

Capacity Measurement in Communication Networks

Leopoldo Angrisani¹, Aniello Napolitano², Michele Vadursi³

¹ *Dipartimento di Informatica e Sistemistica* ² *Dipartimento di Ingegneria Elettrica,*
Università degli Studi di Napoli Federico II, Via Claudio, 21 - 80125 Napoli (Italy)
Ph.: +390817683170 - Fax: +390817683816 - Email: {angrisan; aniello.napolitano}@unina.it

³ *Dipartimento per le Tecnologie, Università degli Studi di Napoli “Parthenope”*
Via Medina, 40 - 80133 Napoli (Italy)
Ph.: +390815474741 - Fax: +390815474777 - Email: vadursi@uniparthenope.it

Abstract- The capacity of the communication network is a key metric for both network planning and management, and performance assessment of applications that take advantage of the network. Several tools for capacity measurement are present in literature, which provide satisfying results when used in wired networks. On the contrary, they provide inaccurate results on wireless networks. The paper presents a refined study of processing times at each protocol layer, which suggests that capacity estimates provided by the existing tools are affected by a systematic error. A suitable measurement methodology and a related testbed are proposed to experimentally validate the theoretical results. The experimental outcomes demonstrate the validity of the study, and permit to explain why existing tools are inaccurate on wireless links.

I. Introduction

In computer science, capacity of a link is defined as the maximum transmission rate of the link. With special regard to a path, constituted by subsequent links, its capacity is defined as the minimum link capacity in the path [1]. Capacity assessment is therefore of extreme importance for both network planning and management. Moreover, capacity significantly affects communication quality in terms of maximum achievable throughput and end-users' satisfaction. To properly fix the performance that distributed and interactive applications can achieve in a wired and/or wireless network scenario, network capacity should be assessed accurately.

Several tools, based on different techniques, are present in the literature for measuring physical capacity [2]-[4]. They were designed for use on wired networks, where they actually provide satisfying results [5]. On the contrary, their outcomes become unreliable on wireless links. This is the reason why new tools have specifically been developed for capacity measurement in wireless networks during last years [6]-[8]. However, their performance are still not satisfying, as proved by the authors in [9]. Experimental results reported in [9] have also shown that such limitations cannot be simply imputed to possible in-channel interference.

The reasons for such an unexpected behaviour of both general purpose and wireless specific capacity measurement tools have not been thoroughly understood yet. Capacity measurements are basically achieved through one way delay (OWD), round trip time (RTT) and interarrival measurements, according to a specific model. It will be shown in this paper that this model neglects some time contributions during transmission and reception phases. Depending on the entity of the those time contributions, their neglect can be acceptable on wired links (i.e. it does not affect capacity measurement results significantly), but leads to erroneous results on wireless links. To overcome this limitation, a new model, based on a refined study of processing times at each protocol layer, is introduced in the paper, which is consistent with experimental outcomes that are commonly experienced on wireless networks [9]. Moreover, it specializes the concept of capacity for each layer of the protocol stack. A new testing methodology and an *ad-hoc* measurement testbed are also introduced to experimentally validate the results in both wired and wireless scenarios.

II. Capacity measurement

In literature there are three major techniques for capacity measurement: *Variable Packet Size probing* (VPS), *Packet Pair* (PP) and *Packet Train* (PT) [1].

The VPS technique permits to estimate the capacity of each hop along a path. For each hop, several probing packets of the same size L are sent from the source host by properly setting the TTL field in the IP header. On the arrival at the destination hop, an ICMP message will be generated and sent back

to the host, which will then measure the RTT. It is assumed that at least one of the probing packets, together with its ICMP message, do not experience any queuing delay; the related RTT is thus the minimum RTT for the given packet size. With reference to a single hop, the minimum RTT, namely $RTT(L)$, is given by:

$$RTT(L) = \alpha + \frac{L}{C_0} = \alpha + \beta L \quad (1)$$

where C_0 is the nominal capacity of the hop and α stands for the delay independent from the probing packet size L . The operations are repeated for different values of L . By estimating the slope of minimum $RTT(L)$ curve, β , and taking its reciprocal, an estimate of C_0 can be achieved.

