Anticipatory Wireless Bitrate Control for Blocks

Xiaozheng Tie, Anand Seetharam, Arun Venkataramani, Deepak Ganesan,
Dennis L. Goeckel
University of Massachusetts Amherst

ABSTRACT

We present BlockRate, a wireless bitrate adaptation algorithm designed for *blocks*, or large contiguous units of transmitted data, as opposed to small packets. In contrast to state-of-the-art algorithms that can either have the amortization benefits of blocks or high responsiveness to underlying channel conditions of packets, BlockRate has both. Our evaluation shows that BlockRate achieves up to 2.8× goodput improvement in a variety of mobility scenarios.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication

General Terms

Design, Experimentation, Performance

Keywords

Wireless bitrate adaptation, block transmission

1. INTRODUCTION

Wireless bitrate control seeks to control the effective rate of transmission by adapting the amount of redundancy in transmitted data to the underlying channel quality so as to optimize the received goodput. In recent times a variety of bitrate control schemes have been introduced and experimentally-verified. They are based on channel quality metrics ranging from packet loss rate [6], packet transmission time [5], signal-to-noise ratio (SNR) [3], bit error rate and PHY-layer hints [9], etc.

Our work is motivated by the growing disparity between state-of-the-art bitrate control algorithms that are optimized to react on a per-packet basis and technology trends that suggest significant performance benefits to amortizing overhead across blocks, or large chunks of contiguous data transmitted as a single unit, consisting of many packets. For example, Li et al [4] demonstrate significant gains in reliable goodput using blocks by reducing the overhead of acknowledgments, timeouts, and backoffs at the link layer and redundant acknowledgments at the transport layer compared to per-packet TCP. Widely deployed 802.11n cards already enable large opportunities of uninterrupted transmission consisting of many packets. However, state-of-the-art bitrate control algorithms continue to be designed with per-packet adaptation in mind. If used as-is with blocks, these algorithms are prone to be unresponsive to changes in underlying channel quality as large blocks imply a commensurately large delay in obtaining channel feedback.

This disparity raises two natural research questions that form the focus of this paper. First, do the performance benefits of large blocks outweigh the performance loss due to the reduced responsiveness of bitrate control to changes in the underlying channel quality? Second, and more importantly, is it possible to have the performance benefits of blocks without compromising on the responsiveness of bitrate control?

Our measurement-based experiments with several static as well as mobile scenarios answer the first question in the affirmative. Our results show that traditional bitrate control algorithms designed to operate on a per-packet basis achieve moderately higher goodput when used as-is with blocks. This net performance benefit shows that there is greater value in amortizing overhead than reacting quickly to channel conditions. Furthermore, our results also show that there is room for improvement, i.e., an ideal block-based bitrate control scheme with future knowledge significantly outperforms packet-based schemes used as-is with blocks, thereby setting up the stage for the second question.

Our main contribution, the design and implementation of BlockRate, a block-based bitrate control algorithm, affirms the second question as well. The key insight in BlockRate is to use multiple bitrates across packets within a block that are predictive of future channel conditions. BlockRate maintains a receiver-assisted technique to maintain a mapping between the SNR and the best bitrate corresponding to that SNR. BlockRate uses this history-trained SNR-to-bitrate mapping in conjunction with its model for predicting the future SNR to pick (possibly different) bitrates for packets in the next block.

BlockRate uses two simple models to predict the SNR of packets in the near future. The first model is applicable to slow-changing scenarios such as static or pedestrian mobility scenarios. In such slow-changing scenarios, BlockRate employs a *linear regression model* based on a historic time series of SNR values to predict the future SNR. The second model is applicable to fast-changing scenarios as is typical under vehicular mobility. In such fast-changing scenarios, BlockRate employs a *path loss model* in conjunction with the historic time series of SNR values and its knowledge of the distance to the receiver to predict the future SNR.

We implemented a prototype of BlockRate in the MadWiFi driver [2] and deployed it on an indoor mesh testbed and an outdoor vehicular testbed. Our evaluation shows that, compared to existing packet-level schemes used as-is with blocks, BlockRate achieves up to 2.8× goodput improvement in a variety of mobility scenarios.

