End-to-End Evaluation of WWW and File Transfer Performance for UMTS-TDD

Andrea Baiocchi and Francesco Vacirca
INFOCOM Dept. - University of Roma *La Sapienza*
Via Eudossiana 18, 00184, Roma - Italy

Abstract— **Wireless data is deemed as a major booster of next generation wireless networks. In this context, the support of Internet applications based on the TCP still presents several open issues. The aim of this work is to evaluate the performance of TCP data transfer over the UMTS TD-CDMA air interface, by means of a rather detailed simulation model. The evaluation includes both bulk FTP like data transfer and interactive client-server (WWW) traffic. The distinctive point of view is exploring the interaction between application and transport level protocol functions and lower layers protocols over the radio interface. The interplay of radio access and fixed core network impairments is analyzed as well. Major results deal with balancing the error recovery performance of the RLC against the radio channel correlation and the fixed network impairments.**

I. Introduction

Wireless data is one of the major booster of wireless communications and one of the main motivation of next generation standards [1]. The Time Division Duplex (TDD) mode of UMTS, based on a TD-CDMA, has been designed to address specifically the needs of data traffic. An in-depth assessment of TCP/IP performance over this interface is still to be carried out. In particular, in this work we highlight how bulk and client server interactive traffic compare as to their ability to efficiently exploit the UMTS TDD air interface. A second related issue explored in this work is the interaction of impairments (delay and loss of IP packets) due to the UTRAN and those that are introduced by the fixed network.

Wireless links may affect TCP essentially because of non congestion packet losses and of relatively large delays that limit the achievable throughput and may also affect fairness. On the positive side, correlated errors that are typical of terrestrial mobile networks are not entirely detrimental, since they bring about better TCP performance, as the average error ratio remains constant [2][3]. These results change a somewhat if lower layers error recovery functions are provided (e.g. FEC, ARQ). In that case we shall see that TCP throughput is a decreasing function of the correlation degree of the radio interface errors.

A general description of the problems and solutions of TCP over wireless networks are reported in [4][5][6][7][8] and some specific solutions are given in [9][10][11].

The specific contribution of this work is to integrate rather complete models form the radio channel and the fixed network impairments at packet level to the application level traffic generation, to verify the interactions and critical points in the interactions of TCP (end-to-end view), RLC (ARQ over the air interface) and MAC (capacity sharing and scheduling over the air interface). The major results are criteria to identify RLC parameters, as a function of the radio channel memory and of the fixed network performance degradation. The rest of the paper defines the simulation model in detail (Section II), then the scenarios simulated in the experiments and the results of simulations are discussed in Section III. Conclusions are drawn in Section IV.

II. Simulation Model

This Section reviews the main features of the simulation model that has been implemented by using the `ns` software. The addressed scenario comprises a UMTS radio cell covered by a Node-B connected to an RNC. The air interface is modeled including the radio channel and up to the RLC layer. The UMTS core network and the external IP network to the remote servers is modeled as an impairment inducing black box (from TCP connections point of view), i.e. by specifying a model for the IP packet delays and losses.

A. Physical Layer

Figure 1 depicts a synthetic layout of the TD-CDMA technique used in the UMTS-TDD interface. The time axis is slotted and 15 time slots are grouped into a 10 ms frame. Within each time slot a set of variable spreading factor orthogonal codes are defined to spread QPSK modulated data symbols of up to 16 different users.

The overall capacity of the radio interface is shared by common channels and data channels. The first time slot of every frame is assigned to the Random Access CHannel (RACH); the downlink common signaling channels (BCH, PCH, SCH and FACH) are in the last slot of the frame. The remaining slots are assigned (dynamically) to the uplink or downlink as need be. Each requesting UE (or the RNC) can get a share of the available data capacity as a given number of RUs for some time. How much and how long bandwidth is assigned is a matter of MAC layer algorithms. As for the radio channel model, we consider the classic model including deterministic path loss, correlated log-normal shadowing and correlated Rayleigh fast fading (Jakes' model). We account for open loop power control and assume that it essentially recovers signal fluctuations due to the deterministic path loss and to shadowing. By including both the effect of non ideal channel estimate and intra and inter cell

