to admit new users while keeping the different service classes separated and their degradation within acceptable bounds.

Third, the granularity of service differentiation in our scheme is per-flow basis. The traditional DiffServ mechanism provides only aggregate based granularity of service differentiation. Our scheme gives better and fine-grained control of resource allocation than the traditional DiffServ model. As a consequence, the accounting and billing can be done on per flow basis whereas in the traditional DiffServ scheme it is done on aggregate basis [1].

Fourth, the proposed scheme involves very little signaling overhead between mobile users and the base-station. The only parameters needed by the base station to solve the optimization problem are the DiffServ class weights, minimum bandwidth requirement, maximum data drop rate and the predicted channel capacity of all the users.

Finally, the proposed resource allocation scheme suits many adaptive multimedia applications such as multilayered encoded video, where users can tolerate certain degradation in the quality of service. It allows the users to specify the maximum degradation in QoS acceptable to them.

The paper is organized as follows. In Section II we discuss the related work in the area of providing differentiated services. Section III formulates the cross-layer optimized optimal bandwidth allocation and data droppage problem for providing differentiated services in a wireless network. Section IV discusses the implementation of our solution techniques and the simulation environment for evaluation. In section V we present the simulation results of our proposed solution and show its desirable features. Finally, we conclude the paper in Section VI.

II. RELATED WORK

In [1] the authors compare different service differentiation schemes such as absolute DiffServ, relative DiffServ, and Proportional DiffServ (Prop-DiffServ). The authors state that the absolute DiffServ approach can offer absolute assurances, which are mainly useful for unelastic applications. The relative DiffServ provides a mechanism for the users to select a service class that best matches the quality-cost tradeoff. The authors propose the Prop-DiffServ model as a special case of relative DiffServ. This model allows the network operator to control the quality spacing between classes independent of class loads. The authors also describe a packet scheduling and buffer management mechanism for approximating the Prop-DiffServ model. However, this model is completely independent of class loads which may not be desirable when the minimum QoS guarantees of the users are not being met. The authors in [2] present a bandwidth allocation mechanism for differentiated services on streaming servers. This mechanism aims to deliver high bit rate streams to high priority requests without over-compromising low priority requests. The authors formulate the bandwidth allocation problem as optimization of a harmonic utility function of stream quality factors and derive optimal streaming bit rates for requests of different classes under various server load conditions. However, this scheme is not applicable to transmission of multimedia data over wireless networks. The authors in [3] address the problem of providing concurrently a wide range of end-to-end throughput and delay assurances in wireless ad hoc networks. Their proposed solution is based on the neighborhood proportional delay differentiation (NPDD) service model. Using game theoretic concepts, the authors model dynamic class selection (DCS) applications in an NPDD network as selfish players in non-cooperative game. In [4], the authors propose a stateless network model which uses distributed control algorithms to deliver service differentiation in wireless ad hoc network. However the previous two schemes address the problem in a wireless ad hoc network which is different from the problem considered in this paper.

Our work differs considerably from the previous work in the area of service differentiation as follows. Our bandwidth allocation scheme provides optimal bandwidth allocation for proportional bandwidth allocation under minimum bandwidth constraints. If the network can not provide minimum bandwidth guarantees to all mobile users, we provide a mechanism of dropping data at the base-station by finding the optimal data drop rates for proportional data droppage at the base-station.

III. Optimal Resource Allocation for Differentiated Services

In this section we formulate the optimal proportional bandwidth allocation and optimal proportional data droppage problems for differentiated services in a mobile wireless network. Fig. 1 shows the time-line of the implementation of our resource allocation scheme. The resources are allocated in advance for a time interval of fixed length, called an "optimization window". Each optimization window is divided into M number of time-slots. We assume that the base-station can transmit to only one user during one time-slot. We solve the time-slot allocation problem for each optimization window of length T_w before its start. As shown in the Fig. 1, at time $kT_w - T_{opt}$ the mobile users predict the average channel SINR for the $(k+1)th$ optimization window and send this information to the base-station. The time interval T_{opt} is selected to make sure that the base-station can solve the problem for the $(k+1)th$-optimization window, within T_{opt} units of time. The base-station then solves the optimization problem and determines the resource allocated to different users for the $(k+1)th$-optimization window before the start of $(k+1)th$-optimization window. This process is repeated in this manner.

A. Optimal Proportional Bandwidth Allocation for Differentiated Services

Our problem formulation requires that each user specifies four QoS related parameters. These include an estimate of the average SINR for the next optimization window, DiffServ class weight, minimum bandwidth requirement (without data droppage at the base-station) and maximum data droppage requirements. The minimum bandwidth requirement, denoted by b_i, is the minimum bandwidth requested by a user without dropping any data at the base-station. The maximum data droppage rate, denoted by d_i, is the maximum amount of data droppage rate acceptable to a user where $d_i \leq b_i$.