PP and PT techniques are utilized to measure capacity when UDP is adopted as transport protocol. In particular, PP technique sends a pair of packets of the same size, L , to the receiver. The receiver measures the dispersion Δ of the packet pair, that is the time interval between the reception of the last bit of the two packets. If a link of nominal capacity C_0 connects the sending host to the receiver, Δ is equal to L/C_0 . In general, if Δ_{in} is the dispersion before the link of capacity C_0 , then the dispersion after the link will be $\Delta_{out} = \max(\Delta_{in}, L/C_0)$, assuming that there is no other traffic (*cross traffic*), on that link. Cross traffic can increase or decrease the dispersion, causing underestimate or overestimate of the hop capacity.

PT probing extends packet pair probing by using N back-to-back packets of size L and measuring the dispersion of a packet train as the time interval between the last bit of the first and last packets. The receiver measures the total dispersion $\Delta_R(N)$ of the packet train and calculates a dispersion rate as

$$D = \frac{(N-1)L}{\Delta_R(N)} \quad (2)$$

If there is no cross traffic along the path, the dispersion rate is equal to the path capacity.

III. Proposed Model

The performance of capacity measurement tools based on the techniques described in the previous section are not satisfying when applied to wireless links [9].

All the techniques obtain an estimate of the capacity by processing the results of time measurements according to a simple delay/capacity model. The question is whether this model is correct, i.e. representative enough. Since it works properly on wired links, the idea is to investigate whether it neglects some time contributions that are negligible for wired links, but not for wireless ones. To do that, a new capacity model is proposed. A comparison of the proposed model with those underlying VPS and PP techniques gives the reasons for the poor performance of existing tools on wireless links. Thanks to the correct and complete definition of all the temporal contributions that each layer adds when sending one packet from a source to a destination, the proposed model is suitable for capacity measurements on both wired and wireless links, as experimental results prove (see section IV).

Moreover, the model allows extending the capacity concept to the each layer of the protocol stack, intended as the maximum quantity of information transferred at the layer. For example, the *transport capacity* is the maximum rate at which the application data (that is, the transport layer payload) can be transmitted to the receiver's application layer. The knowledge of transport capacity is more useful than physical capacity, because the overheads introduced by each layer can reduce *effective* capacity significantly. Note that when there is no cross traffic over the communication channel, the transport capacity coincides with the application throughput.

The idea of extending the capacity concept at different protocol layers is not totally new. For example, in [1] capacity is defined as the maximum available IP layer transfer rate, i.e. the capacity measured at the IP layer. Moreover, the work presented in [10] associates the capacity to the MAC (Medium Access Control) layer rate, i.e. the capacity measured at MAC layer. The model proposed in the following generalizes the capacity concept to each protocol layer and permits to understand why other models fail with regard to wireless links.

The model is based on the notion of *service of a layer*. Each layer offers a service to the upper layer: it permits the upper layer to send information to its even-sided layer at receiver's side. The *service capacity* of layer n is here defined as the maximum rate at which data can be transferred from layer n at sender's side to layer n at receiver's side:

$$C_n(L_{n+1}) = \frac{L_{n+1}}{t_{s_n}(L_n)} \quad (3)$$

where L_{n+1} is the size of the $(n+1)$ -packet, (i.e. the size of the packet at layer $n+1$, which represents the payload size at layer n), and $t_{s_n}(L_{n+1})$ is the service time needed to transfer the $(n+1)$ -packet to the even-sided layer at receiver's side.

To measure capacity, it is necessary to enlist the temporal contributions that compose $t_{s_n}(L_{n+1})$. Assume for simplicity that the $(n+1)$ -packet is smaller or equal to the maximum transfer unit at layer n . In such case, $t_{s_n}(L_{n+1})$ can be recursively expressed as:

$$t_{s_n}(L_{n+1}) = t_{et_n}(L_{n+1}) + t_{s_{n-1}}(L_n) + t_{er_n}(L_{n+1}) \quad (4)$$

The term $t_{et_n}(L_n)$ represents the transmission processing time introduced by layer n . It can be evaluated as the sum of three addends:

1. the time $T_{elab}(L_n)$ required to compute all the fields of the layer n header and add it to the $(n+1)$ -packet. This time can be assumed to be a linear function of L_n if the checksum is calculated;
2. the time T_{queue} spent by packet at layer n in the queue before scheduling operation;
3. the time T_{pro} , which includes different operations carried out by the protocol. Such operations vary from a layer to another. Relevant examples are the time needed for routing, or the time needed to access the physical medium.