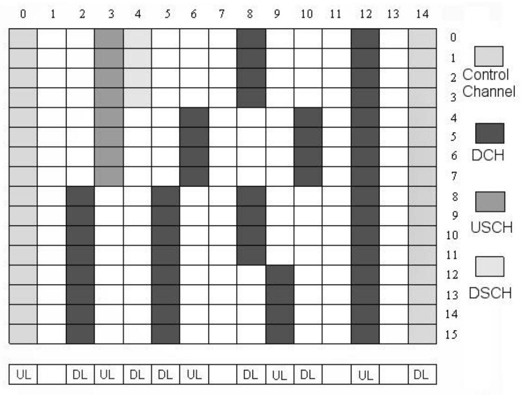


Fig. 1. CDMA-TD layout of one frame.

interference, the SIR of the i-th communication in the tagged cell can be expressed as

$$SIR_i = \frac{e_i^2 \cdot SIR_{TARGET_i} \cdot M_i}{1 + \alpha \cdot \sum_{\substack{j=1 \\ j \neq i}}^{N} e_j^2 \cdot SIR_{TARGET_j} \cdot M_j \cdot \frac{1}{SF_j}} \quad (1)$$

where α is a coefficient accounting for the interference reduction benefit of the joint detection, M is the power margin introduced by the open loop power control [12], e^2 is the fast fading path loss estimation error, SF is the spreading factor and N is the number of ongoing communications in the tagged cell. In the following, we assume that the error e^2 is log-normally distributed. Its logarithm is a correlated Gaussian process, with mean 0, variance σ^2 and negative exponential correlation. We denote by ρ the rate of decay of the autocorrelation of $\log e^2$ over a UMTS frame (10 ms).

The raw BER can be evaluated from (1) by assuming a Gaussian interference model. The BER after decoding (e.g. convolutional coding) can be evaluated (at least an approximation of) by means of well established techniques and by assuming an ideal interleaving and hence independence among bit error events. As a result, the probability that a codeword is found in error at the receiver can be evaluated as a function of the FEC parameters and of the SIR (1). The codeword corresponds to a transport block. In this work we assume that transport blocks are 336 bit long, (including MAC/RLC overhead), TTI=10 ms and convolutional coding with rate 1/2 is used. This implies that four RUs are required to carry the coded transport block (along with the coded block of the associated control channel at 3.4 kbit/s). From a modeling point of view, the entire transport resource of the radio interface (i.e. 240 RUs) is divided into common control channels and resource available for user data multiplexing. Each slot can carry up to four data plus associated signaling transport blocks in each frame, if we require that the four RUs used by a transport block be all in the same time slot.

B. MAC Layer

For bulk data transfer, the MAC layer assigns a properly sized DCH to any requesting UE or to the RNC, out of the capacity pool of unassigned RUs, with the constraint that uplink/downlink attribute applies to an entire slot. The size of a channel is measured by the number of transport blocks it carries, hence it is an integer multiple of 4 RUs (see Figure 1). The assignment is limited to sending up to a maximum number N of RLC-SDU (i.e. IP packets). N is a critical parameter to trade efficiency (high values of N) against fairness (low values of N). The effect on TCP data transfer performance of varying N is studied in [13]. Since DCHs are used, no header is needed in the MAC-PDU [14].

The assumed MAC capacity allocation scheme uses a best-effort and extremely simple approach. It collects all requests of new MAC connection setup coming from the UEs (or from the RNC) and tries to accommodate them in the radio interface spare capacity, by using a rough weighting factor, to account for three quantized levels of backlog in the UE/RNC buffers.

New capacity requests from the UEs for transferring UL RLC-SDU are sent via the RACH and relevant assignments are signaled by means of the FACH. If an UE does not get any capacity, it will try again according to back-off rules aimed at preventing RACH congestion. Capacity requests coming from the RNC for transferring DL RLC-SDUs are handled just the same way, except that no explicit signalling is required on the air.

Once a DCH with given capacity has been allocated to a newly opened MAC connection for transferring up to N RLC-SDUs, an RLC connection is established over it, according to the acknowledged mode of operation. The MAC connection release is implicitly signaled by leaving the relevant capacity unused for one frame.

C. RLC Layer

The main functions of the RLC protocol are Segmenting and Reassembly of RLC-SDUs into the fixed size RLC-PDU, error recovery for loss sensitive data and flow control [15].