Now we formulate the optimal proportional bandwidth allocation problem for DiffServ with minimum bandwidth constraints as the following non-linear optimization problem:

$$\min \sum_{i=1}^{N} \sum_{j=i}^{N} \left(\left(\frac{1}{\omega_i} \right) \left(\frac{x_i \hat{c}_i}{b_i} \right) - \left(\frac{1}{\omega_j} \right) \left(\frac{x_j \hat{c}_j}{b_j} \right) \right)^2 \qquad (QP1)$$

subject to:

$$\frac{x_i \hat{c}_i}{T_w} \geq b_i \qquad i = 1, 2, ..., N \qquad (C1)$$

$$\sum_{i=1}^{N} x_i = M \qquad (C2)$$

$$x_i \geq 0 \qquad i = 1, 2, ..., M \qquad (C3)$$

$$x_i 's \text{ integers} \qquad i = 1, 2, ..., M \qquad (C4)$$

where ω_i is the DiffServ class weight of user i, x_i is the number of time-slots in the optimization window assigned to user i, b_i is the minimum desired throughput of user i during the optimization window, $\hat{c}_i$ is the estimate of the throughput of user i's channel per time-slot, T_w is the length of the optimization window in seconds, N is the total number of users which are being served by the base-station in the optimization window, M is the total number of time-slots in the optimization window. x_i's are the decision variables in the optimization problem $(QP1)$ and ω_i, b_i and $\hat{c}_i$ are constant parameters provided by the mobile users. An explanation of the constraints and objective function follows. Constraint $(C1)$ states that the allocated throughput of all the users is greater than or equal to the minimum throughput requested by the users during the optimization window. We assume that mobile users have sufficient buffers to store multimedia data if it arrives before its playout deadline. Constraint $(C2)$ means that the total number of time-slots allocated to all the users within an optimization window should be equal to the total number of time-slots in an optimization window. Constraint $(C3)$ is the non-negativity constraint on the decision variables. Constraint $(C4)$ is the integrality constraint for the decision variables. The DiffServ class ratio between user i and j is given by $\frac{\omega_i}{\omega_j}$. The goal of the Prop-DiffServ is to satisfy the following equalities:

$$\frac{\left(\dfrac{x_i \hat{c}_i}{b_i}\right)}{\left(\dfrac{x_j \hat{c}_j}{b_j}\right)} = \frac{\omega_i}{\omega_j} \qquad \forall i = 1, 2, ..., N; \; j = 1, 2, ..., N$$

that is to make the *proportional throughput ratio* for each pair of users equal to the DiffServ class ratio. However, because of the minimum bandwidth requirements of the users, satisfying all of the above equalities may be infeasible. In such a case, one way to keep the *proportional throughput ratio* close to the DiffServ class ratio is to minimize the error between the *proportional throughput ratio* and the DiffServ class ratio for all the distinct pair of users. The error between the *proportional throughput ratio* and the DiffServ class ratio for users i and j is given as below:

$$\frac{\left(\dfrac{x_i \hat{c}_i}{b_i}\right)}{\left(\dfrac{x_j \hat{c}_j}{b_j}\right)} - \frac{\omega_i}{\omega_j} \qquad \forall i = 1, 2, ..., N; \; j = 1, 2, ..., N$$

which is equivalent to,

$$\left(\frac{1}{\omega_i}\right)\left(\frac{x_i \hat{c}_i}{b_i}\right) - \left(\frac{1}{\omega_j}\right)\left(\frac{x_j \hat{c}_j}{b_j}\right) \qquad \forall i = 1, 2, ..., N; \; j = 1, 2, ..., N$$

Since the above error can be negative, we minimize the squared error instead. In addition, we want to minimize the squared error for all possible distinct pair of users; therefore, we minimize the sum of the squared errors where the summation is taken over all pairs of users. This gives the objective function of $(QP1)$ as follows:

$$\min \sum_{i=1}^{N} \sum_{j=i}^{N} \left(\left(\frac{1}{\omega_i}\right)\left(\frac{x_i \hat{c}_i}{b_i}\right) - \left(\frac{1}{\omega_j}\right)\left(\frac{x_j \hat{c}_j}{b_j}\right) \right)^2$$

The optimization problem $(QP1)$ belongs to the class of non-linear integer optimization problem, specifically quadratic programming, since the objective function is quadratic and the constraints are linear. The solution to $(QP1)$ brings the objective function closer to zero and hence results in bringing the *proportional throughput ratio* closer to the DiffServ class ratio for all pairs of users. In other words the solution to the above optimization problem finds the optimal proportional bandwidth allocation to keep the different service classes separated while ensuring minimum bandwidth constraints are satisfied. A relative DiffServ scheme keeps the service classes separated by the same relative distance even if the minimum bandwidth requirements of some users are being violated. The problem formulation in $(QP1)$ takes into account the minimum bandwidth requirements of all the users and satisfies these constraints whenever the system has sufficient resources.

Because of the fact that integer programming problems are NP-Complete [5] and the computational complexity of their solution techniques are not practical because of the real-time nature of our problem, therefore we relax the integrality constraint $(C4)$ in the optimization problem $(QP1)$. As a result of this relaxation, the amount of rounding error incurred is a fraction of a slot which is not significant for all practical reasons in the current context.

Due to the dynamic nature of throughput requirements of different mobile users and the channel capacity variations, the base-station may not be able to provide minimum bandwidth requirements to all users. In this case $(QP1)$ is infeasible. Under such a scenario, one option is to pre-fetch enough multimedia data at the mobile terminal to compensate for the slow transfer rate. However, slower transfer rate results in large playout delay which may not be practical. Under the constraints of fixed playout deadline at the mobile terminal, the only way to provide satisfactory services to users is to transmit the data partially and drop some of the data at the base-station. Such a scheme is especially useful for multimedia data in which users can tolerate some loss in data. For instance, if the resources in a wireless network are overly constrained, then the base-station can transmit the base-layer of a multi-layered encoded video, while dropping the enhancement layers. For concurrent multimedia sessions, determining the amount of data dropped for each user is equivalent to distributing penalty across all the users due to limited resources. Criteria for such decision can be based on the DiffServ class weights of the different users. The optimization problem $(QP2)$ provides a mechanism to drop data according to the DiffServ class weights of the users. It is possible that even after dropping maximum amount of data for each user, the network may not be able to provide service to all users. In this case $(QP2)$

becomes infeasible and it is not possible for the base-station to keep on serving all the users within the range of acceptable degradation of QoS. Hence the base-station has to stop serving some of the users, possibly lower service class users, in order to make sure that the higher service class users can be served.

