The Introduction of Soft Information in IP-based Wireless Networks

Anna Brunstrom
Karlstad University
651 88 Karlstad, Sweden
Anna.Brunstrom@kau.se

Tony Ottosson
Chalmers University of Technology
412 96 Göteborg, Sweden
Tony.Ottosson@s2.chalmers.se

ABSTRACT

The underlying physical link is transparent in most IP-based networks. Contrary to this commonly accepted design rule, we propose that the applications should be made aware of the channel conditions. This is especially fruitful for wireless links where the performance is many orders of magnitudes lower than in fixed networks. Instead of wasting resources to make the wireless link behave as a fixed link, the application could take care of the adaptation to the channel condition. The presented solution assumes that soft information consisting of a reliability measure of the received bits is produced in the physical layer. This soft information is then propagated to the application. The application may use this information to distinguish between errors caused by fading and network congestion. Another possible use for soft information is to make the applications adapt the source and channel codes to the current channel condition and thus maximize performance.

INTRODUCTION

The current trend in telecommunications and data communications is towards integration of systems and services. Most networks will be based on the IP protocol, also wireless networks. It is for example believed that the third generation wireless systems will use IP-based core networks and that in a not to distant future all traffic will be IP-based from end-user to end-user. This for instance means that voice will be supported using Voice over IP (VoIP).

The biggest problem with this development towards transparent, general purpose IP networks is that different networks perform very differently. For instance is a fiber-based network almost free of transmission errors whereas transmission errors can be common over wireless networks due to fading and interference. The traditional solution to this problem is to make the wireless network transparent with respect to varying channel conditions. However, this solution has several drawbacks.

From the higher layers it is difficult if not impossible to distinguish between packet loss due to congestion and packet loss due to a fading channel. A problem that is well-known to severely hamper the performance of TCP over wireless networks. Making the wireless network transparent also means that much of the bandwidth need to be spent on channel coding and retransmissions on the link level. A solution that reduces the throughput and increases the delay. This is especially problematic if real-time services such as voice communication are to be supported.

In this paper we propose a fundamentally different solution to the problem described above. We introduce the concept of soft information. Rather than making the channel transparent the idea behind soft information is to make the end-points aware of the channel condition. This allows higher layers and applications to adjust appropriately. After an overview of related work below, application areas for the concept are discussed. How the soft information can be represented and propagated in a structured fashion is considered next. A discussion around implementation issues is also provided.

RELATED WORK

Other proposals for structured information exchange between the application and the network can be found in the literature. Most of the work in the IP community has been focused on aspects related to providing more flexible quality-of-service support over the Internet. Enhancing the QoS support involves extending the Internet's service model, from the single class of best-effort service, to include a wider variety of service classes. Prominent proposals in this area are the Intserv [21, 17] and DiffServ [15, 7] architectures investigated within the IETF.

Key components of the Integrated Services architecture include the packet classifier, the packet scheduler, the admission control routine and the reservation setup protocol. The packet classifier determines the QoS class for each incoming packet and the packet scheduler orders packet transmission to achieve the promised QoS for each flow. Admission control implements the decision algorithm that a router or host uses to determine whether a new flow can be granted the requested QoS without impacting earlier guarantees. The reservation setup protocol is necessary to create and maintain flow-specific state in the end-point hosts, and in routers along the path of a flow. RSVP [8] is the protocol of choice in the implementation framework. RSVP delivers resource reservation requests to the relevant network elements, but plays no other role in providing network services and is thus primarily a vehicle used by applications to communicate their requirements to the network.

RSVP and the Integrated Services approach have been criticized for lack of scalability due to the need of setting up and maintaining state for each flow. The Differentiated Services architecture [15, 7] is more light weight. Rather than providing a per flow service a number of traffic classes are supported. All packets that belong to a given traffic class or traffic aggregate receive a particular forwarding treatment, or per-hop behavior (PHB), at each network node. A small bit-pattern in each packet is used to mark a packet to receive a particular forwarding treatment. In order to support a given service, traffic entering a network is conditioned at the boundaries of the network, but no admission control is performed within the network.