A RLC-PDU is carried by a MAC-PDU into a transport block, whose format is described in Section II-A. The RLC-PDU length is 336 bit. Since 2 bytes are required for the default RLC header in the acknowledged mode of operation [15], there are 40 bytes of payload per RLC-PDU. Therefore the RLC layer net throughput of an RLC connection exploiting a transport block set of size n per TTI=10 ms is bounded above by $4000 \cdot n$ byte/s.

Erroneous RLC-PDUs are recovered by means of a Selective Repeat kind of protocol [15]. Hence there is a need for a feedback channel to carry acknowledgements of data RLC-PDUs. Since a DCH is requested by a backlogged entity (either UE or RNC) for transferring a number of RLC-SDUs in one direction, a small channel would be required in the opposite direction just for carrying RLC acknowledgements.

In [13] it is shown that a significant performance gain can be achieved if two common channels are defined for the purpose of carrying RLC acks: one for the DL (shared by means of simple TDM) and one for the UL, shared by means of a collision type protocol. These common RLC ack channels are shared by all active RLC connections (see Figure 1).

To sum up, whenever an RLC entity has a backlog, it requests capacity on the radio interface. If successful, a unidirectional MAC connection exploiting a suitably sized DCH is established and over this a bi-directional RLC connection can be set up, where the reverse channel for RLC acks exploits the common ack channel. If the RLC connection gets suspended, the MAC connection (hence the DCH) is released. At resume time, a new MAC connection is established over a new DCH, until either the backlog is exhausted or the maximum number of N RLC-SDU has been successfully transferred.

III. Simulation experiments

TCP Reno is used in the simulation. We define two different reference scenarios: i) ideal fixed network, i.e. the fixed network does not lose or corrupt any IP packet nor does it makes any bottleneck to the TCP connections, with respect to the wireless access (Figure 3(a)); ii) non-ideal fixed network, i.e. packets of each TCP connection are delayed and randomly discarded with an assigned probability. The first scenario is used to analyze TCP performances over UTRAN by taking into account only the effects of radio link. In fact, as we can see in Figure 3(a), all wired links have been overprovisioned and no losses occur. The second scenario aims at simulating an actual UMTS core and external IP network. As depicted in Figure 3(b), we added a packet loss generation module in the link between the gateway (GW) and the base station (BS) and a variable Internet-like packet delay generation module in the dedicated links between each wired node and the GW. The IP packet delays are generated according to the model proposed in [16].

We consider two application level traffic patterns: i) greedy FTP sources, that lead to persistent TCP connections; ii) WWW interactive sources, that generate non-persistent TCP connections.

In the FTP scenario the client is placed in the wired nodes, in the WWW scenario the client is on mobiles. Thus, whereas in the first case the greater part of the traffic is in UL, in the latter one the most part of the traffic is in DL. The number of sources in FTP simulations is 16, whereas in HTTP ones is 32.

WWW traffic is generated according to the model described in [17]: the client sends a fixed length request message to the server and when the server receives it, it sends back the file to the client (see Figure 2). The length L of the file in bytes is Pareto distributed, with mean 40 kbytes. The interarrival time T_i of requests is exponentially distributed. Only a single TCP connection need to be opened for each file sent upon a request. The file represents the whole amount of information of the objects forming the requested web page; a single TCP connection per page is consistent with the modeling of HTTP v1.1 [18].

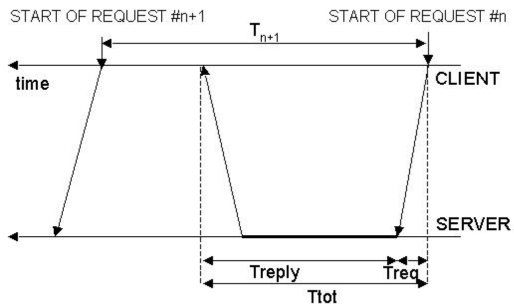


Fig. 2. HTTP client-server interaction.