B. Optimal Proportional Data Droppage for Differentiated Services

We formulate the optimization problem $(QP2)$ for an optimization window as follows. This problem is solved for each optimization window for which $(QP1)$ is infeasible.

$$\min \sum_{i=1}^{N} \sum_{j=i}^{N} \left(\omega_i \left(\frac{y_i}{b_i} \right) - \omega_j \left(\frac{y_j}{b_j} \right) \right)^2 \qquad (QP2)$$

subject to:

$$\frac{y_i}{T_w} \leq d_i \qquad i=1,2,...,N \qquad (C1)$$

$$\sum_{i=1}^{N} \left(\frac{b_i T_w - y_i}{\hat{c}_i} \right) = M \qquad (C2)$$

$$\left(\frac{b_i T_w - y_i}{\hat{c}_i} \right) \text{ integers} \quad i=1,2,...,N \qquad (C3)$$

$$y_i \geq 0 \qquad i=1,2,...,N \qquad (C4)$$

In $(QP2)$, y_i is the amount of user $i's$ data dropped in one optimization window, ω_i's are the DiffServ class weights. d_i is the maximum data droppage rate of user i, b_i is the minimum bandwidth requirement of user i without dropping any data at the base-station, $\hat{c}_i$ is the estimate of user $i's$ channel capacity per time-slot, T_w is the length of optimization window in seconds, M is the total number of slots in an optimization window and N is the total number of users served by the base-station during the optimization window. y_i's are the decision variables and ω_i, b_i, d_i, T_w, M and N are constant parameters. Constraint $(C1)$ of $(QP2)$ states that the droppage rate of each user is less than the maximum acceptable droppage rate to that user. Constraint $(C2)$ imposes that all slots in the optimization window should be used for transmitting data to the users. Constraint $(C3)$ is the integrality constraint, which can be dropped again under the same argument as presented in the context of $(QP1)$. Constraint $(C4)$ is the non-negativity constraint for the decision variables y_i's. Similar to Prop-DiffServ bandwidth allocator, the goal of the Prop-DiffServ data dropper is to satisfy the following equalities:

$$\frac{\left(\frac{y_j}{b_j} \right)}{\left(\frac{y_i}{b_i} \right)} = \frac{\omega_i}{\omega_j} \qquad \forall i=1,2,...,N; j=1,2,...,N$$

i.e., to make the *proportional data drop ratio* for each pair of users equal to the inverse of their DiffServ class ratio. However, because of the minimum data droppage requirements of the users, satisfying all of the above equalities may be infeasible. In such a case, one way to keep the *proportional data drop ratio* close to the DiffServ class ratio is to minimize the error between the *proportional data drop ratio* and the DiffServ class ratio for all pairs of users. The error between the *proportional data drop ratio* and the DiffServ class ratio for users i and j is given as below:

$$\frac{\left(\frac{y_j}{b_j} \right)}{\left(\frac{y_i}{b_i} \right)} - \frac{\omega_i}{\omega_j} \qquad \forall i=1,2,...,N; j=1,2,...,N$$

which is equivalent to,

$$\left(\omega_i \left(\frac{y_i}{b_i} \right) - \omega_j \left(\frac{y_j}{b_j} \right) \right) \qquad \forall i=1,2,...,N; j=1,2,...,N$$

Since the above error can be negative, we minimize the squared error instead. In addition, we want to minimize the squared error for all possible pairs of users; therefore, we minimize the sum of the squared errors where the summation is taken over all pairs of users. This gives the objective function of $(QP2)$ as follows:

$$\min \sum_{i=1}^{N} \sum_{j=i}^{N} \left(\omega_i \left(\frac{y_i}{b_i} \right) - \omega_j \left(\frac{y_j}{b_j} \right) \right)^2$$

Computational complexity of the above mentioned scheme is very important since the computations are performed by the base-station in real-time. The feasibility of a system of linear constraints can be determined in $O(N^3 L)$ time by using the interior-point linear programming algorithms [6], [7], where L represents the input bit complexity. Efficient implementation of interior-point algorithms are available [8]. The Quadratic programming problems, $(QP1)$ and $(QP2)$ can be solved efficiently in polynomial time if the Hessian of the objective function is positive-semidefinite using a variety of techniques [9] such as the method of Complementary Pivoting. However quadratic minimization problems with even a single negative eigenvalue for the Hessian are known to be NP-Hard. The Hessian of the objective functions of $(QP1)$ and $(QP2)$ are symmetric matrices whose leading principal minors have non-negative determinants, hence by Sylvester's criterion the Hessian of $(QP1)$ and $(QP2)$ are positive semidefinite (equivalently all the eigenvalues of Hessian are non-negative). Therefore, these optimization problems can be efficiently solved using polynomial time algorithms [9]. Most methods for solving quadratic programming problems, first derive the Karush-Kuhn-Tucker (KKT) conditions for the quadratic program and then solve the resulting system using the Simplex method [9].

