Performance evaluation of copper-based Gigabit Ethernet interfaces

Conference Paper · December 2002

DOI: 10.1109/LCN.2002.1181845 · Source: IEEE Xplore

Paul A. Gray
University of Northern Iowa

Performance Evaluation of Copper-based Gigabit Ethernet Interfaces

Paul Gray, Anthony Betz
Department of Computer Science
University of Northern Iowa
Cedar Falls, IA 50614-0507
Contact Author: gray@cs.uni.edu

Abstract

In today's high-performance computational environments communication substrates often stand out as the major limiting factor to performance, accessibility, and stability. Such platforms are naturally well-suited for high-performance network foundation built upon Gigabit Ethernet, or "GigE" technologies. Gigabit Ethernet hardware continues to become more attractive in price and availability, making it a viable choice for situations where large bandwidth is demanded. And copper-based Gigabit Ethernet holds additional advantages when considering the additional cost associated with rewiring for fiber technologies. This report characterizes the current capabilities of copper-based gigabit solutions and identifies requisite components for optimal performance. These aspects are compared under various Linux and Windows 2000 configurations as they relate to latency, throughput, and stability for a variety of hardware configurations.

1. Introduction

Inter-process communication is often the most critical aspect in the overall performance of tightly-coupled cluster environments, high-volume database servers, or any distributed system where data movement is imperative to functionality. In general, the measure of an application's performance involves numerous aspects and inefficiencies can occur at many different layers between the actual application and the underlying hardware layers. Naturally, Gigabit Ethernet technology is growing as a viable solution to relieving bottlenecks within compute environments and server connections at the link layer. Although Gigabit Ethernet has been standardized for fiber optic connections since 1998, only recently was the standard for copper-based Category 5 (Cat-5) cable approved. [5] has a thorough presentation of the Gigabit Ethernet standard. See [4] for a description of copper-based Gigabit Ethernet's evolution.

Gigabit Ethernet using Cat-5 cabling is most attractive inasmuch as most networking infrastructure is already utilizing copper, greatly reducing rewiring costs. While Gigabit Ethernet has found a niche in high-performance clustering and server environments, the recent adoption of the 802.3ae standard for 10 Gigabit Ethernet raises the bar, and it is anticipated that Gigabit Ethernet infrastructures will soon be elevated so as to be as commonplace as today's vast installation base of Fast Ethernet.

But unlike the more mature Fast Ethernet technologies where near-peak Fast Ethernet performance is commonplace, deriving peak gigabit performance from a Gigabit Ethernet adapter can be quite difficult. In addition, identification of the factors that significantly diminish performance can also be an elusive task. This paper provides a snapshot into the current capabilities of various copper-based Gigabit Ethernet adapters. Extending the results presented in [2], this report examines the influences of jumbo frames, PCI bus speed, variations of drivers and of operating systems as factors that hold influence on throughput, stability, and latencies of several copper-based Gigabit Ethernet adapters available on the market today.

2. Testing Environment

Figure 1 enumerates the cards utilized in this report along with the associated chipset and drivers. Peak throughput, latency, and stability are analyzed using *netpipe* [6], version 2.4 from Ames Laboratory. Netpipe is a utility that examines latency and throughput using variations on standard memory-to-memory ping-pong tests. Messages increasing in size from 1 byte to 64Mbyte or larger are exchanged between a transmitter and a corresponding receiver. Each ping-pong test is repeated multiple times so as to derive a more accurate timing. Latency measurements for Netpipe are derived from the round-trip times for small-sized packets.

The hardware utilized to investigate performance included a pair of identically-configured Athlon 1500MP sys-

Manufacturer	*Model*	*Linux Driver(s)*	*Windows 2000 Driver*	*Chipsets*
3Com	3C996BT	bcm5700 v.2.0.28 (3Com), Tigon v0.98 (kernel)	v.2.32.0.0	Broadcom BCM5701
ARK	Soho 2000T	ns83820 v0.15 (kernel)	v.5.0.1.24	National Sem. dp83820
ARK	Soho 2500T (32-bit)	ns83820 v0.15 (kernel)	v.5.0.1.24	National Sem. dp83820
SMC	9462TX	ns83820 v0.15/v0.17 (kernel)	v.5.0.1.24	National Sem. dp83820
D-Link	DGE-500T (32-bit)	ns83820 v0.15/v0.17 (kernel)	v.5.0.1.24	National Sem. dp83820
Intel	1000XT	e1000, v4.1.7 (Intel)/ v4.2.4-k2 (kernel)	v.3.63.363.0	Intel 82544EI
NetGear	GA-302T (32-bit)	ac1000 (NetGear), Tigon v0.98 (kernel)	v.1.0.0.0	Altima (nee Broadcom)
NetGear	GA-622T	gam (NetGear), ns83820 v0.15/v0.17 (kernel)	v.5.0.1.24	National Sem. dp83820
SysKonnect	SK98-21	sk98lin v4.0.6 (kernel)	v.3.7.0.0	L5A9338

Figure 1. Card manufacturers, models, drivers and chipsets used in this evaluation.

tems. Each system had a single processor mounted into a Tyan S2466N motherboard with 512MB of DDR RAM. The Tyan S2466N offers two 64-bit and four 32-bit PCI slots. The 64-bit PCI slots are jumperable to 66MHz or 33MHz, which also facilitated our testing on the influence of the PCI bus speed. Each system was configured to boot into Debian 3.0 GNU/Linux and Microsoft Windows 2000 Professional. Versions 2.4.7, 2.4.19-pre9-ac2, and 2.5.7 of the Linux kernel were used for testing on the GNU/Linux operating system. Service pack 2 was applied to the Windows 2000 installation. As switching technologies would introduce an additional layer into the testing environment, the test systems were joined together back-to-back, using Cat-5e crossover cables (6-, 12-, and 18-feet lengths).

The Netpipe utility offers several different communication substrate tests, such as PVM-, MPI-, and TCP-based. The performance measurements presented herein utilized the TCP-based suite. While other communication protocols are arguably better suited for high-speed communication frameworks, the TCP-based benchmarks provide a more universal perspective into the performance that can generally be expected of any particular card.

The testing process includes an analysis of the effect of the speed of the PCI bus and the TCP MTU settings. The PCI bus was jumpered at both 66MHz and 33MHz. In both bus configurations, TCP MTU sizes were adjusted from the standard 1500-bytes, through 3000-, 4000-, 6000-, and 9000-bytes provided the driver was capable of accommodating the larger frame sizes.

The plots presented for each card, reflect measured throughput for various message sizes (see, for example, Figure 2), for the two PCI bus speeds (33MHz, 66MHz), and for various TCP MTU sizes supported by the driver (see, for example, Figure 3).

3. Driver Performance under GNU/Linux

The baseline for our benchmarks in this section involves the cards listed in Table 1 in conjunction with the "out-of-the-box" device drivers. Modules distributed in source form were compiled without modification to the source. Binary module formats of drivers were loaded into the Linux kernel without extra parameters. All tests results reflect performance observed for the 2.4.17 Linux kernel.

Performance results reflecting variations on these specifications, which include supplying of additional module parameters and updated drivers in newer Linux kernels, are given in Section 5.

3.1. 32-bit Cards

This section begins with a look at the three 32-bit copper-based gigabit cards in the collection; the NetGear GA302T, the D-Link DGE-500, and the Ark Soho-2500.