In the simulations presented in the following subsections, some parameters always assume the values shown in Table I:

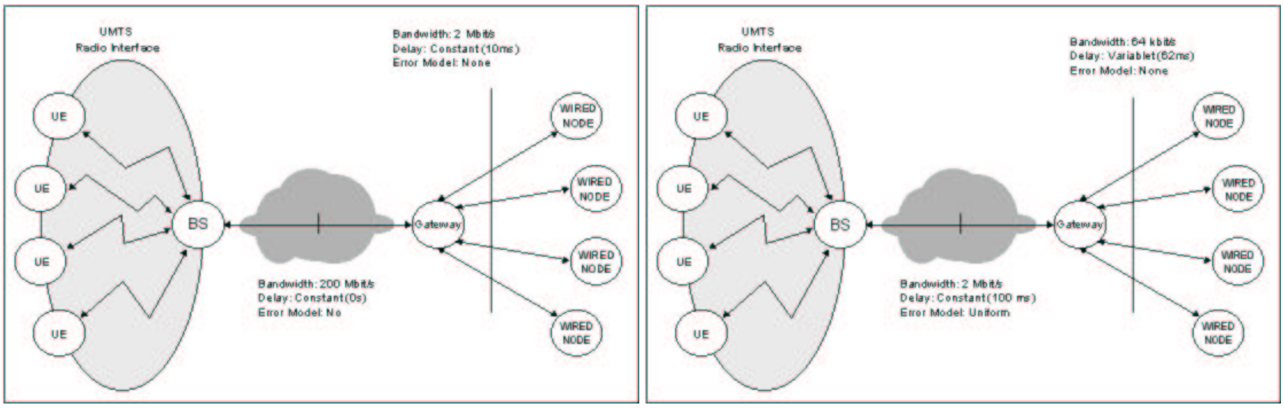
TABLE I
SIMULATION PARAMETERS

SIR_{TARGET}	6.8 dB
M - Open loop power control margin	3 dB
α - Mutual interference reduction factor due to joint detection	0.1
TCP Packet Size	1024 byte
N - Max number of RLC-SDU that can be trasmitted for an assignment of radio capacity	20
σ^2 - Radio channel estimation variance	10 dB

A. Ideal Wired Network

Initially we consider the case of FTP application sources. Figure 4 shows the trend of the TCP throughput obtained by varying the correlation coefficient ρ. The behavior of the TCP throughput is analyzed by considering three different RLC working cases: i) no RLC level retransmissions (dotted line); ii) limited number ($max\_retr = 16$) of RLC level retransmissions (dashed line); iii) unlimited RLC level retransmissions (solid line).

As already pointed out by [2] [3], when no lower layer retransmission mechanisms are used, TCP throughput increases as ρ increases. Conversely, when ARQ mechanisms are introduced, TCP throughput degrades for high values of ρ. This is due to different effects: first of all, when ρ is very large RLC with a limited number of retransmissions is unable to totally hide to TCP the wireless channel impairments. In this case, the residual loss probability increases with ρ and hence also the packet loss experienced by TCP (TCP timeouts grow from less than 1 for a thousand TCP segments to about 10% of TCP segments, as ρ increases from 0.1 to 0.9). However, the TCP performance degrades even if the number of RLC retransmissions is unlimited. In this case, the residual packet loss rate is zero, but the average time needed to carry a TCP segment over the air interface increases by about 37% with respect to the case



(a) Ideal fixed network.

(b) Non ideal fixed network.

Fig. 3. Simulation topologies.

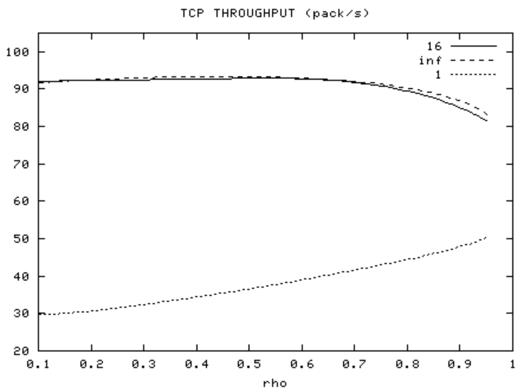


Fig. 4. TCP Throughput vs ρ for different values of max_retr.

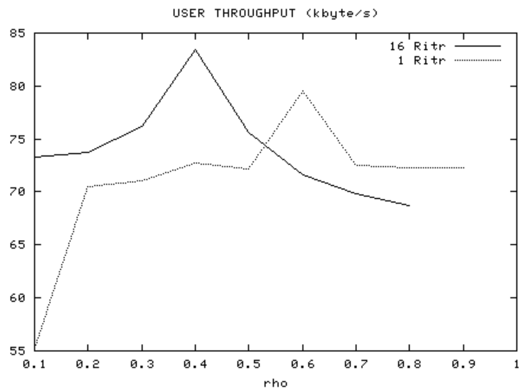


Fig. 5. TCP Throughput vs ρ with WWW traffic.

with $max\_retr = 16$. Moreover, the larger ρ the higher the variability of the TCP segment delay through the air interface. Therefore, even if no TCP segment is lost over the radio interface, the TCP throughput falls slightly since the average and the variance of the TCP connection RTT is larger than in the case with finite max_retr.