IV. SIMULATION ENVIRONMENT

In this section we describe the wireless network simulation environment. We consider a grid of hexagonal cells consisting of a center cell. We implement the optimal resource allocation problem to allocate resources to users within the center cell. The cluster size is fixed to seven cells and the co-channel interference from the six first-tier co-channel interfering cells is taken into consideration while evaluating SINR. Base-station antennas are omni-directional and are located at the center of the cells. Time is slotted and the base-station transmits to one user in one time-slot. The length of one time-slot is taken to be 1ms. The length of the optimization window is taken to be 1 second. We assume that the mobile users in our simulation share the same frequency band. In an actual network multiple frequency bands are allocated to the same cell, however, our scheme can be used on each set of users which are sharing the same frequency band. The velocities of mobile users are independent uniformly distributed random variables between 10km/h to 100km/h. Each mobile user independently uniformly picks a location within the cell and starts moving towards the selected coordinates. During this period the velocity of the mobile is fixed. Once the mobile reaches the selected coordinates, it selects another location uniformly randomly within the cell and selects a speed uniformly randomly distributed between 10km/h to 100km/h and starts moving towards this new location. This process is repeated throughout the entire simulation run. We assume that the pathloss and shadow fading contribute towards the channel gains of individual users and assume that the effects of fast local fading due to multipath is averaged out. We use Lee's area-to-area model to predict path loss [10]. In Lee's path loss model the received signal power is expressed as follows:

$$\mu_{\Omega_p} = \mu_{\Omega_p}(d_o) \left(\frac{d_o}{d} \right)^\beta \left(\frac{f_c}{f} \right)^n \alpha_o$$

where $\mu_{\Omega_p}(d_o)$ is the power at 1mile point of interception, β is the path-loss exponent, d is the distance in kilometers, d_o is equal to 1.6km, f_c is 900 MHz and f is the transmission frequency. The parameter α_o is a correction factor used to account for different mobile station and base station antenna heights, transmit power, and antenna gains. The values of these and other parameters as used in our simulation is given in TABLE I.

The channel gain in dB is given as the sum of the path-loss term and the shadow fading term, as follows:

$$G_{dB} = PL_{dB} + s_{dB}$$

where PL_{dB} is the path-loss in dB and s_{dB} is the shadowing term in dB. We model the shadowing term as a zero mean stationary Gaussian process with autocorrelation function given by following [11]:

$$\phi_{s_{dB}}(k) = \sigma_s^2 \xi_D^{\left(\frac{vT}{D} \right)|k|} \qquad k \geq 0$$

where σ_s is the standard deviation of the shadow fading term, v is the velocity of the mobile, T is the envelope sampling interval and ξ_D is the shadow correlation between two points separated by a spatial distance of D meters. At 1700MHz,

Gudmundson has reported $\sigma_s = 4.3dB$, with spatial correlation ξ_D equal to 0.3 at a distance of 10m. In our simulation we use the same values [11].

The time step of the simulator is 1 ms equal to the length of one transmission time-slot. The optimization window consists of 1000 time-slots. We test our scheme under different loading conditions and compare the performance of our scheme with the weighted proportional fair scheduler (WPFS).

T_{opt} units before the start of next optimization window, each mobile terminal predicts the average channel capacity for the next optimization window using the previous SINR values and transmits this value to the base-station. We use a simple predictor which takes into account the value of the average channel SINR during the current optimization window and the change in the average SINR from the last optimization window to current optimization window to predict the average channel SINR during the next optimization window. The average SINR predictor used in our scheme is given as follows:

$$\hat{SINR}_{av}\left(k+1\right) = SINR_{av}(k) + \Delta SINR_{av}(k)$$

$$\Delta SINR_{av}(k) = SINR_{av}\left(k\right) - SINR_{av}\left(k-1\right)$$

where $\hat{SINR}_{av}\left(k+1\right)$ is the estimate of the average SINR during the $k+1th$ optimization window, $SINR_{av}(k)$ is the time-averaged SINR during the kth optimization window where the time average is taken over the SINR measured during each of the time-slots in the $T_w - T_{opt}$ units of the kth optimization window and $\Delta SINR_{av}(k)$ is the change in the time-averaged SINR measured during the kth and $\left(k-1\right)th$ optimization window. The reason we use the above mentioned prediction scheme is because, the average SINR does not change by large magnitude over the span of few optimization windows (few seconds). In particular, over a time-span of few optimization windows the average SINR increases linearly if the mobile is moving towards the base station and decreases linearly if the mobile is moving away from the base station. This is realistic to assume since in a practical mobile environments mobile users move in a certain direction with a certain speed over a span of few seconds and they don't change their direction and speed rapidly over very short time intervals. We must say that the performance of the optimization scheme is inherently tied to the quality of prediction of the channel capacity during the optimization window; however, in this paper we focus on the optimization problem and use the simple predictor as mentioned above. The simulation results show that even by using the simple SINR predictor we get very convincing results for the superior performance of our scheme. The SINR is mapped to throughput using the staircase function shown in Fig. 2. Once the time-slots are allocated to each user for an optimization window, the order of transmission is determined by transmitting to the user whose throughput prediction error is minimum for the current time-slot. We use this scheduling policy to minimize the channel capacity prediction errors and the deviation from our resource allocation solution.