3.1.1 GNU/Linux Performance: NetGear GA302T

Performance for the NetGear GA302T is shown in Figure 2. The "`ac1000`" Linux drivers used for the NetGear GA302T are developmental drivers that were provided upon request from NetGear. While not available as "production" drivers, the ac1000 drivers were found to be very capable, stable, and delivered smooth performance.

The Broadcom chipset is also driven by the Tigon `tg3` driver, found in the 2.4.19 Linux kernel. Figure 2 shows the results of tests using available bus configurations (66MHz and 33MHz). The NetGear GA302T's peak throughput exceeds all other out-of-the-box 1500 MTU benchmarks – including those of the 64-bit cards.

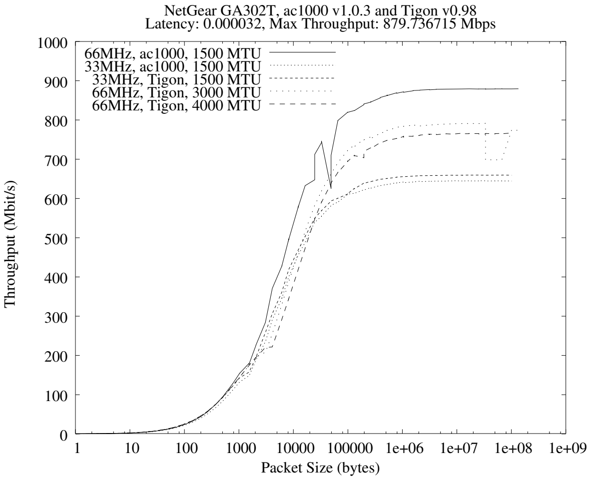

Figure 2. Performance for the NetGear GA302T.

3.1.2 GNU/Linux Performance: D-Link DGE-500T

Performance of the D-Link DGE-500T is shown in Figure 3. The D-Link DGE-500T is one of several copper-based giga-

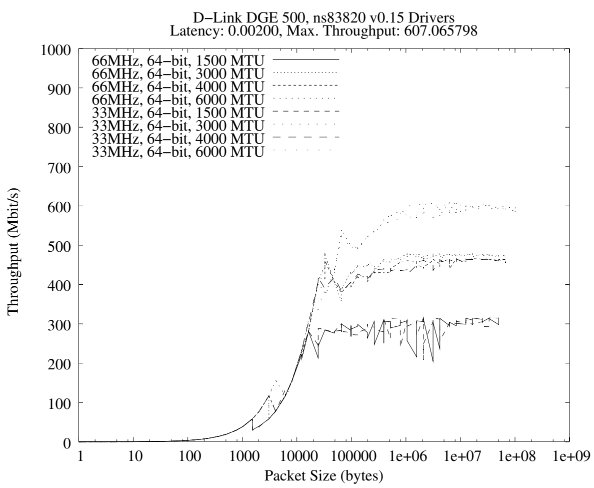

Figure 3. Performance for the D-Link DGE-500T.

bit cards in the test that are based upon the National Semiconductor's dp83820 chipset. There are several drivers for the dp83820 chipsets. D-Link provides tuned drivers for the DGE-500T that work in earlier Linux kernels, but have yet to release binary drivers or source code that is compatible with the later 2.4.x series kernels. Nonetheless, the standard Linux kernel provides support for this chipset through Ben LaHaise's ns83820 driver. The v0.15 version of the ns83820 supports MTU tests up through 6000 bytes.

Figure 3 shows performance characteristics consistent with using the Linux kernel's ns83820 driver, version v0.15, in conjunction with the dp83820 chipset. Note that with the ns83820 v0.15 drivers as seen here, throughput is highly dependent upon MTU size, but not significantly dependent upon the bus speed.

3.1.3 GNU/Linux Performance: ARK Soho-2500

Also based upon the National Semiconductor dp83820 chipset is the 32-bit ARK Soho-2500. This card consists of nearly-identical core components compared to other

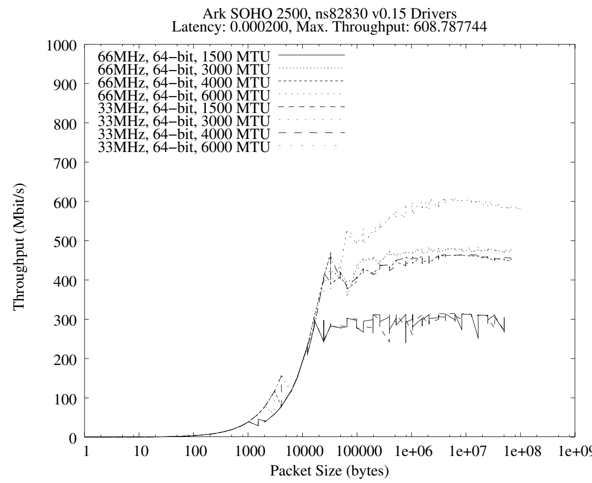

Figure 4. Performance for the Ark Soho-2500.

dp83820-based cards. Observable performance was also similar, yet these cards were purchased for less than half of the cost of the more name-brand varieties.

Compare the performance of the Ark Soho-2500 depicted in Figure 4 with that of the D-Link DGE-500T in Figure 3. Both cards show comparable throughput with the Linux kernel ns83820 v0.15 driver, similar behaviors with the causality of MTU size on throughput, and the same latency measurements.

While there are many similarities between the dp83820-based cards, as these performance graphs indicate, there are subtle differences when working with alternative drivers. See Section 5 for comments on working with multiple drivers.

3.2. 64-bit Cards

The 64-bit copper-based GigE cards in our test include the Ark Soho-2000, SMC 9462TX, NetGear GA622T, 3Com 3C996B, Intel 1000XT, and SysKonnect SK98-21.

The first three of these cards utilize the same dp83820 chipset as presented in Figures 3 and 4, but with twice the bus width. These cards provide an ideal transition in these investigations in moving from 32-bit to 64-bit interfaces.

The 3Com 3C996B, Intel 1000XT, and SysKonnect SK98-21 offer cards with chipsets unique to the testing pool. These cards in particular, are more commonly marketed to the server and clustering market where large-bandwidth demands are the status quo.

3.2.1 GNU/Linux Performance: Ark Soho-2000

The Ark cards form the framework for the transition from the 32-bit tests. The Soho-2000, like the Soho-2500 and the D-Link DGE500T, is based upon the dp83820 chipset, and so this provides insight into the benefits of extending the bandwidth of the PCI bus from 32-bit to 64-bit, within

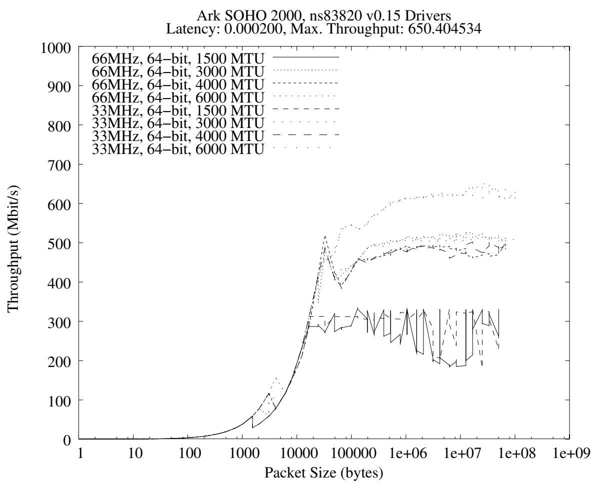

Figure 5. Performance results for the 64-bit Ark Soho-2000.

the framework of the same chipset on the network interface. Comparing the performance results for the 64-bit Soho-2000, shown in Figure 5, with the performance results for the 32-bit varieties that utilize the same chipset, Figures 4 and 3, we see an modest increase in throughput at the upper end, and a bit of more instability in the 1500 MTU tests: from 609Mbps to 650Mbps. So in this case, with this particular driver, there is not much benefit derived from the extended bus connection.