Other models for service differentiation also exist [1]. The common characteristic of all the models is that information primarily flows from the application into the network. The application can either choose from a fixed set of traffic classes when sending a packet or negotiate the service on a per connection or flow basis. The characteristics of and the dynamic state of the underlying network are still transparent from the application. In contrast to the various models on service differentiation, we argue for the structured information exchange from the network, and in particular from the wireless segment, to the application. Allowing information to flow in this direction will allow applications to fully utilize the information that has been passed over the wireless link and to adjust dynamically to varying channel conditions. The two approaches are complementary and the concept of soft information could be applied over a network that supports service differentiation.

Proposals for a general and structured information exchange of network conditions to higher layers are more rare in the literature. The mechanism for explicit congestion notification (ECN) defined in [16] is, however, one example. ECN provides routers in the network with a mechanism to inform transport end nodes that congestion is forming without requiring packets to be dropped. To signal congestion the router sets a congestion experienced (CE) bit in the IP-header. ECN is designed for signaling network congestion whereas the concept of soft information is intended to signal information about wireless channel conditions.

Some proposals for special purpose information exchange about wireless link conditions have also appeared. Many of the suggested optimizations for TCP over wireless channels involve some form of interaction between the link layer and the transport layer. The snoop protocol [5] is one example of a TCP-aware link layer mechanism. The idea behind snoop is to cache unacknowledged TCP data at the base station and to perform retransmissions locally over the wireless link when possible. Link layer retransmissions are triggered by duplicate acknowledgements from the TCP receiver. These duplicate acknowledgements are removed from the TCP data stream to prevent retransmissions by the TCP sender. In [3] and [4] problems over the wireless link are monitored at the base station and reported to the TCP sender using different forms of explicit notifications. Information about mobile hand-offs can also be passed to the transport layer and used to improve TCP performance [9, 19]. The examples presented above serve as motivation for the need of passing information upwards in the communication stack. Our work can be seen as a generalization of this information exchange for wireless channels. As will be discussed in the next section one application of soft information could be to enhance the performance of TCP.

APPLICATIONS

A wide range of applications for soft information can be envisioned. In its simplest form the channel state information could consist of a single bit per packet. This information could be used by for instance TCP. TCP treats packet loss as an implicit notification of congestion in the network and reduces its transmission rate in response. This is correct in most wired networks but is the wrong counter-action in a wireless network, where errors caused by severe fading channel conditions dominate. Typically, packet loss due to fading is a temporary condition and reducing the transmission rate will only result in a very inefficient use of the wireless channel. Including simple channel state information for the TCP packets would allow the end-points to determine whether packet loss is due to congestion or fading and to take action accordingly.

When beneficial, more complete channel state information could be generated in the receiver and passed to the application. Some applications such as voice, audio and image transmission are rather insensitive to a small amount of errors and can utilize partially correct information. An image transmission application might for instance be satisfied with one soft value for each small block of an image, and use this soft information to decide whether or not to ask for a retransmission of that block. Alternatively, the image transmission application could use the soft information to detect errors in a block and then use error concealment techniques [2] to compensate for the data loss.

For applications that can tolerate some errors (e.g. speech, audio, image and video applications) it is a waste of resources to make the network transparent. Retransmissions increase the delay and the bandwidth is in most cases better spent on increasing the source quality using higher rate source encoders. Moreover, if the application receives channel state information, the application could adapt to the channel so that it introduces extra redundancy if the channel is bad at the cost of decreased source quality. Once the channel gets better the redundancy is decreased and the data rate is used to increase the source quality. The goal is of course to find the optimum trade-off between source coding and channel coding to achieve a good representation of the source and low enough error rate on the channel. Since the wireless channel is time-varying this trade-off needs to be adapted, and hence the application needs to be aware of the channel condition.

The extreme end of what has been discussed above is to let the wireless networks transmit the received data uncoded and force the applications to adapt to the channel using the channel state information. The networks would be very simple and most complexity would be in the applications. Even if this may sound bad, it actually increases the flexibility of the systems since applications are software and software can be upgraded or replaced rather easily while network standards and hardware are more difficult to update or replace. Software defined radio, where the baseband processing is moved from hardware into DSP-software to allow for reconfiguration, may be thought of as the first step in this direction. The extreme case described above would be to move much of this DSP-software (the source and channel encoding/decoding parts) up to the applications.

