On-line storage versus local storage for mobile users

HUANG LIANG

KTH Information and Communication Technology

Master of Science Thesis
Stockholm, Sweden 2006

COS/CCS 2006-8

On-line storage versus local storage for mobile users

Huang Liang

10 April 2006

Wireless@KTH

Supervisor: Gerald Q. Maguire Jr.
Examiner: Gerald Q. Maguire Jr.

School of Information and Communication Technology (ICT)

Royal Institute of Technology (KTH)
Stockholm, Sweden

Abstract

When a user has a mobile device with lots of built-in functions, what would they like to do with it? Of course, interactive voice and videoconferencing, sending SMS & Instant Messaging, listening to music, taking photos, etc. People want to have a device with a large storage capacity, much as they do on a desktop or laptop PC. But sometimes the user does not have sufficient local storage capacity on their mobile device. Online storage is a good solution for this, but the limited battery capacity connectivity must be balanced such require that the mobile decided what should be uploaded/downloaded and when - along with what should be stored locally.

This problem is very significant not only theoretically, but also practically. We expect that the online storage will replace storage media, such as CDs and DVDs. Today use of a mobile device is a very popular. Users would like to be able to easily send files to friends in other parts of the work, and share files with these friends. Additionally, users to not want to loose important data (photos, files, ...), these functions can all be implemented using on-line storage. Use on-line storage should be simpler for the user, thus smart mobile devices should simplify the user's experience, provide safer file storage (i.e., with a lower risk of data loss), and to store files in the most appropriate location(s).

Sammanfattning

När har en användare en mobil apparat med raddor som builten-in fungerar, vad skade dem gillar med den? Naturligtvis växelverkande uttrycka, och videoconferencingen och att överföra SMS & ögonblickMessaging och att lyssna till musik som tar foto, Etc.-folk, önskar att ha en apparat med en stor lagringskapacitet, mycket, som de gör på en skrivbords- eller laptopPC. Men ibland har användaren inte tillräcklig kapacitet för lokal lagring på deras mobila apparat. Direktanslutet är lagring en god lösning för denna, men den inskränkt batterikapacitetsconnectivityen måste vara balanserat sådan kräver att det mobilt avgjort vad bör uploadeds/nedladdas och när - tillsammans med vad bör lagras lokalt.

Detta problem är mycket viktigt inte endast teoretiskt, utan också praktiskt. Vi förväntar att lagringen skar direktanslutet byter ut lagringsmassmedia, liksom CDs och DVDs. I dag är bruk av en mobil apparat ett mycket populärt. Användare skade något liknande överför lätt sparar till vänner i annan begåvning av arbetet och delar sparar med dessa vänner. Dessutom fungerar användare som ska inte önskas att lossa viktiga data (foto, sparar,…), dessa kan alla genomföras genom att använda on-line lagring. Bör on-line lagring för bruk vara enklare för användaren, således bör smart mobila apparater förenkla användaren erfar, ger säkrare sparar lagring (, med ett lägre riskera dvs. av dataförlust), och att lagra sparar i mest anslår lägen.

TABLE OF CONTENTS

Acknowledgements

First of all I would like to express my most sincere gratitude to my project advisor, Professor Gerald Q. Maguire Jr., for helping me when I needed, encouraging me when problems rose, answering all my doubts, support me with equipments, and being always willing to transmit his positivism and share his knowledge.

I would like also to thank all my colleagues at the lab for maintaining such a good atmosphere that made work easier and more comfortable. Especially thank help from Pietro Lungaro.

All my friends need to be mentioned here, those who were encouraging me from China and those who were here, thanks to all them for supporting me in bad moments and make me feel much better when it was necessary.

Finally, thanks the Wireleess@KTH and thanks to KTH.