In Figure 5 TCP throughput in the case of WWW traffic is shown as a function of ρ. We use 32 active HTTP clients with the average interarrival time T_i set to 21.5 s (see Figure 2) and the mean downloaded file size L_{mean}=40 kbyte. In this case ρ influences TCP behavior in quite a different manner with respect to the case of FTP sources. Up to values of about ρ=0.4, TCP throughput improves rapidly, but after this threshold TCP performance experiences a quick degradation, stronger with respect to the case of FTP sources. Moreover, there is a degradation of TCP throughput also when no retransmissions are made by RLC. This is in opposition to what is shown in earlier works, where models of persistent TCP sources were considered and also to the dotted curve shown in Figure 4.

B. Real Wired Network

In the second simulation scenario we introduce losses and delays in the wired network segment and test it with greedy traffic sources. In our simulations the mean value of delays introduced by the single delay module on each dedicated link is set to 62 ms, the maximum value is 1.5 s and the minimum one is 35 ms. The bandwidth of the link between BS and GW is 2 Mbit/s and the bandwidth between the GW and the FTP clients is 64 kbit/s (see Figure 3(b)).

In Figure 6, TCP throughput is represented as a function of the wired loss rate of the loss module in the link between GW and BS. We consider two cases: i) ideal wireless channel, i.e. no packet loss on wireless link, (line with crosses); ii) real wireless channel, i.e. wireless link with a non-zero RLC packet loss probability (line with diamonds). For the sake of comparison, two straight lines representing the upper limit values of the throughput, obtained when in the two above described cases no packet loss is experienced in the wired segment of the network, are depicted.

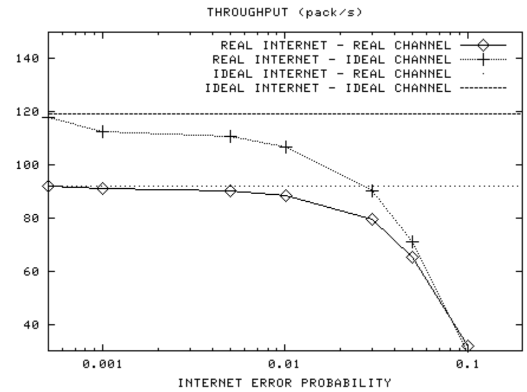


Fig. 6. TCP Throughput as a function of wired error probability.

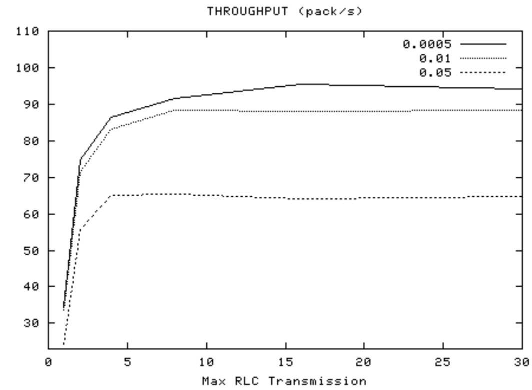


Fig. 7. TCP Throughput vs max_retr for different values of the wired error probability.

In the case of wired network without losses the gap of TCP throughput between ideal channel and real channel is about 23%. By increasing the wired error probability, this gap reduces until it cuts out when the packet loss ratio of the wired network is about the same as the wireless error ratio (0.05). The prevalence of wired network non-idealities cancels the effects of the wireless network segment. This behavior could be exploited in order to design RLC and RRC layers to be adaptive to wired network avoiding useless retransmissions and FEC that increase power dissipation and decrease channel capacity (especially in a pure CDMA system like W-CDMA where mutually interference is also detrimental to the overall network capacity).