The homologue of $t_{et_n}(L_{n+1})$ at receiver's side is $t_{er_n}(L_{n+1})$. Finally, the term $t_{s_{n-1}}(L_n)$ is layer $n-1$ service time.

Let us particularize the proposed model to an IEEE 802.11b protocol stack. In this example, UDP is considered as the transport protocol and fragmentation is supposed to be absent. It can be shown that expression (4), specified for transport layer (i.e. $n=4$), can be rewritten as

$$t_{s_4}(L_5) = ts_{4 \rightarrow 1}(L_5) + \frac{L_5 + \sum_{i=1}^4 H_i}{C_0} + ts_{1 \rightarrow 4}(L_5) \quad (6)$$

where $ts_{4 \rightarrow 1}(L_5)$ includes transmission processing times from transport to physical layer, and is constituted of two terms. The first term, $\alpha_t L_5$, includes all the processing times that are linearly dependent from packet size. The second one, ϕ_t , is the sum of all the processing times that are constant with packet size. The homologue of $ts_{4 \rightarrow 1}(L_5)$ at receiver's side is $ts_{1 \rightarrow 4}(L_5)$, which includes reception processing times from physical to transport layer. It is composed from two contributes, $\alpha_r L_5$ and ϕ_r .

The relation (6) thus becomes

$$t_{s_4}(L_5) = \left(\alpha_t + \alpha_r + \frac{1}{C_0} \right) L_5 + \phi_t + \phi_r + \frac{1}{C_0} \sum_{i=1}^4 H_i = \rho L_5 + \kappa \quad (7)$$

The proposed model takes into account some time contributions that are neglected by VPS and PP techniques. Please note that the RTT in (1) is equal to transport layer service time in (6) plus the time needed to transmit an ICMP message back to the sender, which is independent from packet size. Therefore, when VPS techniques estimate the slope of the "RTT versus L " curve they actually estimate coefficient ρ in (7), which is different from coefficient β in (1). The latter relation could look like to the relation (1), but it is different because:

$$\rho = \alpha_t + \alpha_r + \frac{1}{C_0} \neq \beta = \frac{1}{C_0} \quad (8)$$

Hence, by taking time measurements at transport layer, VPS techniques always underestimate physical layer capacity C_0 . Experiments described in next section will quantify such underestimation for both wired and wireless links.

Let us now consider the functioning of PP and PT techniques. On wired links like Ethernet, assuming no cross traffic is present, the second packet in the pair waits in the queue at physical layer until the first packet is serialized. The dispersion time at the receiver is thus

$$\Delta = T_{end} - T_{start} = \frac{L_1}{C_0} \quad (9)$$

On 802.11b wireless links, assuming no cross traffic is present and if the RTS/CTS procedure is not activated, the second packet in the pair waits in the queue at physical layer until the first packet is serialized, then it waits a SIFS time (Short Inter Frame Spacing) and the time needed for the transmission of the MAC-layer acknowledgment (ACK_{MAC}), and finally it waits a DIFS (Distributed Inter Frame Spacing) time to occupy the channel. The dispersion time measured at the receiver's side is

$$\Delta = \left(2 \frac{L_2}{C_0} + \frac{H_1}{C_H} + \alpha_r L_5 + \phi_r + SIFS + ACK_{MAC} + DIFS \right)_{second\ packet} - \left(\frac{L_2}{C_0} + \frac{H_1}{C_H} + \alpha_r L_5 + \phi_r \right)_{first\ packet} \quad (10)$$

where respectively L_2 and L_5 are *MAC-packet* and *application-packet* size, H_1 the *physical-overhead* size, and C_H the transmission rate of the header (which in the 802.11b standard is different from the data transmission rate, and can be equal to either 1Mbps for a long preamble, or 2 Mbps for a short preamble). Note that the time needed for the propagation of the physical medium is negligible.

If $T_{pro} = SIFS + ACK_{MAC} + DIFS > (\alpha_r L_5 + \phi_r)$, then the dispersion time reduces itself to

$$\Delta = \frac{L_2}{C_0} + \frac{H_1}{C_H} + T_{pro} \quad (11)$$

Packet dispersion on wireless links presents two additional contributions with respect to wired links. This can lead to an underestimate of capacity C_0 , as it will be shown in next section.