V. SIMULATION RESULTS

A. Performance Metrics

The performance metrics for $\left(QP1\right)$ are the satisfiability of the minimum bandwidth requirement of all users, the ratio between the average throughputs of various classes, the standard deviation of the throughput of each class and the total network throughput. The ratio between average throughputs of various classes is a measure of the inter-class separation while the standard deviation of the throughput of a class is a measure of the intra-class fairness. For $\left(QP2\right)$ we use the following performance metrics; average data drop ratio of various classes, the standard deviation of data drop rate of various classes and the total network throughput. While evaluating $\left(QP2\right)$ we set d_i equal to b_i so that we don't have to worry about the infeasibility of $\left(QP2\right)$ during our simulations. In other words we assume that the users can tolerate a maximum data droppage rate equal to their minimum bandwidth requirement. We also test the scalability of both $\left(QP1\right)$ and $\left(QP2\right)$ by changing the number of users in the network from 10 to 20, 30, 40 and 50. We scale the throughput vs SINR mapping given in Fig. 2 by the same factor as the ratio of increase in the number of users.

We compare the performance of our scheme with the weighted proportional fair scheduler (WPFS) under various traffic scenarios. Each simulation is run for one hour, i.e., for each simulation run we solve the optimization problem for 3600 windows. The service class related parameters for $\left(QP1\right)$ and $\left(QP2\right)$ are given in TABLE II and TABLE III respectively. While comparing $\left(QP1\right)$ with WPFS, we set the weights of users for WPFS scheme equal to $\omega_i b_i$. When comparing $\left(QP2\right)$ with WPFS, we set the weight of user i equal to $\dfrac{\omega_i}{d_i}$. Although in [1] the authors have implemented a proportional loss rate dropper for hop-by-hop network, however, it can not be implemented at the base-station. The reason is that the proportional loss rate dropper makes a decision of dropping a packet by selecting the backlogged class j with minimum loss rate. The dropper

computes the loss rate for each class i as the fraction of class i packets recorded in loss history buffer (LHB) that were dropped. Such a loss rate dropper is suitable for hop-by-hop network and can be implemented in a router which has limited buffer capability, however it can not be implemented in the base-station since the base-station has enough buffer capacity to hold all the multimedia data. In other words the resource bottleneck is because of the wireless channel and not the buffering capacity of the base-station. In addition as noted in [1] the proportional loss rate dropper is not suitable for short time-scales and the deviations of their loss rate dropper from the proportional loss model increase as the time scale decreases. However, in our case we present the optimal data dropper for an optimization window which is a problem parameter and can be made as small as desired.

B. Simulation Results for (QP1)

Fig. 3 shows the throughput of all the users during the length of the simulation under $(QP1)$ and WPFS schemes when the total number of users $N = 10$. Under the WPFS scheme, all the users of class 1 are getting less than 60% of their desired bandwidth during most of the time, however, under $(QP1)$ scheme the class 1 users are served at their desired rate i.e., 250kpbs. $(QP1)$ satisfies the minimum bandwidth requirement of all the users and in addition keeps the different service classes separated. Fig. 4(a) shows that this desirable feature of $(QP1)$ persists as we scale the problem from 10 to 50 users. Fig. 4(b) and 4(c) plot the average throughput of class 2 and class 3 users respectively under $(QP1)$ and WPFS schemes. Fig. 5(a) plots the ratio of average throughput of class 2 users to the average throughput of class 1 users. Under no constraints, the minimum value of the objective function of $(QP1)$ is zero. If the minimum value of the objective function of $(QP1)$ is zero then the average throughput ratio of class 2 to class 1 is equal to 4. However, because of the minimum bandwidth constraints, the objective function can not attain this value in the feasible solution space. The WPFS scheme keeps the average throughput ratio of class 2 to class 1 around 4 without considering the minimum bandwidth constraints. $(QP1)$ adjusts this ratio around 2.5 while satisfying minimum bandwidth requirements of all the users.

Fig. 5(b) shows the average throughput ratio of class 3 users to class 1 users. If we ignore the constraints of $(QP1)$, then the minimum objective function value corresponds to a value of 12 for the average throughput ratio of class 3 to class 1. The WPFS scheme keeps this ratio close to 12. $(QP1)$ scheme however keeps this ratio around 5 while satisfying the minimum bandwidth constraints. Hence WFPS scheme does not keep into account the minimum bandwidth requirements of the various classes while setting the relative distance between various classes, however, $(QP1)$ finds the best possible separation between service classes, in the sense of minimizing the objective function of $(QP1)$ while satisfying the minimum bandwidth requirements of all the users.

Fig. 5(c) plots the average throughput ratio of class 3 to 2. WPFS scheme keeps this ratio close to 3, the DiffServ class ratio between class 3 to 2, however $(QP1)$ keeps this ratio close to 2.

C. Simulation Results for (QP2)

Fig. 6 shows the throughput of all the users under $(QP2)$ and WPFS. Fig. 6 plots the data drop rate of all the users under $(QP2)$ and WPFS respectively. The minimum value for the objective function of $(QP2)$ is zero. The objective function takes on this value when the proportional data droppage ratios for different users are equal to the DiffServ class ratio. i.e.,

$$\frac{\left(\dfrac{y_j}{d_j}\right)}{\left(\dfrac{y_i}{d_i}\right)} = \frac{\omega_i}{\omega_j} \qquad \forall\, i, j = 1, 2, 3, ..., N$$

$$\frac{y_j}{y_i} = \frac{\omega_i d_j}{\omega_j d_i} \qquad \forall\, i, j = 1, 2, 3, ..., N$$

For simulation of $(QP2)$, we fix the parameters as follows. $\omega_1 = 1$, $\omega_2 = 2$, $\omega_3 = 3$, $d_1 = 500\text{kbps}$, $d_2 = 1000\text{kbps}$ and $d_3 = 1500\text{kbps}$. These values imply that,

$$y_j = y_i \qquad \forall\, i, j = 1, 2, 3, ..., N$$

This is indeed the result we get from simulations. As Fig. 7 shows, under $(QP2)$ scheme all the users have the same data droppage rate. However, under WPFS, the data droppage rate as shown in Fig. 7 is different for different classes, with class 3

having the highest data drop rate and class 1 having the lowest data drop rate. The WPFS scheme behaves like this because of the way the weights are determined. Figs. 8(a), 8(b) and 8(c) show the average data drop rate of class 1, class 2 and class 3 users respectively.

