PERFORMANCE ANALYSIS OF NOKIA N800 PDA DEVICE

Aritra Mukherjee
Clemson University
Clemson,SC-29631
aritram@g.clemson.edu

Raunaq Nilekani
Clemson University
Clemson,SC-29631
rnileka@g.clemson.edu

ABSTRACT

It is known that TCP data transfer in a wireless multihop network experiences a degradation in inter-connection fairness and throughput. This is because TCP is designed for use in wired networks, and the wireless multihop network has characteristics of sharing of the medium resources among nodes, which wired networks do not have. In particular, in wireless multihop networks where wireless nodes widely exist, hidden/ exposed terminal problems are caused even if an RTS/CTS handshake is used. In this project paper, we have studied the traces involving the baseline TCP transfers on N800 PDA Device and Laptop and assess the relative performance of both the devices based on peak throughput, average bandwidth, number of packets transmitted and many more.The analysis has been documented in details supporting with the graphical representation of the performance.

INTRODUCTION

The Nokia N800 Internet Tablet is a wireless Internet appliance from Nokia. N800 allows the user to browse the Internet and communicate using Wi-Fi networks or with mobile phone via Bluetooth. It includes FM and Internet radio, an RSS news reader, image viewer and a media player for audio and video files.The N800, like all Nokia Internet Tablets, runs **Maemo**, which is similar to many handheld operating systems, and provides a "Home" screen—the central point from which all applications and settings are accessed. Maemo is a modified version of Debian GNU/Linux. The N800 is bundled with several applications including the Mozilla-based MicroB browser, Macromedia Flash, Gizmo, and Skype. It is compatible with any software designed for Maemo, and supports most common file formats.

For these types of applications, non-functional requirements such as end-to-end delay (latency) and the variation of the delay (jitter) must be taken into account. Usually multimedia applications use TCP and UDP as their transport protocol, but they may present many drawbacks regarding these non-functional requirements, and hence decrease the quality of the multimedia content being transmitted.

The performance of TCP in wireless networks can be done by studying the user-level and network-level performance for a mobile client browsing theWeb using a wireless Personal Digital Assistant (PDA). A wireless network analyzer is used to collect traces of the TCP/IP network traffic generated to and from the wireless PDA. In this project our analyses focus on server response time, Peak Throughput ,Total amount of bytes transferred per connection,Protocol breakdown in terms of Number of packets transmitted per second,Timestamp,The number of packets passed through the interface,The average size of each packet,The standard deviation of the packet size,The bandwidth in bits per second , TCP performance, and wireless channel quality. Today's mobile computing devices offer users an unprecedented level of connectivity and convenience. Many mobile users rely on their wireless laptop, Personal Digital Assistant (PDA), or cell phone for Internet access, with Web browsing as one of the primary applications. The widespread availability of WiFi hot spots (i.e., the IEEE 802.11b Wireless LAN standard) makes "anything, anytime, anywhere" information access a reality for many mobile Internet users. Wireless Web access, however, is not without its performance problems. Wireless channel bandwidth is often limited compared to desktop wired-Internet access, and the wireless channel is typically shared amongst multiple users. In addition, the wireless channel quality can vary significantly with time, with an inherent error rate much higher than that for wired network technologies. TCP is the primary transport-layer protocol in the Internet protocol stack, providing reliable data delivery for electronic mail, file transfer, and the Web. Some of TCP's control algorithms are arguably optimized for bulk data transfer on a wired network; TCP performance for Web browsing, particularly in a wireless environment, can be quite unpredictable. The performance of the Nokia N800 PDA device has been compared to the Laptop based on various criterias as explained above and the relative performance results of both has been documented in later parts of this report. Finally we realized that the performance of the Laptop is several times better than the Nokia N800 PDA device. The exact figures of the results and the graphical representation has been documented further down.