CHANNEL STATE INFORMATION

In the most extreme case the applications should be able to perform all source and channel coding. Thus we need to be able to give the application all the information needed to perform the decoding with good results. We propose that the maximum a-posteriori (MAP) estimates of the received bits are generated and forwarded to the application since these estimates are transparent to modulation type and also embed the channel estimates. MAP estimates of course depend on the structure of the transmitted information, that is the format in which the application generate the bits. A knowledge that is generally unavailable to the receiver. Instead we propose that the MAP estimates are generated assuming that the bits are independent and equally likely (full entropy assumption). This is probably a very mild assumption since the receiving application has knowledge of the transmitted structure and can thus adjust the MAP estimates accordingly. In fact, there are indications that this approach, at least in some cases, is optimum [18]. However, a thorough investigation is needed to more exactly quantify how the MAP estimates should be calculated and what if any performance loss results from the assumption of independent random bits.

Another issue is how many bits are needed to represent a MAP estimate. A reasonable guess is 3 bits for every bit. However, this means that each packet received in the wireless network is forwarded as a three times as large packet. This may not be feasible or desirable in all situations. Hence, we need a flexible format so that we can adjust the amount of soft information generated. It may also be that some applications do not have use for soft information for every bit. Although powerful enough to provide complete information, the format should also work for our simple TCP example described earlier.

To meet the requirements stated above, we propose a soft information format based on MAP estimates of a block of L bits assuming that the bits in the block are independent and equally likely. The resolution of the soft information should be n bits per block. Considering our examples above, in the most extreme case L would be 1 and n would be 3 (or even higher) and in the simplest case L would equal the number of bits in a packet and n would be 1.

IMPLEMENTATION ISSUES

In this section we briefly consider some implementation issues for the concept of soft information. We will consider three communication scenarios of varying complexity.

In the first scenario soft information is used only in the mobile station (MS). This scenario is typical of an asymmetric data transfer where the bulk of the information is transfered downlink. In this scenario the information is generated in the physical layer in the terminal and passed up through the layers to the application. This is the simplest case since there is no need to transfer the soft information across the network. However, if the application cannot deal with soft information it should be possible to just remove it.

In the second scenario we consider communication between an application on an MS and an application in the fixed network. We assume that bandwidth is not a problem in the fixed network. In this scenario soft information must be generated both in the terminal and at the edge of the wireless network. In the MS the soft information is passed up through the layers as in scenario 1, but in the base station the soft information need to be forwarded to the host on the fixed network. How the information should be forwarded will depend heavily on the amount of information generated. For small amounts of soft information it may be possible to use an IPv6 extension header for destination options [10]. If the receiver can not handle the option, then it would simply ignore it. Large amounts of soft information are more problematic and may have to be forwarded in separate packets.

In the third scenario we consider communication from a MS via a fixed network to another MS in the wireless network. This is for example the setting for mobile telephone systems. Here soft information generated over one wireless link would need to be forwarded over the other wireless link to reach the end node. However, in this case bandwidth is a big problem and we therefore recommend that this connection is made through a proxy server. An application layer proxy could decode the soft information and regenerate the data before transmission over the wireless network. If information about the application is not available at the proxy, then the proxy could strip the soft information prior to transmission.

A proxy solution could also be an alternative for the second scenario discussed above, especially when a large amount of soft information is generated. There are also cases where the application always would prefer a proxy solution. Having a separate connection over the wireless link allows special optimizations to be performed. A performance enhancing application level proxy can provide functionality such as caching, data compression or address translation. Several proxy systems have been developed, both commercially [11, 12] and by the research community [14, 13, 6]. A proxy solution placed close to the wireless network also has the advantage that fast adaptation to the channel would be possible [20].

In the discussion above, we have considered where the soft information should be generated and propagated for three different communication scenarios. The amount of soft information that should be generated will vary depending on higher layer requirements. Mechanisms for negotiating the setting of L and n between application end-points and for signalling these settings to the physical layer must thus also be defined. The details regarding these mechanisms are left for future work.