As a consequence of the problem parameters, the average data drop ratios between different classes should be around 1. This is indeed what we get from the simulations. The plots in Figs. 9(a), 9(b) and 9(c) show that using $(QP2)$ scheme, the ratio between the average data drop rate between various classes is around 1. However, this is not true for WPFS.

In Figs. 10(a), 10(b) and 10(c) we plot the simulation results for standard deviation of the throughput of various classes under the $(QP2)$ and WPFS schemes. Fig. 10(a) shows that the standard deviation of class 1 throughput stays under 25kbps during most of the time of simulation run. This corresponds to a maximum standard deviation of 10% of the desired throughput of class 1 users, i.e., 250kbps. The plot in Fig. 10(b) shows that the standard deviation of the throughput of class 2 users under $(QP1)$ is at most 50kbps which is 10% of their desired throughput. Similarly, from Fig. 10(c) we see that the standard deviation of the average throughput of class 3 is below 10% of class 3's desired throughput for most of the time of the simulation run. The standard deviation plots show that the intra-class distance is bounded and is a small percentage of their desired throughput.

Figs. 11(a), 11(b) and 11(c) show the standard deviation of the data drop rate of various classes under $(QP2)$ and WPFS schemes. In Fig. 11(a) we see that the standard deviation of the data drop rate of class 1 users is below 30kbps which is 6% of their desired throughput. In Fig. 11(b) we notice that the standard deviation of the data drop rate of class 2 stays below 60kbps which is also 6% of the class 2's desired throughput. Finally, in Fig. 11(c) we see that the standard deviation of the data drop rate of class 3 is less than 100kbps during most of the time, which also corresponds to a value of around 6% of the desired throughput of class 3. The standard deviation plots show that the intra-class distance is bounded and is a small percentage of the desired throughput.
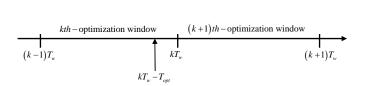
TABLE IV and TABLE V show a comparison of the total network throughput under our resource allocation schemes and WPFS. The values in the table show that the drop in the total network throughput by using $(QP1)$ and $(QP2)$ schemes is a small percentage of the total network throughput under the WPFS scheme.

VI. Conclusion

In this paper we have proposed an optimal proportional bandwidth allocator and an optimal proportional data dropper under minimum QoS constraints. Our scheme finds the best possible service class separation between users while satisfying the minimum QoS guarantees. Our scheme combines the desirable features of absolute DiffServ and the relative DiffServ approaches. We show the desirable features of our scheme by using extensive simulations.

REFERENCES

[1] C. Dovrolis and P. Ramanathan, "A case for relative differentiated services and the proportional differentiation model," *IEEE Network*, vol.13, no.5, pp.26-34, Sep./Oct. 1999.

[2] X. Zhou, C.-Z. Xu, "Harmonic proportional bandwidth allocation and scheduling for service differentiation on streaming servers," *IEEE Trans. Parallel and Distributed Systems*, vol. 15, no. 9, pp. 835-848, Sept.2004.

[3] K.-C. Wang and P. Ramanathan, "QoS assurance through class selection and proportional differentiation in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 3, pp. 573-584, Mar. 2005.

[4] G.-S. Ahn, A. T. Campbell, A. Veres, L.-H Sun, "Supporting service differentiation for real-time and best-effort traffic in stateless wireless ad hoc networks(SWAN)," *IEEE Trans. Mobile Computing*, vol. 1, no. 3, pp.192-207, July/Sept. 2002.

[5] C. H. Papadimitriou and K. Steiglitz, "*Combinatorial Optimization: Algorithms and Complexity*," Englewood Cliffs, N.J. : Prentice Hall, 1982.

[6] N. Karmarkar, "A new polynomial time algorithm for linear programming," *Combinatorica,* vol. 4, pp. 373-395, 1984.

[7] J. Renegar, "A polynomial-time algorithm based on Newton's method for linear programming," *Math Programming*, vol. 40, pp. 59-93, 1988.

[8] K. A. McShane, C. L. Monma, and D. Shanno, "An implementation of a primal-dual interior point method for linear programming," Rutcor res. rep., Rutgers Univ., New Brunswick, NJ, 1988.

[9] M. S. Bazaraa, H. D. Sherali and C. M. Shetty, *Nonlinear Programming Theory and Algorithms*, New York: Wiley, 1993.

[10] W. C. Y. Lee, *Mobile Communications Design Fundamentals*. Indianapolis, IN: Sams, 1986.

[11] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electronics Letters*, vol. 27, pp. 2145-2146, Nov. 1991.
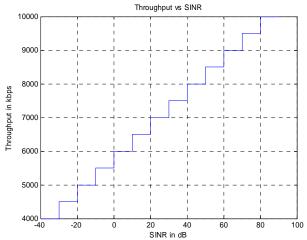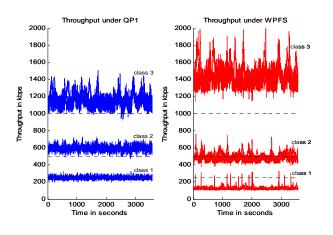
Fig. 1. Time-line of the optimization problem implementation.