S3'11, September 19, 2011, Las Vegas, Nevada, USA.

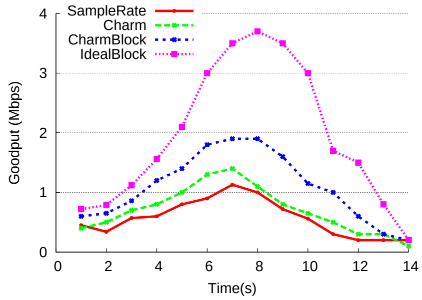


Figure 1: Goodput under vehicular mobility. IdealBlock has up to 2× goodput improvement over CharmBlock.

2. MOTIVATION

In this section, we experimentally motivate the need for block-based bitrate control. Our measurement-based experiments reveal two findings. First, the performance benefits of amortizing overhead with blocks outweigh the performance loss due to less responsive bitrate control. Second, an ideal block-based protocol with future knowledge can significantly improve upon this gain compared to using existing packet-based bitrate control schemes as-is with blocks.

We conduct a simple experiment in an outdoor vehicular setting to compare the effectiveness of bitrate control using blocks and packets. This experiment involves a laptop placed in a car driven by one of the authors sending back-to-back UDP packets to another stationary laptop acting as an "access point". The packet size is 1.5KB and each block contains 200 packets.

We compare the performance of two block-based bitrate control algorithms, IdealBlock and CharmBlock, against two packet-based ones, SampleRate [5] and Charm [3]. IdealBlock is obtained using a tedious a priori measurement process where one of the authors experimented with all possible bitrates at points spaced 5 meters apart from the access point. It is intended to serve as a lower bound on the performance achieved by an optimal bitrate control scheme. CharmBlock uses the Charm bitrate control algorithm[3] as-is in conjunction with blocks. Charm and SampleRate are both well-known packet-level bitrate control algorithms that have been shown to work well in 802.11a/b/g networks; the former adapts the bitrate based on the SNR while the latter adapts the bitrate based on the packet transmission time.

Figure 1 shows the goodput achieved by the compared bitrate control algorithms as a function of time. All schemes show a period of increasing goodput followed by decreasing goodput, which is consistent with the vehicle first moving towards the access point and then away from it. The experiment yields two important insights. First, the block-based schemes, IdealBlock and CharmBlock, significantly outperform the packet-based schemes. Second, there is a significant difference (of up to 2×) in the goodput achieved by IdealBlock compared to CharmBlock, i.e., a well-designed block-based bitrate control scheme significantly outperforms a packet-based scheme used as-is with blocks, making the case for a bitrate control scheme optimized for blocks.

3. BlockRate DESIGN

In this section, we present BlockRate, a block-based bitrate control algorithm that achieves high goodput in a variety of static and mobile settings. The key insight is to use multiple bitrates across packets within a block that are selected based on the predicted SNR trend over the transmission of the block.

BlockRate uses the SNR measured at the receiver to learn for each SNR regime and bitrate the corresponding packet loss rate. The receiver learns this mapping, referred to as the *SNR-bitrate table*, by monitoring the loss rates of packets sent at various attempted bitrates by the sender in the recent past. The receiver uses this table to select the best bitrate for each packet in the next block based on the predicted SNR for the packet, and conveys this information to the sender piggybacked with a selective acknowledgment for packets within the current block.

The high-level design described above is similar to existing SNR-based bitrate control schemes [3], but with an important difference in the prediction step. Packet-based bitrate control schemes simply assume that channel conditions do not change significantly in the course of a packet or two, so the SNR predicted for the next packet is the same as that of the current packet. However, this simplistic prediction performs poorly at the block granularity during which the channel condition may vary a lot.

To address this problem, BlockRate uses two simple models to predict the SNR for packets within the next block. The first is a *linear regression model* invoked in slow-changing environments such as in static or pedestrian mobility scenarios. The second is a *path loss model* invoked in fast-changing scenarios such as under vehicular mobility.