3.2.2 GNU/Linux Performance: SMC 9462TX

Figure 6 shows the performance results for the SMC 9462TX. A notable observation in comparing the 9462TX with similarly-equipped dp83820 cards (Figures 3, 5, 4, and the NetGear GA622T's v0.15 ns83820 performance to be shown in Figure 7) is the reliability of the card under the 1500 MTU runs. The SMC 9462TX shows a much smaller deviation in the asymptotic limit as packet size increases.

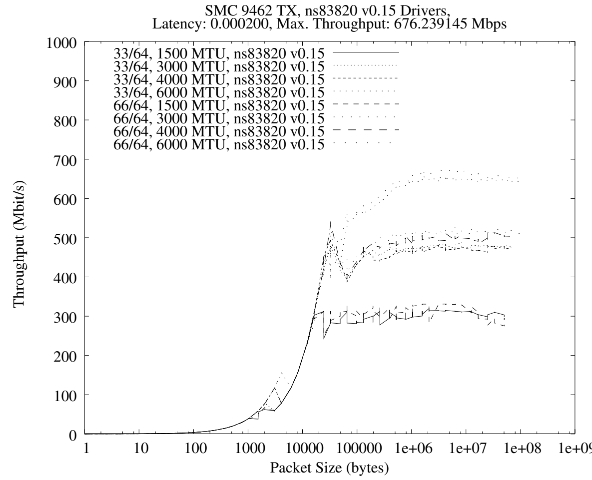

Figure 6. Performance for the SMC 9462TX.

3.2.3 GNU/Linux Performance: NetGear GA622T

Of all the dp83820-based cards tested, the NetGear GA622T was the only card that offered additional Linux drivers that were compatible with the more recent Linux 2.4.x kernels. The unmodified NetGear "`gam`" drivers provide further testing and insight into the capabilities of cards based upon the dp83820 chipset.

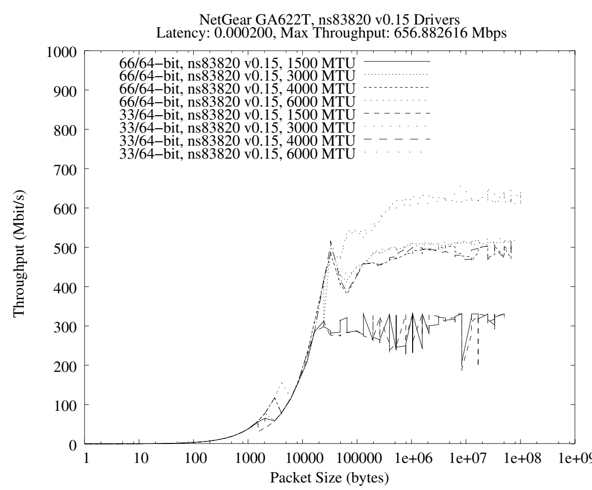

Figure 7. Performance for the NetGear GA622T (ns83820 v0.15).

Performance for the NetGear GA622T under the ns83820 v0.15 drivers distributed with the Linux 2.4.17

kernel is shown in Figure 7. As one might expect, under the ns83820 v0.15 drivers the NetGear GA622T performs on par with the performance mentioned previously for similarly-equipped cards (c.f. Figures 3, 4, 5, and 6).

The NetGear gam drivers are extensions to the core GNU/Linux drivers from National Semiconductor, adapted

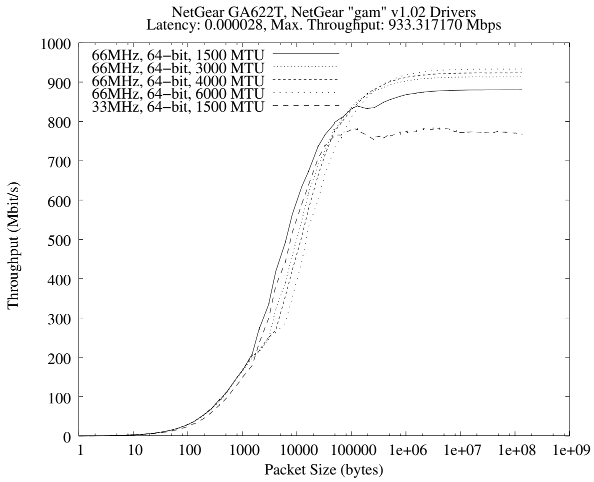

Figure 8. Performance for the NetGear GA622T (NetGear "gam").

for the header file changes found in the later 2.4.x series Linux kernels. The unmodified gam driver supported other MTU sizes, but were omitted from Figure 8 for clarity. As shown in Figure 8, the unmodified gam driver offers very stable, competitive high performance.

3.2.4 GNU/Linux Performance: 3Com 3C996BT

The 3Com 3C996BT is built around a Broadcom BCM5701 chipset, and supports a range of full-featured configurations for high-bandwidth servers. Traffic shaping, trunking, and fault tolerance aspects are some of the additional features offered by the 3C996BT.

In terms of throughput, performance results for the 3C996BT are shown in Figure 9. The graph indicates a performance hit for using a 9000 MTU size in the 33MHz bus. Peak throughput increases from 741.33 Mbps at 1500 MTU to a peak peak throughput of 785.93 Mbps at 6000 MTU, and then decline to 618.86 Mbps at 9000 MTU. The 3C996BT was also benchmarked using the Tigon drivers of the 2.4.19-pre9-ac2 kernel (see Figure 26).

3.2.5 GNU/Linux Performance: Intel 1000XT

The Intel 1000XT is also a feature-rich card, offering features such as extensive jumbo frame support, load balancing features, fault tolerance, and 133MHz PCI capabilities.

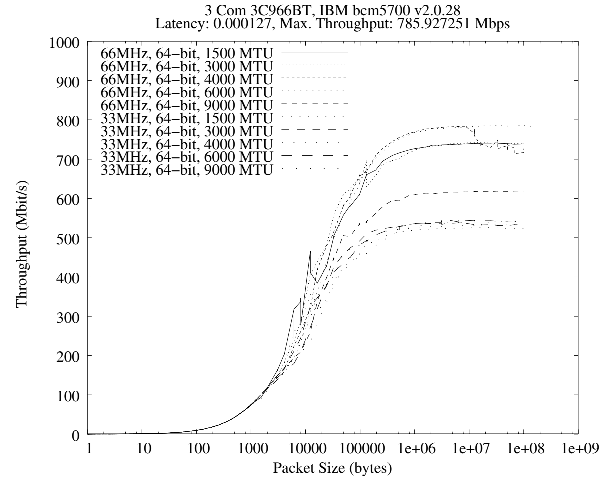

Figure 9. Performance for the 3Com 3C996BT.