Fig. 2. Throughput vs. SINR plot.
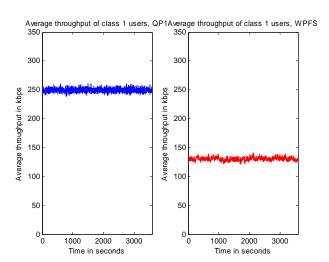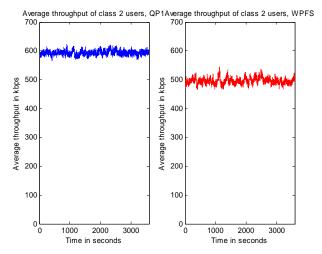
Fig. 3. Throughput of all users, $N = 10$ (a) QP1 (b) WPFS.

Fig. 4(a). Average throughput of class 1 users, $N = 50$ (i) QP1 (ii) WPFS.
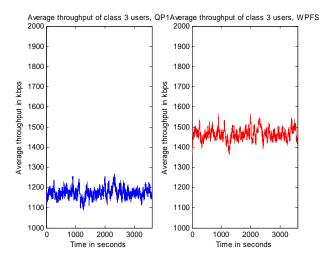
Fig. 4(b). Average throughput of class 2 users, $N = 50$ (i) QP1 (ii) WPFS.

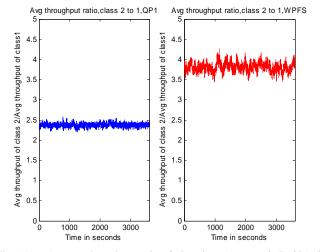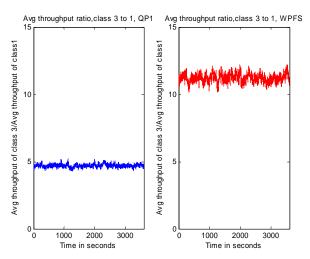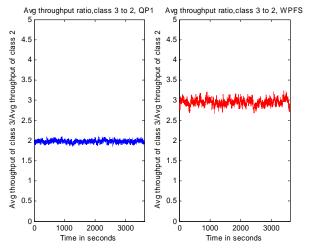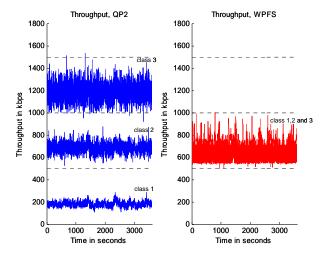Fig. 4(c). Average throughput of class 3 users, $N = 50$ (i) QP1 (ii) WPFS.

Fig. 5(a). Average throughput ratio of class 2 to 1, $N = 50$ (i) QP1 (ii) WPFS.

Fig. 5(b). Average throughput ratio of class 3 to 1, $N = 50$ (i) QP1 (ii) WPFS.
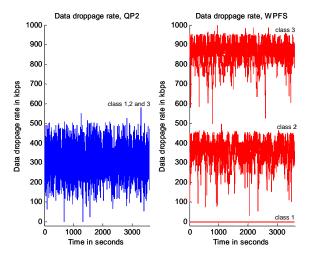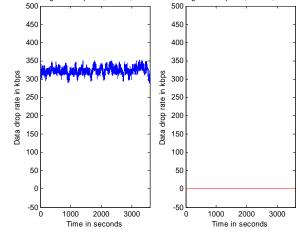
Fig. 5(c). Average throughput ratio of class 3 to 2, $N = 50$ (i) QP1 (ii) WPFS.

Fig. 6. Throughput of all users, $N = 10$ (i) QP2 (ii) WPFS.
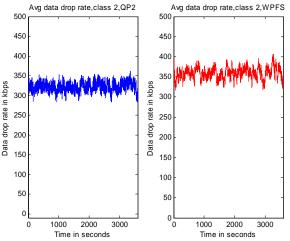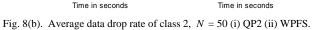
Fig. 7. Data drop rate of all users, $N = 10$ (i) QP2 (ii) WPFS.

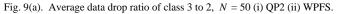Fig. 8(a). Average data drop rate of class 1, $N = 50$ (i) QP2 (ii) WPFS.

Fig. 8(b). Average data drop rate of class 2, $N = 50$ (i) QP2 (ii) WPFS.
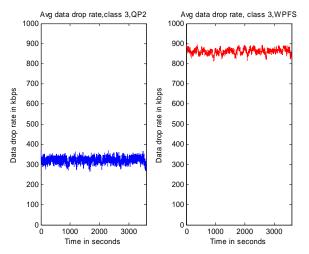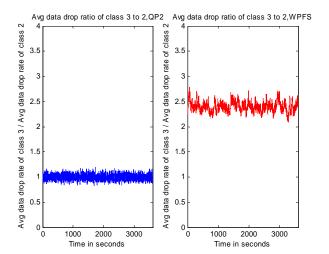
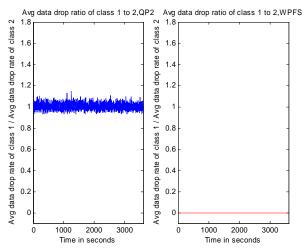Fig. 8(c). Average data drop rate of class 3, $N = 50$ (i) QP2 (ii) WPFS.