BACKGROUND

As Personal Digital Assistant (PDA) usage increases, so does the demand for better performance while adhering to strict power consumption limitations. As of the time of this study, the majority of PDA usage involves user-level applications which are not yet as computationally intensive as their desktop PC workstation counterparts. Because of that, it seems inappropriate to apply common benchmarks to these architectures, as it would not yield a particularly relevant analysis of how well the performance of these machines meets with current needs. Accordingly, new benchmarks specifically targeting current usage patterns must be devised. By analyzing the performance patterns and power usage footprint of codes that specifically target common user tasks and hardware subsystems (such as LCD display, audio, etc) one can isolate areas of weakness in present architectures.

MOTIVATION AND OBJECTIVE

Currently, we have interested in transmission of data including multimedia data over a wireless network when Personal Digital Assistant (PDA) is used as mobile host(MH) instead of Laptop PC.Much of the work have been focused on the performance analysis of data transmission over a wireless network using laptop as MH.We think that PDA has advantage over Laptop in the point of mobililty due to its smaller size .But PDA is inferior to Laptop in the overall performance.Thus,in this project we are focussing on the performance analysis of data transmission over a wireless network when we are using PDA as MH.
So far, Various experiments have been conducted to measure and analyze the performance of transmission of media/data with PDA as mobile host over an infrastr ucture network.Our test-bed network consists of the infrastucture network integrating a wireless network with wired network.For an analysis of the performance ,we have measured the transferring of data from Nokia PDA to a desktop server or viceversa over the wireless network by changing the size of the packet.As the size of the packet becomes larger,the time elapsed to transfer the data from PDA to the server or viceversa becomes longer.

METHODOLOGY

1. Approach
Based on usage patterns on N800 PDA Device,we chose to isolate several areas to examine for performance. Because of the current niche that the portable device N800 device that we are examining hold, it is imperative that they be capable of relatively fine-grained graphics. Consumers expect a high-level graphical user interface (GUI) and most importantly a very good network performance while transmitting data/multimedia data over wireless networks.
We have used the iperf tool which is a performance measurement tool, to perform the baseline transfers on N800 PDA device and laptop.Here we are running a desktop machine as the server , PDA and laptops as clients respectively to examine the data transmission over a wireless network.Additionally we are running tcpdump at the client side to capture the traffic.This process is repeated for laptop as well as N800 PDA Device.Once we had sufficient dump files to examine,we employed various scripts,networking tools to study the charactersics of PDA/Laptop and assessed the relative performance.The key parameters used to study the behavior of the
Transmission contral protocol are as follows: 1) Average Throughput 2) Peak Throughput 3) Loss packet rate 4) standard deviation of packet size 5) interface load 6) Average packet size

Experiment 1:

Server: Darkhelmet

DarkHelmet (a linux machine) with internet protocol address 130.127.48.87

Client: Nokia PDA Device

Nokia N800 device with the internet protocol Addess 172.22.7.89

On the Nokia N800 Device, which is acting as the client, run the following command:

iperf -c 130.127.48.87 -i 1 -t 20 -p 5001 -w 900k

On the server side i.e Darkhelmet, run the following command:

iperf -p 5001 -w 900k -s -i 1

Also, on the client side i.e Nokia N800 we are running the tcpdump to capture the data transferred between the PDA and Darkhelmet.we ran the following command

tcpdump -s 1500 -w NokiaSample1.dmp

here, we were substituting different values for s varying from 300,900 1200 etc and filtering the required data using the command

tcpdump -v -n -s 60 -i wlan0 -w NokiaSample2$1

Steps:

Step 1: We first started the tcpdump on the client(Nokia PDA)

Step 2: Then we ran the iperf command on the server side (darkHelmet)

Step 3: Finally,iperf command on the client side (Nokia N800)

Step 4: After sometime we stopped the tcpdump using ctrl+c

Step 5: Then using the dump file as an input, we used the tcptrace tool to study the performance of the Nokia PDA device.