While tuning the parameters and tweaking drivers improved the card's throughput and stability, Figure 10 shows that care should be taken when using default parameters with the 4.1.7 version of the e1000 drivers in conjunction with jumbo frames. In benchmark tests where the MTU size was

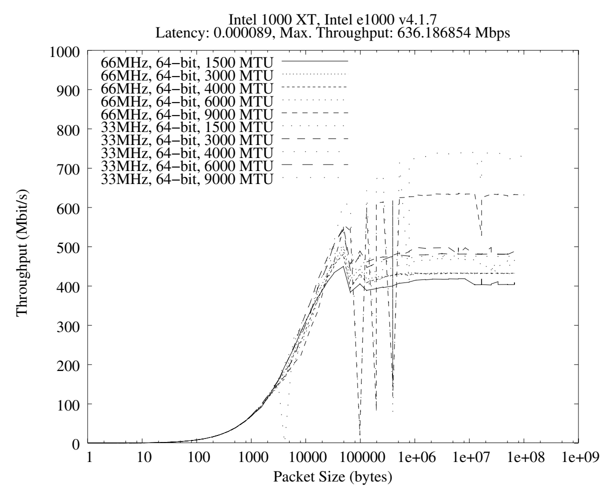

Figure 10. Performance for the Intel 1000XT.

4000 or larger, the 1000 XT with default parameters experienced systematic and reproducible connectivity dropouts.

However, this behavior is easily solved by modifying the default driver options. With the help of the GNU/Linux technical support team within Intel's Network division, the dropout phenomena was able to be resolved completely. In addition, suggestions from the Intel support team provided dramatic increases in overall throughput and stability (discussed in detail in Section 5, see Figure 21), making the 1000XT one of the top all-around performers.

3.2.6 GNU/Linux Performance: SysKonnect SK98-21

The SysKonnect SK98-21 proved to have the best throughput, the lowest latencies, and greatest stability of all cards in the out-of-the-box test. As can be seen in Figure 11,

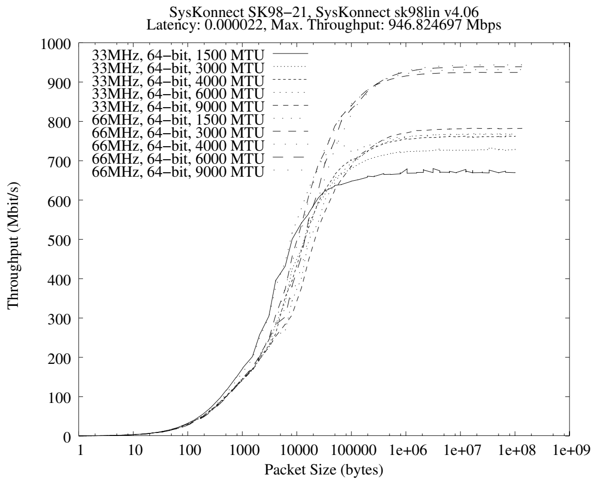

Figure 11. Performance for the SysKonnect SK98-21.

the SK89-21's proved to be especially capable in the upper-MTU range. The peak throughput of the SK98-21 pair of cards in the GNU/Linux systems was the highest of all cards tested, including all Windows 2000 runs and all runs performed on other cards with tweaked drivers and parameter options.

4. Driver Performance under MS Windows 2000

In this section, performance results are presented for the collection of GigE network cards under the same hardware environment, with drivers aligned to the Microsoft Windows 2000 operating system. In general, these results show vastly different characteristics between GNU/Linux and Windows performance. The performance results below reflect scenarios similar to those depicted in Section 3. At times, the Windows 2000 drivers for a certain card offered only pre-selected frame sizes in which case the frame sizes consistent with those selected in Section 3 were used.

4.1. 32-bit Cards

This section begins with another look at the three 32-bit copper-based gigabit cards in the collection: this time under a Windows 2000 environment. These include the NetGear GA302T, the D-Link DGE-500, and the Ark Soho-2500.

4.1.1 Windows 2K Performance: NetGear GA302T

Similar to the behavior seen in Figure 2, the NetGear GA302T performance in Microsoft Windows 2000, Figure 12, is significantly dependent upon the speed of the underlying bus. Throughput, however, was not identical. The

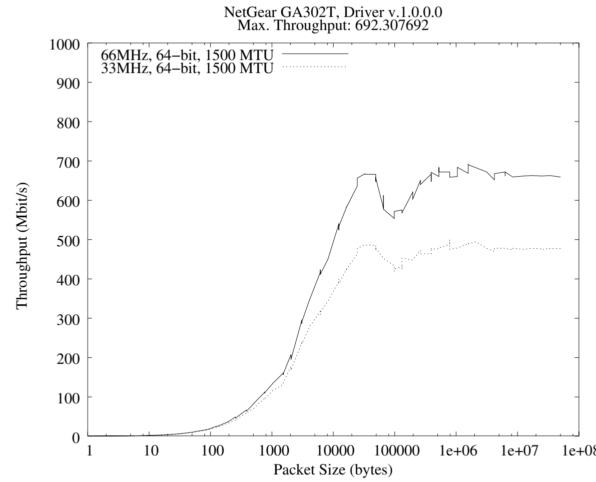

Figure 12. W2K Performance for the NetGear GA302T.

peak W2K throughput of 692.3Mbps is only 73% of the 878.7Mbps peak throughput attained in Linux.

4.1.2 Windows 2K Performance: D-Link DGE-500T

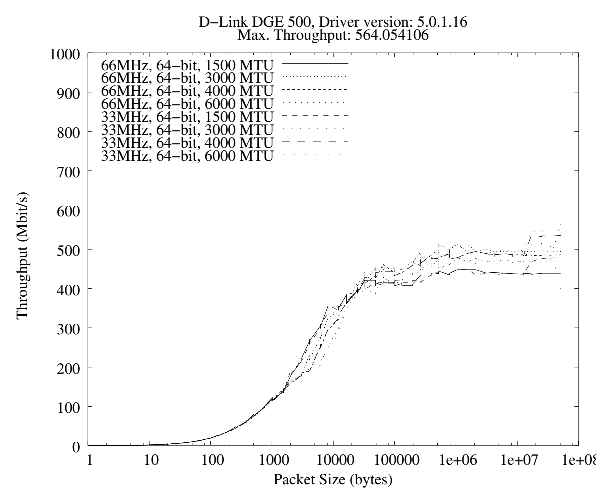

Figure 13. W2K Performance for the D-Link DGE500T.

The Windows 2000 performance for the 32-bit D-Link DGE500T shown in Figure 13 shows a characteristic consistent with all of the National Semiconductor dp83820-based chipsets: namely, a general indifference to the MTU setting. The W2K peak throughput of 654.1Mbps is 7% greater than the 607.1Mbps achieved using the v0.15 ns83820 driver found in the 2.4.17 Linux kernel (shown in Figure 3) and is 5% lower than the 688.2Mbps achieved by the v0.17 ns83820 driver found in the 2.4.19 Linux kernel (not shown).