3.1 Linear regression model

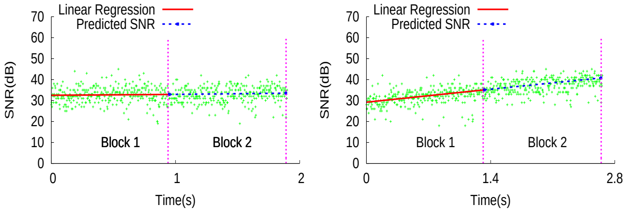
In static or pedestrian mobility scenarios, the SNR trend changes slowly. To appreciate this, consider Figures 2 showing the variation in packet SNRs from one block to the other in an indoor setting, for the static and the pedestrian mobility scenarios respectively. To measure the SNR, we use two laptops configured in a sender-receiver mode. In the first experiment (Figure 2(a)), both laptops are kept static, while in the second (Figure 2(b)) one of them is kept stationary with the other being moved towards it at walking speeds.

Figure 2(a) shows that in the static case, the SNR almost remains constant; with most fluctuations confined to a small range of 10 dB. Figure 2(b) shows that in the walking case, the small fluctuations persist, but are accompanied by a weak upward trend as the receiver is moved towards the sender. Depending on the bitrates selected, each block can take between 0.2 and 1.5 seconds to be transmitted. Thus, if a pedestrian user's motion (speed and acceleration) remain unaltered for the next second or two, the SNR trend can be expected to continue across consecutive block transmissions. We find that this is indeed the case throughout our entire set of pedestrian mobility traces (not shown for brevity), not just the specific illustrative example shown in Figure 2(b).

In such slow-changing scenarios, a simple linear regression model [1] can effectively capture SNR trends across consecutive blocks. The model assumes that the received SNR has a simple linear relationship with time, i.e., the model fits a straight line through the set of SNR samples using the least squares estimate. BlockRate extends this regression line to predict the SNR variation over the course of the next block.

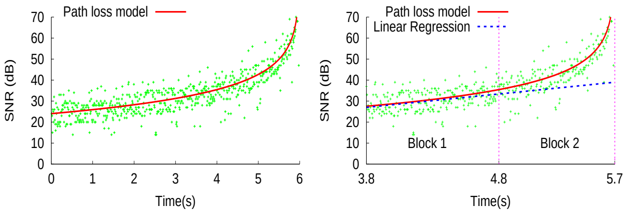
3.2 Path loss model

In outdoor vehicular mobility scenarios, SNR variations can be rather steep within the course of a block, as the vehicle's distance from the access point can change significantly during that time. For example, Figure 3(a) shows the time series of packet SNRs observed by a receiver kept in a car



(a) Static (b) Pedestrian

Figure 2: SNR variation: static and pedestrian mobility.



(a) Vehicular (b) Vehicular (zoomed)

Figure 3: SNR variation: vehicular mobility.

approaching a sender (AP) at 30 mph. We conducted this experiment in a downtown area using two laptops with the sender laptop placed at a fixed location near the edge of the road. The figure shows that the SNR changes rapidly within several seconds and, more importantly, does not show a simple linear relationship with time (especially in the 3–6 second regime).

Figure 3(b) further zooms into the packet SNRs within two consecutive blocks in this regime. The figure shows that the linear regression line deviates considerably from the actual SNR, so any estimation based on linear extrapolation will likely result in suboptimal performance. More importantly, the SNR shows this sharp, non-linear trend when the vehicle is closest to the AP and has a good channel (SNR $>$ 50 dB). We show in §4 that the highest goodput is achieved during this time period, so it is critical to capture these sudden changes in SNR.