IV. Experimental validation

The proposed capacity model has been experimentally validated on both wired and wireless links. To this end, a proper measurement methodology has been designed, and a measurement testbed has been set up (Fig. 1). The purpose is to measure physical and transport layer capacity by measuring each of the three addends in (4). The measurement testbed has consequently been equipped with software and hardware probes, which permit to capture the time instants when the transmitted packet passes through transport and physical layers, at both sender's and receiver's sides. The measurement methodology is the same for both wired and wireless link, whereas the testbeds are slightly different.

Both testbeds include two laptops: the sender, *SA*, the receiver, *SB*, and a processing and control workstation, *RC*. The laptops *SA* and *SB* can be connected either through a 10 Mbps wired link, or through an IEEE 802.11b wireless link; they are, in fact, equipped with both Ethernet port and PCMCIA wireless LAN card.

Moreover, the testbeds consist of a Digital Storage Oscilloscope (DSO) *Agilent Technologies™ Infinium 81004A* (10 GHz bandwidth, 40 GSamples/s maximum sample rate, 32 MSample memory depth), a pair of probes that connect the parallel ports of the two workstations to the DSO, and a third probe that picks the signal on the physical channel. The latter is either a typical DSO probe for a wired link, or an omnidirectional antenna *EM6865™* (frequency range 2 GHz-18 GHz) for a wireless link. The workstation *RC* and the DSO are connected through an IEEE-488 standard interface bus. The software Distributed Internet Generator (D-ITG) has been utilized to generate probing packets [11].

Software probes have been specifically inserted in the source code of D-ITG, in order to activate and deactivate a specific pin of the parallel port each time a packet is sent or received at transport layer. Two probes pick the signals carried by the pin of the parallel ports at sender's and receiver's side and take them to a channel of the DSO. A third probe picks the signal on the communication channel and takes it to another channel of the DSO. The signal picked from the parallel port is exploited as the trigger source for the DSO, which starts the capture in correspondence of the voltage edge (that is, when a packet is sent at transport layer and the pin at sender's side consequently commutes).

The DSO is configured in single acquisition mode. This way, four significant time instants can be singled out:

- T_s , when the data packet leaves the transport layer at sender's side;
- T_r , when the data packet reaches the transport layer at receiver's side;
- T_{pf} , when *the* first bit of the data packet is sensed by the probe on the physical channel;
- T_{pl} , when *the* last bit of the data packet is sensed\ by the probe on the physical channel.

Fig. 2 shows the DSO display after a single acquisition in wireless case. The four relevant time instants are indicated in the figure. Considering that propagation time on the channel can be neglected, the three addends at the right side of equation (6) can thus be calculated as

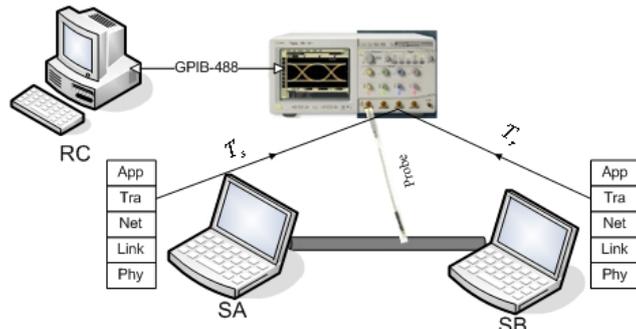


Fig. 1. Wired testbed.

$$ts_{4 \rightarrow 1} = (T_{pf} - T_s) \quad (12)$$

$$\frac{1}{C_0} \left(L_s + \sum_{i=1}^4 H_i \right) = T_{pl} - T_{pf} \quad (13)$$

$$ts_{1 \rightarrow 4} = (T_r - T_{pl}) \quad (14)$$



Fig. 2. The physical signal through DSO

Measurements are performed for different packet sizes from 50 up to 1450 byte, with a step of 100 byte. A LabView™ Virtual Instruments (VI) is utilized to automate the measurements. The VI downloads the three waveforms from the DSO and measures the quantities of interest (12)-(14). For each packet size, measurements are repeated one hundred times. Then, the VI calculates the average value of the quantities of interest (12)-(14) over one hundred acquisitions.