Command used for tcptrace:

tcptrace -n -l NokiaSample1.dmp > NokiaSample1.tcptrace

Step 6: While studying the traces, the parameter employed by us to assess the performance of PDA was the Throughput obtained on the Client side and drop packet rate.

Step 7: The throughput obtained from the tcptrace was cross verified by studying the traces using the following commands:

tcpdump -X -n -tt -r NokiaSample.dmp > Nokiasample.txt

From the traces obtained from the above command we used the following script to extract the relevant data :

awk '/IP/{print $1}' Nokiasample1.txt > extractedtime.txt

then we used the following script to compute the difference of time and then the following script to compute the Bandwidth of the PDA Device **ScriptToExtractTime.sh** && **ScriptToComputeBandwidth.sh**
The loss packet rate was computed using the script packetLoss.awk

Using **Tcpstat**, an important networking tool that reports certain network interface statistics. It gets its information by either monitoring a specific interface or by reading previously saved tcpdump data from a file.

The tcpstat computes the following statistics :

Bandwidth
Number of packets
Average packet size
Standard deviation of packet size
Interface load

Now running the following command for Nokia PDA Device and Laptop :

Tcpstat –r nokiatrace1.dmp
Tcpstat –r computer1.dmp

We obtain following statistics :

- **timestamp**
- **the number of packets passed through the interface**
- **the average size of each packet**
- **the standard deviation of the packet size**
- **the bandwidth in bits per second.**

Then, we let tcpstat run over the data and output the data into different files for Nokia PDA device

```
tcpstat -r nokiatrace1.dmp -o "%R\t%A\n" 0.08 > arp.data
tcpstat -r nokiatrace1.dmp -o "%R\t%C\n" 0.08  > icmp.data
tcpstat -r nokiatrace1.dmp -o "%R\t%T\n" 0.08  > tcp.data
tcpstat -r nokiatrace1.dmp -o "%R\t%U\n" 0.08  > udp.data
```

The data produced by tcpstat can also be graphed using gnuplot.

This is how we run the command to plot the graph

```
gnuplot gnuplot.script > plot1.png
```

Then we have made a script named **peakthroughput.pl** that calculates the total amount of bytes transferred per connection and also the peak throughput (the peak throughput is the maximum throughput consumed by the connection for a 5 second time period).

2. Tools:

The following tools were used to monitor,capture and assess the relative performance of laptop and Nokia N800 PDA

1) **tcpdump**

   tcpdump is a common packet sniffer that runs under the command line. It allows the user to intercept and display TCP/IP and other packets being transmitted or received over a network to which the computer is attached

2) **iperf**

   Iperf is a commonly used network testing tool that can create TCP and UDP data streams and measure the throughput of a network that is carrying them. Iperf is a modern tool for network performance measurement written in C++.

3) **tcptrace**

   Using Tcpstat, an important networking tool that reports certain network interface statistics. It gets its information by either monitoring a specific interface or by reading previously saved tcpdump data from a file.

4) **gnuplot**

   It is a portable command-line driven interactive data and function plotting utility for UNIX, IBM OS/2, MS Windows

5) **tcpstat**

   tcpstat reports certain network interface statistics much like vmstat does for system statistics. tcpstat gets its information by either monitoring a specific interface, or by reading previously saved tcpdump data from a file.

6) **perl** and **awk** scripts attached in the Code documentation section

ANALYSIS

Using the methodologies explained we have done the relative performance analysis of the Nokia N800 PDA device and the Laptop.

Throughput and Packet Loss

Experiment 1 performed with **Nokia PDA Device** as documented above, we got the following results :

Throughput of Nokia PDA: 397011 Bps

Loss packet Rate: 4 percent (Approximately)

Experiment 2 performed with **Laptop** as documented above, we got the following results:

Throughput of Laptop: 11652388 Bps

Loss packet Rate: 1-2 percent (Approximately)...Note: Varying for different sample traces

In communication networks **throughput** is the average rate of successful message delivery over a communication channel. This data may be delivered over a physical or logical link, over a wireless channel, or that is passing through a certain network node, such as data passed between two specific computers. The throughput is usually measured in bits per second (bit/s or bps), and sometimes in data packets per second or data packets per time slot.