4.1.3 Windows 2K Performance: Ark Soho-2500

Based upon the same chipset as the D-Link DGE-500T described above, the Ark Soho-2500 exhibits nearly identical characteristics. Minor differences between these

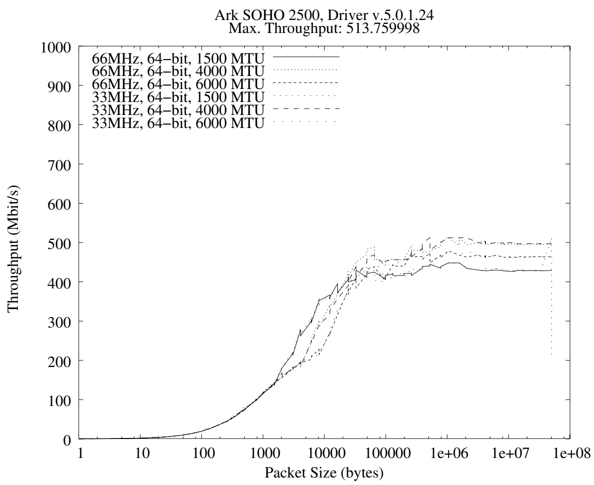

Figure 14. W2K Performance for the Ark SOHO-2500.

two dp83820-based chipsets can be seen in the maximum throughput. While the DGE-500T has a greater throughput, this is attributable to the spurious throughput peaks at the larger packet sizes in the DGE-500T tests.

4.2. 64-bit Cards

This section examines the performance of the six 64-bit cards in the test under the drivers for Windows 2000. This includes the Ark Soho-2000, SMC 9462TX, NetGear GA622T, 3Com 3C996BT, Intel 1000XT, and SysKonnect SK98-21.

4.2.1 Windows 2K Performance: Ark Soho-2000

The Ark Soho-2000 (Figure 15) shares the same chipset as the SMC 9462TX (Figure 16) and the NetGear GA622T (Figure 17). Comparing these performance results, one sees that the Soho-2000 enjoys an advantage in peak throughput:

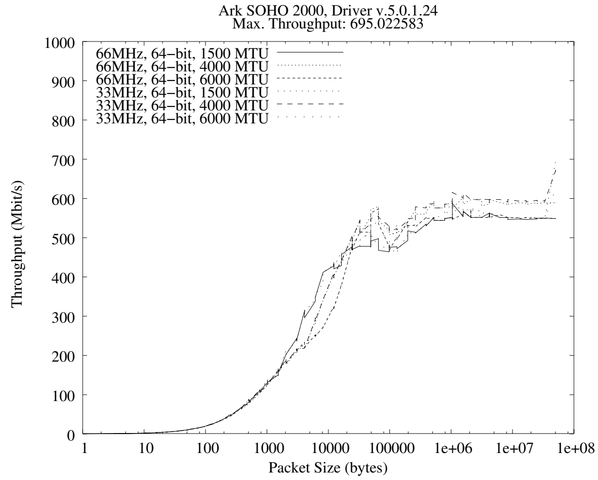

Figure 15. W2K Performance for the Ark SOHO-2000.

695Mbps, 684.5Mbps, and 630Mbps respectively. However, the peak throughput for the Soho-2000 is more of an anomaly, appearing almost spuriously at the end of the trial (reference the trailing throughput peak in Figure 15).

4.2.2 Windows 2K Performance: SMC 9462TX

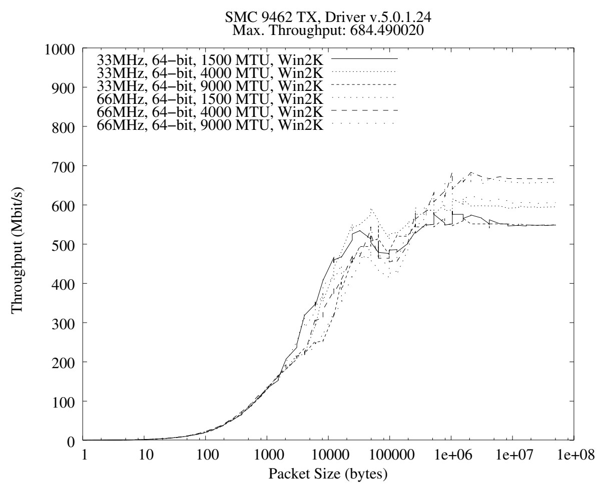

Figure 16. W2K Performance for the SMC 9462TX.

Whereas the W2K driver for the Ark soho line support arbitrary frame size specification up through 6000 MTU, the W2K driver for the SMC 9462TX supports pre-defined frame sizes. Performance results for the frame sizes consistent with the tests in Section 3 are presented in Figure 16.

As is generally true with the dp83820-based cards, comparing Figure 16 with the corresponding ns83820 v0.15 driver performance in Figure 6 one can see that the W2K driver holds a significant advantage for the smaller MTU sizes and in peak throughput.

4.2.3 Windows 2K Performance: NetGear GA622T

With the NetGear GA622T, one is able to put together a full picture of the influence of the drivers. Depicted in Figure 17 is the performance of the NetGear GA622T under Windows 2000. Note that like the Ark soho-2500, the Ark soho-2000,

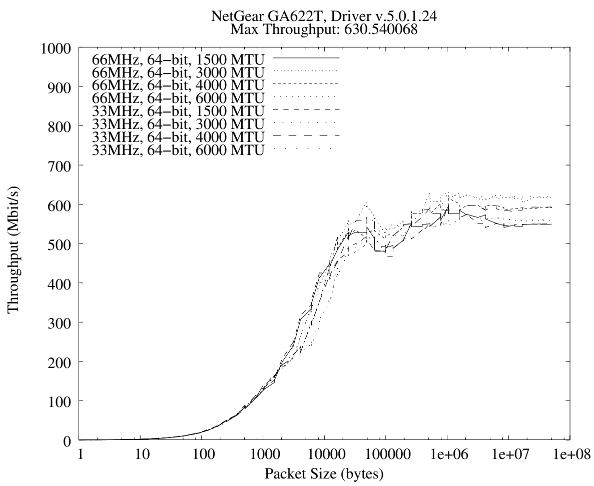

Figure 17. W2K Performance for the NetGear GA622T.

and the SMC 9462TX, this card is built around a 64-bit interface to the National Semiconductor dp83820 chipset. As such, it can be driven by one of several drivers (see Figure 7, Figure 8, Figure 17 and Figure 24).

4.2.4 Windows 2K Performance: 3Com 3C996BT

Of all the cards tested under the Windows 2000 operating system, the 3Com 3C996BT shows the most partiality to the size of the frame, but to the detriment of performance. Peak throughput drops nearly linearly from 666.7Mbps at 1500MTU to 514.5Mbps on the 66MHz runs. Similar drops are observed for the 33MHz runs as well, and for the outcomes of the GNU/Linux runs (reference Figure 9). This observable phenomena and other aspects of the 3Com 3C996BT are addressed in Section 5.4, where modifications to the operating system setup improve performance significantly and reverse the decreasing performance caused by increased MTU size.

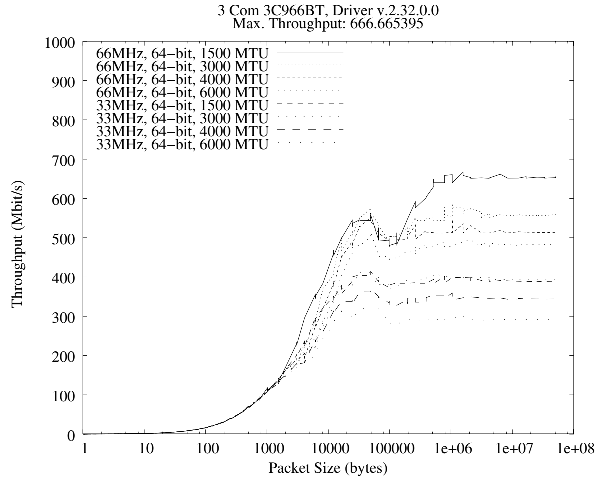

Figure 18. W2K Performance for the 3Com 3C996BT.