The same measurements are carried out for both TCP and UDP. The coefficients α_i and ϕ_i in relation (7) are then calculated through a linear

regression on the pairs $\{L; ts_{4 \rightarrow 1}\}$. The same is done for coefficients α_r and ϕ_r using pairs $\{L; ts_{1 \rightarrow 4}\}$. Measurement results are enlisted in Table 1.

A. Wired case

The values of $\alpha_i + \alpha_r$ is much smaller than $1/C_0$, which means that

$$\frac{1}{\rho} = \frac{1}{\alpha_i + \alpha_r + 1/C_0} \cong C_0 \quad (15)$$

This explains why VPS techniques, which actually underestimate physical capacity C_0 as $1/\rho$, give quite good results on 10 Mbps wired links. In general, the systematic error ε_s experienced when physical capacity C_0 is estimated as $1/\rho$ is equal to

$$\varepsilon_s = \frac{1}{1 + 1/k} \cdot C_0 \quad (16)$$

where $k = (\alpha_i + \alpha_r)C_0$.

Hence, ε_s is equal to less than 0.3% C_0 in the 10 Mbps link under test, and the contributions of α_i and α_r do not influence the estimate of nominal physical capacity significantly. It is worth noting that if the same workstations are connected through a faster (e.g. 100 Mbps) wired link, ε_s affecting capacity measurement results provided by VPS-based tools is expected to increase quasi-proportionally with the nominal capacity of the link. This means that if the systematic error affecting VPS techniques can be ignored on 10 Mbps links, it can become too high not to be corrected on 100 Mbps and 1 Gbps links.

Correlation coefficients related to the wired link are all not smaller than 0.90. This means that the

Table 1. Measurement results.

	Wired link		Wireless link	
	UDP	TCP	UDP	TCP
Transport time	$\alpha_i = 2.45 \cdot 10^{-10} [s/bit]$ $\phi_i = 22,00 \mu s$	$\alpha_r = 2.71 \cdot 10^{-10} [s/bit]$ $\phi_r = 22,51 \mu s$	$\alpha_i = 2.08 \cdot 10^{-8} [s/bit]$ $\phi_i = 60,02 \mu s$	$\alpha_r = 2.12 \cdot 10^{-8} [s/bit]$ $\phi_r = 90,84 \mu s$
Reception time	$\alpha_i = 4.87 \cdot 10^{-10} [s/bit]$ $\phi_i = 23,09 \mu s$	$\alpha_r = 5.87 \cdot 10^{-10} [s/bit]$ $\phi_r = 23,75 \mu s$	$\alpha_i = 2.21 \cdot 10^{-8} [s/bit]$ $\phi_i = 61,65 \mu s$	$\alpha_r = 2.32 \cdot 10^{-8} [s/bit]$ $\phi_r = 90,36 \mu s$
Transmission Correlation Coefficient	0.90	0.92	0.94	0.97
Reception Correlation Coefficient	0.91	0.95	0.96	0.99

hypothesis of a linear dependence of $ts_{4 \rightarrow 1}$ and $ts_{1 \rightarrow 4}$ from packet size, which was made in section III, can be accepted.

According to the proposed model, transport layer capacity, $C_4(L_5)$, is also estimated for both protocols. The evolutions of C_4 versus packets size shows an increasing trend, and reaches about 9 Mbps for UDP and 8 Mbps for TCP. As expected, UDP transport capacity is always greater than TCP transport capacity. This is a consequence of the overhead introduced by TCP protocol, which is heavier than UDP overhead.

When capacity is measured through dispersion Δ , no significant underestimation is experienced.

B. Wireless case

When measurements are performed on the wireless link, the testbed is set up in a semi-anechoic chamber that prevents from in-channel interference, in order to guarantee the same operating conditions over successive measurements.

Also in the wireless case, the hypothesis of a linear dependence of $ts_{4 \rightarrow 1}$ and $ts_{1 \rightarrow 4}$ from packet size holds. It is noticeable that the values of α_r and α_t are much higher than in the 10 Mbps wired link. In particular, the systematic error is higher than 10% C_0 , since $k > 0.1$. The proposed model thus explains why the accuracy of capacity measurements carried out by VPS-based tools on wireless links is seriously compromised [2].