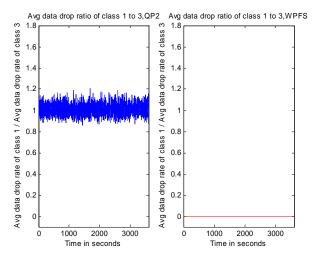Fig. 9(a). Average data drop ratio of class 3 to 2, $N = 50$ (i) QP2 (ii) WPFS.

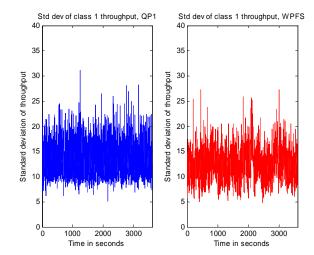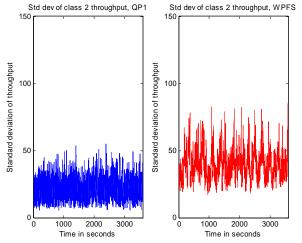Fig. 9(b). Average data drop ratio of class 1 to 2, $N = 50$ (i) QP2 (ii) WPFS.
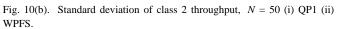
Fig. 9(c). Average data drop ratio of class 1 to 3, $N = 50$ (i) QP2 (ii) WPFS.

Fig. 10(a). Standard deviation of class 1 throughput, $N = 50$ (i) QP1 (ii) WPFS.

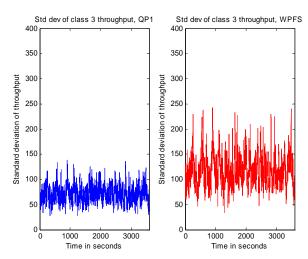Fig. 10(b). Standard deviation of class 2 throughput, $N = 50$ (i) QP1 (ii) WPFS.

Fig. 10(c). Standard deviation of class 3 throughput, $N = 50$ (i) QP1 (ii) WPFS.
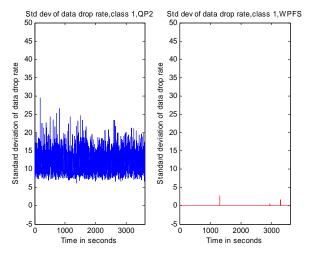
Fig. 11(a). Standard deviation of class 1 data drop rate, $N = 50$ (i) QP2 (ii) WPFS.
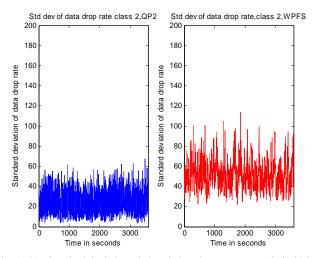

Fig. 11(b). Standard deviation of class 2 data drop rate, $N = 50$ (i) QP2 (ii) WPFS.
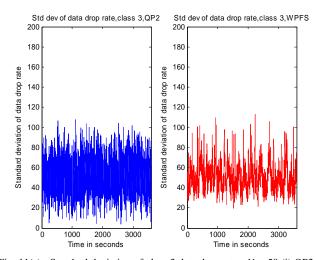

Fig. 11(c). Standard deviation of class 3 data drop rate, $N = 50$ (i) QP2 (ii) WPFS.

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Number of users (N)	10, 20, 30, 40, 50
Cell Radius	1000 m
Propagation Environment	North American Suburban
Path-loss exponent , β	3.84
Frequency	1.7 GHz
Standard deviation of shadowing term	4.3 dB
Base station antenna gain	20dBi
Mobile antenna gain	1.5dBi
Power at 1 mile interception, $\mu_{\Omega_p}(d_o)$	-61.7dB
Power Transmitted by base station	200 Watt
Shadowing Decorrelation Distance	10 m
Background Noise Power	40dBm
Maximum Speed	100km/h
Minimum Speed	10km/h
Length of Optimization Window(T_w)	1 second (1000 time-slots)
T_{opt}	10 ms (10 time-slots)
Simulation Time-Step	1 ms
Simulation Length	3600,000 Time-steps

TABLE II
SERVICE CLASS RELATED PARAMETERS FOR QP1

Mobile User	Minimum desired bandwidth (b_i) KBPS	DiffServ class weight (ω_i)
1	250	1
2	250	1
3	250	1
4	250	1
5	500	2
6	500	2
7	500	2
8	1000	3
9	1000	3
10	1000	3

TABLE III
SERVICE CLASS RELATED PARAMETERS FOR QP2

Mobile User	Minimum desired bandwidth (b_i) KBPS	Minimum data drop rate (d_i) kbps	DiffServ class weight (ω_i)
1	500	500	1
2	500	500	1
3	500	500	1
4	500	500	1
5	1000	1000	2
6	1000	1000	2
7	1000	1000	2
8	1500	1500	3
9	1500	1500	3
10	1500	1500	3

TABLE IV
TOTAL NETWORK THROUGHPUT: QP1 VS. WPFS

Number of users	WPFS (Mb)	QP1 (Mb)	% Decrease
10	22578.097	22376	1
20	45602.407	45019.995	1
30	69272	68197.419	1
40	92390.676	90983.246	1
50	115170	113370	3

TABLE V
TOTAL NETWORK THROUGHPUT: QP2 VS. WPFS

Number of users	WPFS (Mb)	QP2 (Mb)	% Decrease
10	23083.921	22716	1.59
20	46101.366	45393.106	1.53
30	69268	67950.672	1.9
40	92057.706	90205.928	2.01
50	115288.380	112834.465	2.12