The factors that mainly affects throughput are :

- Flow control, for example in the Transmission Control Protocol (TCP) protocol, affects the throughput if the Bandwidth-delay product is larger than the TCP window, i.e. the buffer size. In that case the sending computer must wait for acknowledgement of the data packets before it can send more packets.

- TCP congestion avoidance controls the data rate. So called "slow start" occurs in the beginning of a file transfer, and after packet drops caused by router congestion or bit errors in for example wireless links.

By performing the above 2 experiments we analyzed that the throughput of Nokia N800 was only 397011Bps compared to the Laptop whose throughput was 11652388Bps. The throughput obtained with respect to Laptop is approximately **29 times** better than Nokia N800 PDA Device.

The number of packets dropped during the transmission was only 1-2% approximately in the Laptop compared to 4% in the Nokia N800 PDA device.

Statistics based on standard deviation of the packet size, bandwidth number of packets passed through the interface average size of each packet

After running the following command for Nokia N800 PDA Device :

Tcpstat –r nokiatrace1.dmp

We get the following output

Time:1226499213	**n=12**	**avg=81.33**	**stddev=33.84**	**bps=1561.60**
Time:1226499218	**n=556**	**avg=889.97**	**stddev=702.42**	**bps=791721.60**
Time:1226499223	**n=78**	**avg=477.13**	**stddev=633.63**	**bps=59545.60**

These statistics are (from left to right):

*** timestamp**
*** the number of packets passed through the interface**
*** the average size of each packet**
*** the standard deviation of the packet size**
*** the bandwidth in bits per second.**

Now running the following command for Laptop :

Tcpstat –r computer1.dmp

We get the following output

Time:1226607020	**n=8577**	**avg=2654.72**	**stddev=11422.42**	**bps=36431238.40**
Time:1226607025	**n=2229**	**avg=2659.76**	**stddev=11361.50**	**bps=94895864.00**
Time:1226607030	**n=2234**	**avg=2669.32**	**stddev=11436.43**	**bps=95450550.40**
Time:1226607035	**n=22361**	**avg=2657.12**	**stddev=11344.48**	**bps=95065241.60**
Time:1226607040	**n=1707**	**avg=2656.90**	**stddev=11546.29**	**bps=7256513.60**

From these results we have analyzed that the maximum number of packets passed through the interface at a particular time for **Nokia N800 was 556**, whereas in case of **Laptop this was 22361**, which was significantly very high. The average size of each packet in a particular time in case of Nokia N800 was only **889.97** as compared to Laptop which was **2669.32**. The standard deviation of the packet size was also very high in Laptop than the Nokia PDA.

Bandwidth

Bandwidth in computer networking refers to the data rate supported by a network connection or interface. One most commonly expresses bandwidth in terms of bits per second (bps). Bandwidth represents the capacity of the connection. The greater the capacity, the more likely that greater performance will follow. The bandwidth of the **Nokia N800 PDA device was 791721.60 Bps** whereas in case of **Laptop it was 95450550.40 Bps**.This clearly evident that Laptop has far better performance as compared to Nokia PDA device.