In Figure 7 the TCP throughput is plotted as function of the maximum number of RLC retransmissions. Each curve represents a different value of wired loss rate. It is interesting to note that when the loss probability in the wired network is high (0.05 dashed line) it is useless to design RLC layer with a large maximum number of retransmission because the performance of TCP reach the maximum limit, already for small values of the maximum number of retransmissions.

IV. Final Remarks

We proposed an `ns` [19] based simulation analysis of bulk and interactive application traffic carried by TCP connections over the UMTS-TDD radio interface and UMTS core network. We found a different behavior of persistent TCP connection and non-persistent TCP connection with bursty errors. Criteria are given for the dimensioning of the number of retransmission of the RLC ARQ so as to balance the impairments introduced by the radio interface and those inherent in the fixed network.

Further work is needed to investigate the interaction between MAC layer algorithms and upper layer traffic patterns and TCP congestion control. Another issue is adaptive channel scheduling and error protection to exploit at best the radio channel correlated error pattern.

References

[1] Chaudhury P., Mohr W.: *The 3GPP Proposal for IMT-2000*, IEEE Communications Magazine, December 1999, pp. 72-81.
[2] M. Zorzi, A. Chockalingam, R. R. Rao, *Throughput Analysis of TCP on Channels with Memory*, IEEE Journal on Selected Areas in Communications, Vol.18, No. 7, July 2000.
[3] E. Altman, K. Avrachenkov, C. Barakat, *TCP in Presence ofBursty Losses*, Proc. of ACM Sigmetrics 2000, Santa Clara, California (USA), June 17-21 2000.
[4] M. Allman, Editor, *Ongoing TCP Research Related to Satellites*, RFC2760, IETF, February 2000.
[5] C. Barakat, E. Altman, W. Dabbous, *On TCP Performance in a Heterogeneous Network: A Survey*, IEEECommunications Magazine, January 2000.
[6] G. Xylomenous et al., *TCP Performance Issues over Wireless Links*, IEEE Communications Magazine, April 2001.
[7] Chiasserini, C.F.; Meo, M , *Improving TCP over wireless through adaptive link layer setting*, Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE , Volume: 3 , 2001 Page(s): 1766 1770
[8] F. Anjum and L. Tassiulas, *On the Behavior of Different TCP Algorithms over a Wireless Channel with Correlated Packet Losses*, Proc. of ACM SIGMETRICS'99, Atlanta, USA, May 1-4, 1999.
[9] R. Ludwig, R.H. Katz, *The Eifel Algorithm: Making TCP Robust Against Spurious Retransmissions*, ACM Computer Communications Review, Vol. 30, No. 1, January 2000.
[10] I. F. Akyildiz, G. Morabito, S. Palazzo, *TCP-Peach: A New Congestion Control Scheme for Satellite IP Networks*, IEEE/ACM Transactions on Networking, Vol. 9, No. 3, June 2001.
[11] C. Casetti et al., *TCP with Faster Recovery*, Proc. of IEEE MILCOM'2000, Los Angeles,CA, USA, September 2000.
[12] 3GPP TS 25.331 V3.9.0 (2001-12) *Radio Resource Control (RRC) Protocol Specification*, Technical Specification Group Radio Access Network, December 2001.
[13] A. Baiocchi, F. Cuomo: *Efficiency and fairness trade-off for TCP over UMTS-TDD*, IEEE ICC 2001, Helsinki, 11-15 June 2001.
[14] 3G TS 25.321: *MAC Protocol Specification*, 3GPP Tech. Spec. Group RAN WG2, March 2000.
[15] 3G TS 25.322: *RLC Protocol Specification*, 3GPP Tech. Spec. Group RAN WG1, March 2000.
[16] A. De Vendictis, A. Baiocchi: *Wavelet Based Synthetic Generation of Internet Packet Delays*, 17th International Teletraffic Congress, Salvador de Bahia, Brasile, 2-7 December 2001.
[17] A.Reyes, E.Gonzlez, E.Casilari, J.C. Casasola and A. Daz, *A page-oriented WWW traffic model for wireless system simulations*, Proc. ITC 14 pp1271-1280, 1999.
[18] Fielding R.T et alii, *Hypertext Tranfer Protocol HTTP/1.1*, Internet draft, IETF, 1996.
[19] *Network Simulator* `ns`, UC Berkeley, LBL, USC/ISI, Xerox PARC. Internet Site: http://www.isi.edu/nsnam/ns