4.2.5 Windows 2K Performance: Intel 1000XT

Under Windows 2000, the Intel 1000XT loses the dropouts that were observed in the GNU/Linux environment, and is

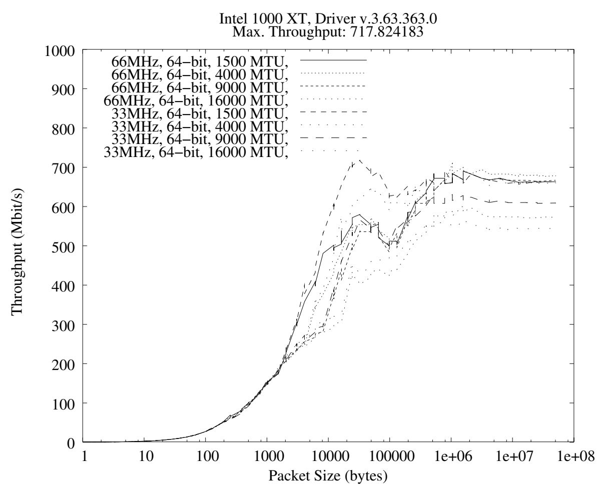

Figure 19. W2K Performance for the Intel 1000XT.

one of the top out-of-the-box performers. The W2K performance for the Intel 1000XT is shown in Figure 19.

4.2.6 Windows 2K Performance: SysKonnect SK98-21

The SysKonnect SK98-21 is another top performer under these out-of-the-box runs. Figure 20 indicates that the card has widely-varying throughput for low-MTU settings and the higher bus speed. One explanation for this is the driver's

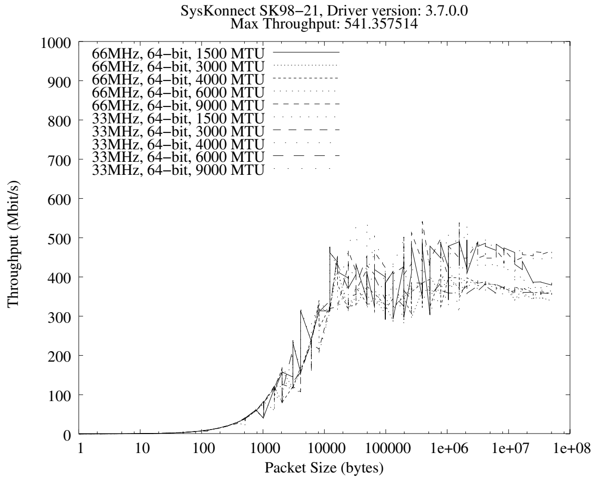

Figure 20. W2K Performance for the SysKonnect SK98-21.

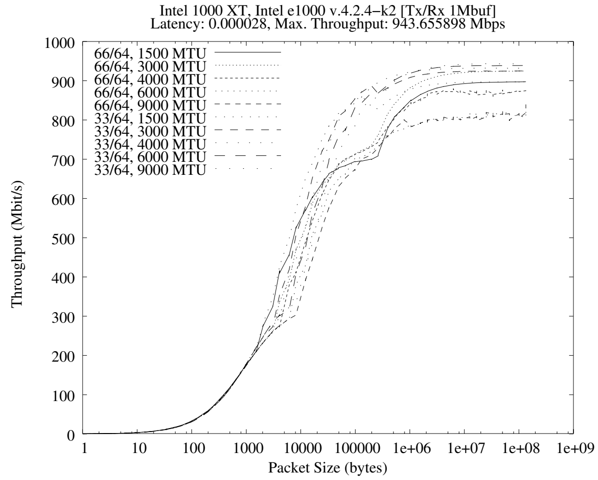

Figure 21. Improved performance in the Intel 1000XT 4.2.4-k12 Driver.

interrupt moderator feature, which throttles interrupt requests when they are being issued at too fast of a rate. At the lower MTU settings with higher bus speeds, the number of interrupts thrown by the card would reach the interrupt moderator threshold faster, causing the wide variance in the throughput. See Figure 23 in Section 5 for a perspective on the impact of the interrupt moderator and throughput.

5. Performance Tuning

The performance results presented in Section 3 and Section 4 reflect the standard implementation of unmodified drivers provided by either the Linux kernel or the card vendor. Beta drivers, parameter options, and variations on environmental aspects provide a seemingly infinite array of testing opportunities. Finding "optimal" combinations of variables can be a true exercise in patience. This sections presents a few findings of specific interest, where aspects of throughput or stability are measurably increased by variations on the testing environment.

5.1. Throttled Interrupts

Many drivers offer features that limit, queue or otherwise throttle the number of interrupt requests that originate from the card. Relaxing these limitations often increases throughput considerably but at the cost of CPU utilization. For a complete analysis of the relationship between CPU utilization, interrupt settings, and throughput, see [3].

Earlier (see Figure 10), it was shown that the Intel 1000XT exhibited performance dropouts when tested under MTU sizes larger than 4000 bytes. By providing the module options `RxIntDelay=0` and `TxIntDelay=0`, in conjunction with the newer version of the Linux kernel driver, performance rebounded significantly (see Figure 21, compare with Figure 10). By default, the `RxIntDelay` and `TxIntDelay` are determined by the driver's `DEFAULT_TIDV` parameter, which is set to 64 in the 4.1.7 driver.

Decreasing the interrupt delay however, does not always increase a card's performance and can be detrimental to stability. Variations on reducing the 3Com 3C996BT's interrupt delay through the use of the driver's `rx_coalesce_ticks` and `tx_coalesce_ticks` parameters or automatically, using `rx_adaptive_coalesce=1` and `tx_adaptive_coalesce=1` parameters *decreased* the peak throughput of the card. That is to say, lowering the interrupt delay had a negative impact on throughput for the 3C996BT. Only by increasing the size of the socket buffers was this phenomena rectifiable with the 3C996BT (see Figures 27 and 28).

The dp83820-based chipsets offer a feature referred to as "interrupt holdoff." The v0.17 version of the ns83820 driver in the 2.4.19-pre Linux kernels takes this into account, which is one of the reasons for the increased performance and decreased latency in the v0.17 driver (see the next section). One can also modify the v0.15 driver so as to reduce the interrupt delay, leading to a latency drop from 200 μsec down to 20 μsec. Throughput increases considerably, but stability was an issue for larger frame sizes. Figure 22 shows the corresponding improvement in performance in reducing the interrupt delay for the v0.15 for MTU sizes of 1500 and 3000.