CONCLUDING REMARKS

In this paper we have introduced the concept of soft information. The idea behind soft information is to provide structured information about the channel conditions to the communication end-points. This allows the applications to utilize knowledge about and even adjust to the current channel condition. Providing soft information to higher layers in the communication stack could open up a market for designing intelligent wireless-aware software. Products could for instance compete based on their ability to adapt the source and channel codes to the current channel condition in an optimal way.

REFERENCES

[1] ATM Forum. Atm traffic management specification version 4.0, April 1996.

[2] L. Atzori and F. De Natale. A novel method for the recovery of lost visual data using correlated edges information. *Proceedings Packet Video 99*, New York, April 1999.

[3] B. Bakshi, P. Krishna, N. H. Vaidya, and D. K. Pradhan. Improving performance of TCP over wireless networks. *Proceedings 17th Int. Conf. Distributed Computing Systems*, Baltimore, USA, May 1997.

[4] H. Balakrishnan and R. Katz. Explicit loss notification and wireless web performance. *Proceedings Globecom Internet Mini-Conference*, Sydney, Australia, November 1998.

[5] H. Balakrishnan, S. Seshan, and R. Katz. Improving reliable transport and handoff performance in cellular wireless networks. *ACM Wireless Networks*, 1(4), December 1995.

[6] H. Bharadvaj, A. Joshi, and S. Auephanwiriyakul. An active transcoding proxy to support mobile web access. *Proceedings of the 17th IEEE Symposium on Reliable Distributed Systems*, 1998.

[7] D. Black, S. Blake, M. Carlson, E. Davies, Z. Wang, and W. Weiss. RFC 2475: An architecture for differentiated services, December 1998.

[8] R. Braden, Ed., L. Zhang, S. Berson, S. Herzog, and S. Jamin. RFC 2205: Resource ReSerVation Protocol (RSVP) — version 1 functional specification, September 1997.

[9] R. Cáceres and L. Iftode. Improving the performance of reliable transport protocols in mobile computing environments. *IEEE Journal on Selected Areas in Communications*, 13, June 1995.

[10] S. Deering and R. Hinden. RFC 2460: Internet Protocol, Version 6 (IPv6) specification, December 1998.

[11] Ericsson AB. Webonair software product information. *http://mobileinternet.ericsson.se/*.

[12] R. Floyd, B. Housel, and C. Tait. Mobile web access using eNetwork Web Express. *IEEE Personal Communications*, 5(5):47–52, October 1998.

[13] A. Fox and E. A. Brewer. Reducing WWW latency and bandwidth requirements by real-time distillation. *Proceedings Fifth International World Wide Web Conference*, Paris, France, May 1996.

[14] M. Kojo, K. Raatikainen, and T. Alanko. Connecting mobile workstations to the Internet over a digital cellular telephone network. *Proceedings Workshop on Mobile and Wireless Information Systems (Mobidata)*, Rutgers University, NJ, November 1994.

[15] K. Nichols, S. Blake, F. Baker, and D. Black. RFC 2474: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 headers, December 1998.

[16] K. Ramakrishnan and S. Floyd. RFC 2481: A proposal to add Explicit Congestion Notification (ECN) to IP, January 1999.

[17] S. Shenker, C. Partridge, and R. Guerin. RFC 2212: Specification of guaranteed quality of service, September 1997.

[18] M. Skoglund and T. Ottosson. Soft multiuser decoding for vector quantization over a CDMA channel. *IEEE Transactions on Communications*, 46(3), March 1998.

[19] M. Stangel and V. Bharghavan. Improving TCP performance in mobile computing environments. *Proceedings International Conference on Communications '98*, Atlanta, GA, June 1998.

[20] M. Sternard, A. Svensson, A. Ahlén, and T. Ottosson. PCC wireless IP - optimizing throughput and QoS over fading channels. *Proceedings Nordic Radios Symposium (NRS01)*, April 2001.

[21] J. Wroclawski. RFC 2211: Specification of the controlled-load network element service, September 1997.