As observed in the wired case, the underestimation grows up as nominal physical layer capacity increases. In fact, if C_0 is 11 Mbps the capacity estimated through VPS techniques is about 6 Mbps, while in the case of 802.11g, whose nominal capacity equal to 54 Mbps, the estimated capacity would be about 16 Mbps.

PP and PT techniques measure capacity by evaluating the dispersion time expressed by relation (11). The values of H_1 , C_H and T_{pro} in (11) can be looked up in the standard [12], whereas $L_2/C_0 + H_1/C_H$ is measured through the DSO. Thus, Δ comes out to be about 7.8 Mbps, which implies an underestimation of physical capacity C_0 of about 3.2 Mbps.

According to the aforementioned model, the transport capacity is evaluated for both transport protocol. Also in this case, the UDP transport capacity is always greater than the TCP transport capacity. Moreover, the smallest transport layer capacities are obtained in the wired case for both TCP and UDP. This is a consequence of the heavier overhead introduced by CSMA/CA protocol.

V. Conclusion

A new model for the capacity of a communication network, which extends the concept to all the layers of the protocol stack, has been presented. Thanks to a refined study of the dependence of transmission delays from processing times at different protocol layers, the systematic capacity underestimation provided by existing techniques has been theoretically explained, and experimentally validated. The proposed validation methodology can be also used to carry out accurate capacity measurements on both wired and wireless links. Further details on the model and experiment results will be given in the full paper.

References

- [1] R. S. Prasad, M. Murray, C. Dovrolis, K. Claffy, "Bandwidth Estimation: metrics, measurement techniques and tools," IEEE Network, Nov/Dec. 2003.
- [2] A. B. Downey, "Using pathchar to estimate Internet link characteristics," *Proc. of the 1999 ACM SIGMETRICS Int. Conf. on Measurement and Modeling of Computer Systems*, May 01-04, 1999, Atlanta (USA), p.222-223.
- [3] B. A. Mah, "pchar: a Tool for Measuring Internet Path Characteristics", Feb. 1999, <http://www.employees.org/~bmarsh/Software/pchar>.
- [4] V. Jacobson, "Pathchar: A Tool to Infer Characteristics of Internet Paths," <ftp://flp.ee.lbl.gov/pathchar/>, Apr. 1997.
- [5] L. Angrisani, S. D'Antonio, M. Vadursi, G. Ventre, "Packet Delay Models in Packet-Switched Networks: Performance Assessment through Capacity Measurements", *Proc. of ESMC 2003 European Simulation and Modeling Conference*, October 2003, Naples, Italy, pp.465-469.
- [6] R. Kapoor, L. J. Chen, L. Lao, M. Gerla, M. Y. Sanadidi "CapProbe: A Simple and Accurate Capacity Estimation Technique," *Proc. of ACM SIGCOMM 2004*, Portland, USA, 30 Aug.- 03 Sept. 2004, pp.67-78.
- [7] L. J. Chen, T. Sun, G. Yang, M. Y. Sanadidi, M. Gerla "Ad hoc probe: path capacity probing in wireless ad hoc networks," *Proc. of IEEE First Intern. Conf. on Wireless Internet*, July 2003, pp.156-163.
- [8] T. Sun, L. J. Chen, G. Yang, M. Y. Sanadidi, M. Gerla "Path capacity estimation in IEEE 802.15.4 enabled wireless sensor network via senprobe," *Proc. of IEEE Consumer Comm.unications and Networking Conf. 2006*, vol.2, 8-10 Jan. 2006, pp.1301-1302.
- [9] L. Angrisani, A. Botta, A. Pescapè, M. Vadursi, "Measuring Wireless Links Capacity," *Proc. of IEEE Int. Symp. on Wireless Pervasive Computing ISWPC 2006*, 16-18 January 2006, Phuket (Thailand), pp.1-5.
- [10] M-J. Ho, J. Wang, K. Shelby and H. Haisch "IEEE 802.11g OFDM WLAN Throughput Performance," IEEE Vehicular Technology Conf. 2003, Vol.4, 6-9 Oct. 2003, pp.2252-2256.
- [11] <http://www.grid.unina.it/software/ITG>
- [12] IEEE Standards for Information Technology- Telecommunications and Information Exchange between System- Local and Metropolitan Area Networks- Specific Requirements- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999.