Relaxing the interrupt restrictions also benefits perfor-

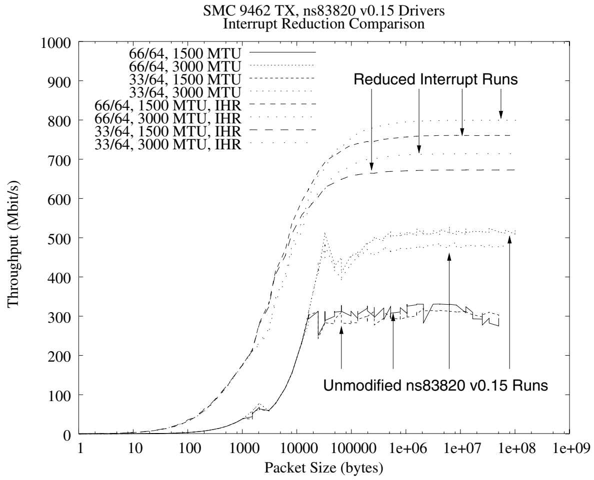

Figure 22. Improved performance through reduction of interrupt delays.

mance under the Windows 2000 drivers. Figure 20 showed widely-varying performance for the SysKonnect SK98-21 under low MTU and high bus rates. Figure 23 shows the effect of increasing the number of allowable interrupts, relaxing the role of interrupt moderator.

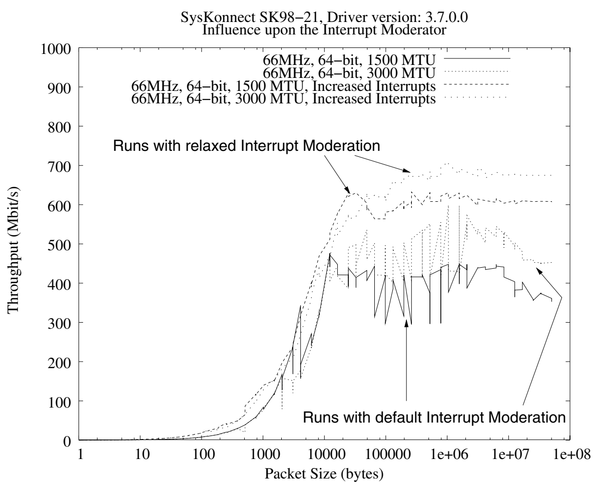

Figure 23. Improved performance through reduction of interrupt delays.

5.2. Improved ns83820 v0.17 Drivers

A significant number of cards that are part of the performance tests in this report are built around the National Semiconductor dp83820 chipset. While some vendors strive to provide Linux drivers for these cards, all but NetGear had drivers compatible with the latest 2.4.x series kernels. In lieu of vendor supported drivers, Ben LaHaise from RedHat actively maintains a robust, full-featured, open source driver for the dp83820 chipset, the ns83820 driver. The v0.15 release of the ns83820 driver was used for the ns83820-related performance results depicted throughout Section 3.

Version v0.16 of the ns83820 driver was a code cleanup, and v0.17 is to focus on performance enhancements. v0.17 of the ns83820 kernel driver has been integrated into the 2.4.19 Linux kernel. At the time of this writing, the 2.4.19 kernel is still under development. The v0.17 ns83820 driver used for testing in this report was part of the 2.4.19-pre9-ac2 version of the Linux kernel, which was a preliminary version of the 2.4.19 kernel with Alan Cox extensions.

The performance of the v0.17 ns83820 Linux kernel drivers can be seen in Figure 24. While the unmodified

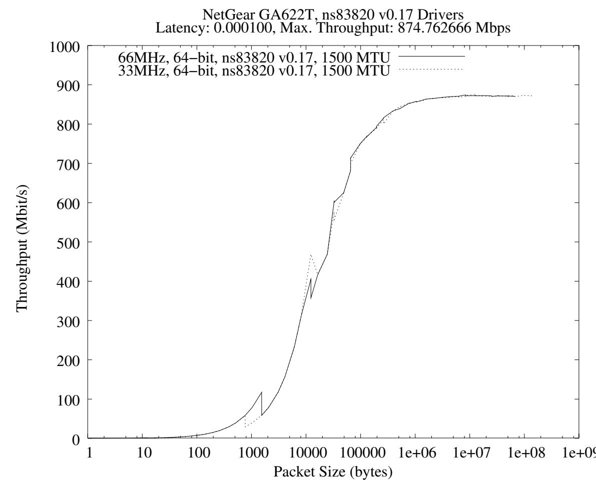

Figure 24. Improved performance in the ns83820 v0.17 drivers; shown here with the NetGear GA622T card.

v0.17 drivers supported only the 1500MTU run of our tests, compare Figure 24, which shows the NetGear GA622T being driven by the updated v0.17 ns83820 driver, with Figure 7, the same NetGear GA622T driven by the 2.4.17 Linux kernel's ns83820 driver. In addition, Figure 24 and Figure 8 show that the open source driver provides slightly better performance over the vendor-supported driver.

5.3. Socket Buffers

While more of an operating system and application tweak, the ns83820 driver has been shown to perform significantly better if one increases the default socket buffer size (64K in our environment). To put this performance attribute in the context of this report (driver performance), Figure 25 shows the performance resulting from a mixture of increasing the socket buffers to 512KB and decreasing the interrupt delay, as was discussed in Section 5.1. Note the

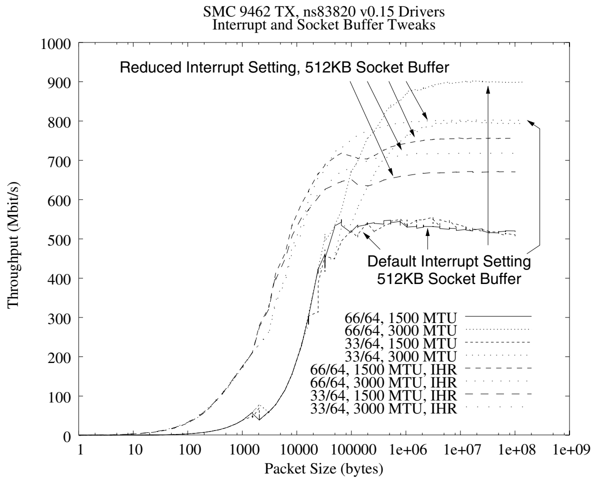

Figure 25. Reducing interrupt delays and increasing socket buffers (w/ SMC-9462T).

mixed results for performance when using increased socket buffers in conjunction with reduced interrupt delay settings.

5.4. Performance Enhancements for the 3C996BT

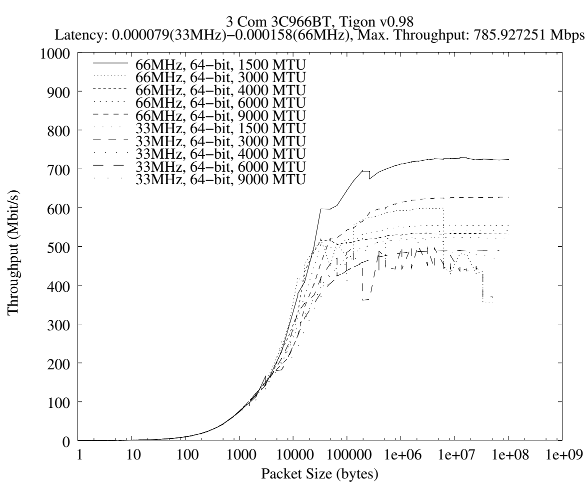

Figure 26. Performance for the 3Com 3C996BT using the Tigon v0.98 drivers

As was seen in Figures 9 and 18, increasing the frame size in the standard operating system environments *decreased* performance of the 3Com 3C996BT. Further, unlike the Intel 1000XT and ns83820-based cards under GNU/Linux or the SysKonnect SK98-21 under Windows 2000, adjustments to the interrupt delay (interrupt moderation) did not increase the card's performance.