Total Data Bytes and Peak Throughput

Running the script named peakthroughput.plthat calculates the total amount of bytes transferred per connection and also the peak throughput for the **Nokia PDA** device, the following datas has been gathered :

connDetails 1 :
Connection number = 22 total Data Bytes = 1760 bytes

connDetails 2 :
Connection number = 60973 total Data Bytes = 495032 bytes

The Peak throughput for connDetails 2 is = 394876.8 bps

The Peak throughput for connDetails 1 is = 972.8 bps

Running the same script for the **Laptop** , we got the following datas :

connDetails 1 :
Connection Number = 56842 total Data Bytes = 201664684 bytes

The Peak throughput is = 46709836.8 bps

The peak throughput is the maximum throughput consumed by the connection for a 5 second time period, as considered in our case. The Nokia PDA has the peak throughput of only **394876.8 bps but the Laptop's peak throughput was 46709836.8 bps.** This significantly proves that the Laptop has anyday a better performance that the Nokia N800 PDA.

The total Data Bytes sent by the Nokia PDA for a particular connection was 495032 bytes.
But in case of the Laptop the total Data Bytes sent was 201664684 bytes, which was far more than the PDA device.

Graphical Representation

The script gnuplot2.script has been used to plot the graphs for the Protocol breakdown representation :

After running the command

gnuplot gnuplot2.script > GraphforN80andLaptop.png

the plotted graph looks as follows :

From the protocol breakdown, we could analyze that in case of Laptop it sends 4500packets/sec and its quite consistent throughout 100 seconds of transmission but in case of Nokia PDA its only 1000packets/sec and that too it reduces drastically in the following seconds of transmission.
It is quite evident from the graph plotted above, for transmission of data over wireless network, the behavior of Transmission control protocol in Laptops is far better than the performance of Nokia N800 PDA Device.

CONCLUSION/DISCUSSION

The aim of this project was to assess the performance of Nokia PDA device and Laptop also explore the possibility of creating a standard toolkit for benchmarking and evaluating PDA performance and to use those tools to evaluate PDA samples. The approach to move away from standard scientific benchmark sets and to more accurately simulate real user code was valid and promising and tying those results into architectural details and weaknesses.We were able to develop several good examples of the sorts of generic cross-platform metrics that we originally set out to do.However, we feel that with the toolset that we have developed, and the methodology outlined above and adequate

samples, that one can begin to tackle the questions that we set forth to answer.Furthermore,various experiments(in the TCP stack of N800 PDA) can be conducted to improve the performance of N800 PDA device and make it as efficient as Laptops.

The performance of Laptop is far better than the performance of PDA in transmitting data/multimedia data over an infrastructure network.

The weakness lies in the traditional benchmark set for evaluating the performance of PDA with respect to a laptop.

RELATED WORK

There has been little work into benchmarking PDAs. The performance studies that have been carried out to date have been case studies focusing on a particular architecture and testing its performance and/or power consumption. By standardizing our codes, we seek to be able to use our results to do a comparative analysis across various architectures.

CODE DOCUMENTATION

ScriptToExtractTime.sh was being used to extract the time for the Laptop and Nokia N800 PDA device.

ScriptToExtractTime.sh

```
while read line
    do

    F1=$(echo $line | cut -d : -f1)
    echo "$F1 -$F2" |bc
   F2=$F1

   done < extractedtime.txt
```

ScriptToComputeBandwidth.**sh** was being used to compute the bandwidth from the tcpdump files for the Laptop and Nokia N800 PDA device .

ScriptToComputeBandwidth.sh

```
while read line
    do
   F1=$(echo $line | cut -d: -f1)
   echo "1448/$F1"  | bc
   F2=$F1

   done < timediff.txt
```

PacketLoss.awk script has been used to calculate the packet loss rate for Nokia PDA and Laptop

```
 BEGIN {


  # Initialization. Set two variables. fsDrops: packets drop. numFs: pa        ckets sent

   fsDrops = 0;
   numFs = 0;

   }


{
 action = $11;
drop = $14;
to = $4;
type = $5;

pktsize = $6;
flow_id = $8;
src = $9;

dst = $10;
seq_no = $11;

packet_id = $12;

if (action == "46")
```

```
numFs++;
if ( drop == "1")

fsDrops++;

}
END {
drop_rate        =(fsDrops/numFs)*100;

printf("number of packets sent:%d lost:%d\n", numFs, fsDrops);

printf("\ndrop rate=%f",drop_rate);

}
```