To address this problem, BlockRate adopts the wireless channel path loss model [7] to model the non-linear variation of SNR over time. The path loss model postulates a logarithmic relationship between path loss and distance:

$$PL(d) = PL(d_0) + 10\alpha \log_{10}(d/d_0) \qquad (1)$$

where PL is the path loss in dB, PL = transmit power - receive power, d_0 is the reference distance and α is the path loss exponent. Since on a dB scale, SNR = transmit power - PL - noise, and as the transmit power at the sender and the noise at the receiver are usually fixed, we can rewrite Eq. 1 as:

$$\text{SNR}(d) = \text{SNR}(d_0) - 10\alpha \log_{10}(d/d_0) \qquad (2)$$

i.e., SNR increases logarithmically with decreasing distance. Figure 3 shows that the path loss model indeed fits the SNR variations under vehicular mobility well. To use the path loss model for SNR prediction, BlockRate employs the readily available GPS devices in vehicles to estimate the distance d between the sender and receiver. A detailed description of the prediction mechanism is deferred to [8].

4. EVALUATION

We implemented a prototype of BlockRate in the MadWiFi driver [2] and deployed it on an indoor mesh testbed and an outdoor vehicular testbed. Figure 4 shows the goodput over time for different bitrate control algorithms in the same outdoor vehicle-to-AP setting as that in §2. BlockRate has a peak goodput of 3.1Mbps at the 8^{th} second, which is 1.6× better than CharmBlock and 2.8× better than SampleRate and Charm. Note that BlockRate achieves the highest gains between the 5^{th} and 11^{th} seconds when it is in close proximity to the AP. This observation is consistent with the results in §3.2 showing that the SNR changes sharply when the vehicle is near the AP. The median goodput gain is 1.3× over CharmBlock and 2× over SampleRate and Charm.

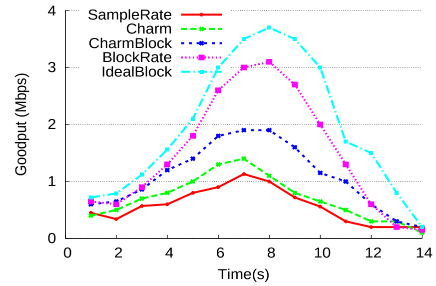


Figure 4: Goodput in vehicle-to-AP experiment. BlockRate achieves maximum goodput gains of 1.6× over CharmBlock and 2.8× over SampleRate and Charm.

Figure 4 also shows that there is still room for improvement compared to IdealBlock, e.g., the peak goodput of IdealBlock is 3.7Mbps, which is 20% higher than BlockRate. This difference is due to the less accurate channel SNR estimation by the path loss model and the GPS devices, and we are currently working on solving this problem.

5. ONGOING WORK

We are working on improving BlockRate in the following aspects. First, the two prediction models in §3 are applicable to different mobility scenarios, i.e., pedestrian and vehicular mobility respectively. How to dynamically switch between them according to the mobility pattern is part of our ongoing efforts. Second, the evaluation presented so far only considers the performance of BlockRate against the unreliable goodput. Since most practical applications (e.g., FTP and VOIP) require reliable transfer, we are also evaluating BlockRate under the reliable transport protocol TCP.

6. REFERENCES

[1] Linear regression. http://en.wikipedia.org/wiki/Linear_regression.

[2] Madwifi. http://madwifi-project.org.

[3] G. Judd, X. Wang, and P. Steenkiste. Efficient channel-aware rate adaptation in dynamic environments. MobiSys, 2008.

[4] M. Li, D. Agrawal, D. Ganesan, and A. Venkataramani. Block-switched networks: a new paradigm for wireless transport. In *NSDI*, 2009.

[5] R. T. Morris, J. C. Bicket, and J. C. Bicket. Bit-rate selection in wireless networks. Technical report, MasterÕs thesis, MIT, 2005.

[6] I. Pefkianakis, S. Hu, S. H. Wong, H. Yang, and S. Lu. Mimo rate adaptation in 802.11n wireless networks. MobiCom, 2010.

[7] T. Rappaport. *Wireless Communications: Principles and Practice.* 2nd edition, 2001.

[8] X. Tie, A. Seetharam, A. Venkataramani, D. Ganesan, and D. L. Goeckel. Anticipatory wireless bitrate control for blocks. Technical Report, UMASS, 2011.

[9] M. Vutukuru, H. Balakrishnan, and K. Jamieson. Cross-Layer Wireless Bit Rate Adaptation. In *ACM SIGCOMM*, 2009.