This phenomena was not isolated to the bcm5700 drivers. New to the standard 2.4.19 Linux kernel are the *Tigon* drivers (tg3) for the Broadcom chipset. Figure 26 shows the performance of the Tigon drivers from the 2.4.19-pre9-ac2 kernel. Note that Figure 26 illustrates that the phenomena of *decreasing* performance with increasing MTU carries over to the Tigon drivers as well.

The decreased performance by increased MTU size indicated that a closer look a the socket buffer sizes was warranted. Figures 27 and 28 support this hypothesis. By increasing the size of the socket buffers to 1-Mbyte, the performance of the 3C996BT is "rectified." That is, by increasing the socket buffers specified in `/proc/sys/net/core/` from 64KB to 1MB, the 3C996BT displays increased performance with increased frame sizes, and exhibits overall superior performance.

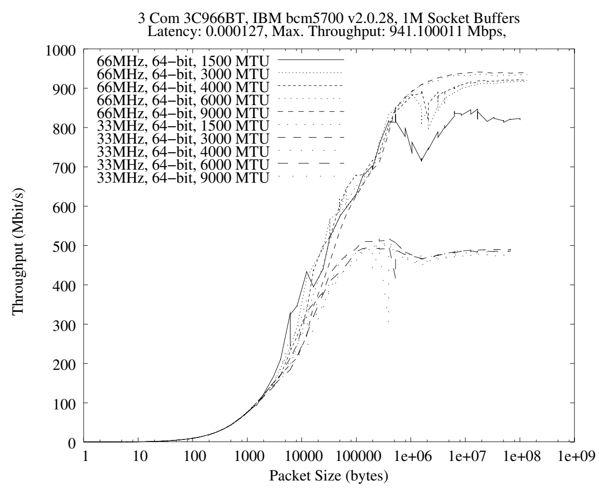

Figure 27. Performance for the 3C996BT: bcm57000 with 1Mb socket buffers.

One final observation from the experiments is the significantly different latencies observed in the Tigon drivers:78 μsec for a 33MHz bus, 158μsec for the 66MHz bus.

6. Related Work and Follow-up Investigations

While this report focuses on architectural, driver and TCP-related issues of performance, related benchmarks addressing the impact of the communication layer implementation (MPICH, LAMPI, PVM, and such) can be found in [7]. There are several groups working on performance

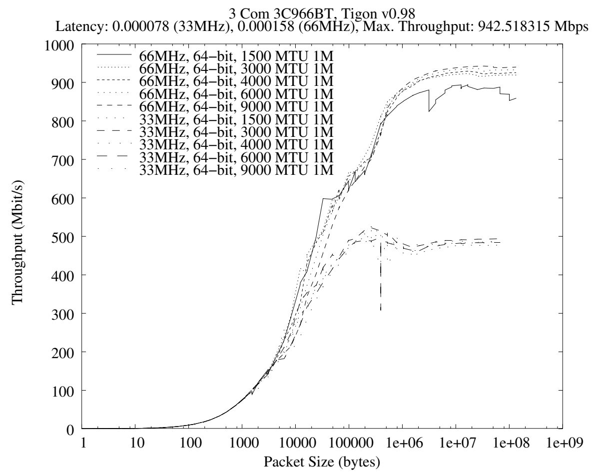

Figure 28. Performance for the 3C996BT: Tigon v0.98 with 1Mbyte socket buffers.

analysis associated with various fiber and copper solutions based upon the National Semiconductor dp83820 chipset. A portion of the work previously mentioned in [7] examines issues associated with tuning several communication libraries atop of TrendNet cards driven ns83820 drivers. Various package compilation options, header tweaks, DMA approaches, and socket options are discussed as they relate to overall performance of the communication package. The aforementioned work also addresses communication library performance results associated with the SysKonnect SK98. The performance of the fiber-based NetGear GA621, also driven by the National Semiconductor dp83820 chipset, is discussed in the context of the GAMMA project (the Genoa Active Message MAchine) in [1].

Ongoing investigations by the authors of this report include aspects of CPU utilization. One of the issues that arises when decreasing interrupt delays, as was depicted in Section 5, is the consequential utilization of the processor(s) for interrupt handling. This is of considerable concern in the area of compute clusters, where cpu utilization for communication would be at the expense of computation.

Another aspect that is being examined in conjunction with switching technologies is link aggregation, 802.3ad. To a certain degree, linear speedups in throughput can be observed in Fast Ethernet by aggregating multiple network cards into a single link. However, the aggregation of Gigabit Ethernet cards into a single link runs the risk of PCI bus over-saturation and marginal increases in throughput. The findings for aggregation of Gigabit Ethernet technologies for the cards presented in this report that support 802.3ad can be found in [3].

7. Conclusions

The findings of this report show that throughput, stability, and performance from copper-based GigE cards are intimately connected with judicious use and understanding of the corresponding driver(s) in GNU/Linux, and to a lesser extent in Windows 2000.

The influence of the driver can be drastically seen by examining, for example, the performance of the NetGear GA622T and the 3Com 3C996BT under various drivers and operating systems. The performance of the same NetGear GA622T is shown in Figures 7, 8, 17 and 24. Further, these performance graphs can be compared directly with Figure 25 – an SMC card with the same chipset as the NetGear. A card's dependence upon the configuration of the operating system is seen with the 3Com 3C996BT in Figures 9, 18, 26, 27, and 28. Throughput, stability, latency, and bus speed influences vary to extremes.

In addition, the results presented in this document show several counter-intuitive phenomena: that the range of throughput for GigE cards under a single driver is extremely broad; higher bus rates can have detrimental effects; larger MTU sizes can lead to poorer throughput; and combining driver optimizations can lead to sub-optimal performance. As a consequence, poorly-configured drivers and unoptimal frame sizes can result in remarkably sub-par performance whereas tuned drivers show a considerable payoff in low latency, high throughput, and stability.

References

[1] G. Ciaccio, M. Ehlert, and B. Schnor. Exploiting gigabit ethernet capacity for cluster applications. Submitted to High-Speed Local Networks, Nov. 2002, Tampa FL., Jun 2002.

[2] P. Gray and A. Betz. Gigabit over copper evaluation. Technical Report TR040202-CS, University of Northern Iowa, Department of Computer Science, Cedar Falls, IA, April 2002.

[3] P. Gray and X. Wu. System performance issues surrounding copper-based gigabit ethernet. Technical Report TR081502-CS, University of Northern Iowa, Department of Computer Science, Cedar Falls, IA, Nov 2002.

[4] Intel Corporation. Gigabit Ethernet over copper. An Intel white paper report, available at http://www.intel.com/network/connectivity/resources/doc_library/, Aug 2000.

[5] SysKonnect Corporation. White paper – Gigabit Ethernet. Available at http://www.syskonnect.com/syskonnect/technology, 1999.

[6] Q. O. Snell, A. R. Mikler, and J. L. Gustafson. Netpipe: A network protocol independent performance evaluator. In *Proceedings of the IASTED International Conference on Intelligent Information Management and Systems*, Jun 1996.

[7] D. Turner and X. Chen. Protocol-dependent message-passing performance on linux clusters. Submitted to Cluster 2002, Sept. 2002, Chicago IL., May 2002.