The **peakthroughput.pl** file has been used to calculate total data bytes sent per connection and the peak throughput per connection based on 5seconds time period.

```
#!/usr/bin/perl -w
use strict;


my $inputFile = 'computer1.txt';
open( IN, "$inputFile" );

my @line   = <IN>;
my $count = @line;

my @connDetails;

for ( my $k = 0 ; $k < 100 ; $k++ ) {
	for (my  $j = 0 ; $j < 2 ; $j++ ) {
		$connDetails[$k][$j] = 0;

	}
}


my $k = 0;
my $cnt = 0;
my $bool = 0;
my $port=0;
my $bytes=0;

for (my $i=0 ; $i < $count; $i++){

	$bool = 0;
	if($line[$i] =~ /(130.127.49.110)+\.+(\d+)+/){
		$port = $2;

		if($line[$i] =~ /\(+(\d+)+\)+/){

			$bytes = $1;
```

```
				for(my $z = 0; $z < $cnt ; $z++){
					if($port == $connDetails[$z][0]){

						$connDetails[$z][1] = $connDetails[$z][1] + $bytes;
						$bool = 1;

					}
				}
				if($bool == 0){
					$connDetails[$cnt][0] = $port;

					$connDetails[$cnt][1] = $bytes;

					$cnt ++ ;
				}

			}


		}

}


for(my $m = 0; $m < $cnt; $m ++){
	for(my $n = 0; $n < ($cnt-1); $n++){
		if($connDetails[$n][1] > $connDetails[$n+1][1]){
			my $temp = $connDetails[$n+1][1];
			my $temp1 = $connDetails[$n+1][0];
			$connDetails[$n+1][1] = $connDetails[$n][1];
			$connDetails[$n+1][0] = $connDetails[$n][0];
			$connDetails[$n][1] = $temp;

			$connDetails[$n][0] = $temp1;
		}
	}


}



for(my $k = 0; $k < $cnt ; $k++){
	for(my $j = 0; $j < 2; $j++){
		print "connDetails[$k][$j] = $connDetails[$k][$j]\t";
	}
	print "\n";
}

my $init	= 0;
my $peakThroughput	= 0;
my $data=0;
my $time=0;
my $sum=0;
my $cntTime=0;
```

```
for ( my $i = 0 ; $i < $count ; $i++ ) {
	if ( $line[$i] =~ /\s+(130.127.49.110.56842)+/ ) {
		if ( $line[$i] =~ /\(+(\d+)\)/ ) {
			$data = $1;
			if ( ( $init == 0 ) && ( $line[$i] =~ /^(\d+\.+\d+)/ ) ) {
				$time     = $1;
				$sum = $data;
				$init = 1;
			}
			else {
				if ( ( $line[$i] =~ /^(\d+\.+\d+)/ ) ) {
					$cntTime = $1;
					if ( $cntTime < ( $time + 5 ) ) {
						$sum = $sum + $data;
					}
					else {
						$init = 0;
						if ( $sum > $peakThroughput ) {
							$peakThroughput = $sum;
						}

					}
				}
			}
		}
	}
}
$peakThroughput = $peakThroughput* 0.8;
print " The Peak throughput is = $peakThroughput bps\n";
```

The script **gnuplot2.script** was being used to plot the graphs for protocol breakdown representation

```
 set term png

set data style lines
set grid
set yrange [ -10 :  ]
set title "Performance Analysis of N800 PDA and laptop based on protocol breakdown"
set xlabel "seconds"
set ylabel "packets/s"
plot 	"Ntcp.data" using 1:($2/0.08) smooth csplines title "TCP-N800 PDA" \
  ,"Ltcp.data" using 1:($2/0.08) smooth csplines title "TCP-Laptop"
```