List of figures and tables

1. Introduction

Mobile users will increacingly expect that anything they can do on a desktop computer can be done on their mobile device. Processing performance will continue to get better and better. However, today storage capacity is a bottleneck, due to both the limited capacity and limited security. Current secure digital (SD) memory cards for mobile devices offer upto 2Gbyte of storage capacity. Such SD memory cards can be used in a wide variety of digital equipment and supports many application formats. It does not even require the use of a PC, while providing speedy transfers of high-volumes large amount of content and offering content protection for recordable media. Despite these benefits, such devices should only be seen as temporary storage for data, since due to its per Mbyte costs it is unsuitable for permanent storage.

In a tomorrow's fully networked society, users would not take a storgage device with them, they only need a mobile device to remotely access their online storage space. When a student wants to download a document from a computer server, they simply transfer it to their online storage space. When a teacher ask his or her students to hand in their homework or a test, he or she simply assigns each students an file ID and the student can directly store their document in the teacher's storage space. Thus students need not use flash memory or e-mail to transfer such documents. Consider a reporter who wants to quickly submit a breaking news story. As there is high value in publishing this news (which may include photos, audio, video clips, …) quickly, the reporter wishes to upload their report to the publishing company both quickly and successfully. As the system should handle the transfers automatically, the only requirement is for the report to grant the publishing company access to this report. Since the report is only cached locally, once a copy has been placed into a suitable on-line storage service the reporter does not have to worry about either data loss or how much storage capacity they will need (the later is because of the very rapid decline is mass storage prices; while the former is due to the easy of replicating the data and the low cost of storage for these additional copies).

Today we do not have a fully networked society, so we must deal with the problems of intermittent connectivity.Therefor, this thesis focuses on how to choose what files to upload/download and how to exploit intermittent connectivity. This requires finding and intelligently selecting both files (objects) and the appropriate access point(AP) to utilize. It include implementing and evaluating an application which runs on the mobile device. Additionally, many mobile devices are equipped with multiple wireless interfaces, thus the device must decide when it should turn on which wireless interface, then it should decide which files it should upload or download. The device must estimate the current network condition (available around connectivity in its current location) and calculates the probability of successfully transferring via the (wireless) network then selected the item(s). Because limited battery capacity is also a problem for the mobile device, the software must find a good balance between turning on the wireless interface to perform checks for connectivity & making transfers and likely future demands upon the battery.

The initial step in developing such as application is to collected data indicating time, location, and potential connectivity which the user is likely to experience in a simple database. This data (collected in Kista, Sweden) will be used to generate probabilitic models which can later be used to reason about when and where to power on the wireless interface, the probability of having connectivity if the user's current location is a given location or if they are moving in a given direction from this location, … . From these experimentations I derive the probability of accessing a suitable AP in Kista and derive a model which describes the probability of access via a hotspot as a cumulative distribution function. If the model calculates that the wireless connection should be turned on and if there is potential connectivity via more than one AP at the same time at a given location, then the device will choose the most suitable one AP to associate with. This might be based on price, predicted bandwidth, etc. Once the device has selected a suitable AP (and potentially authenticated and been granted authorization to use this AP), then the software will schedule the transfer of the items according to some priority list and based upon its estimate of considering the current conditions of the network environment. For example, if the mobile device is expected to be able to continue to access this AP for some time and the network currently offers lots of bandwidth, then it might choose to download a 50 Mbyte file even though it has a lower priority than a smaller file - so as to exploit the currently available high bandwidth. Otherwise, it might choose to transfer the files in priority order, but fetch the larger file in such a way that if the transfer is not completed that the transfer can be continued when connectivity is next available.

Although users might want internet access wherever they are and at anytime, this is extremely expensive[1]. Instead, this work will focus on “spotty coverage” [2] - where connectivity is only available in some places at some times. Because few people need network access in the forest in the mountains, it is not cost effective for an operator to provide coverage in places where there is a low probability of a user. Conversely in places where there are lots of people, there will be a high probability of one or more operators providing connectivity. To make use of this, the user’s device needs to forecast the probability of having (wireless) network access. Today, many vendors are developing online storage service (see for example, ByteTaxi Inc.’s FolderShare - https://www.foldershare.com/). These companies provide users with on-line storage space. Such a service allows you to access your files from any internet connected device. Thus making it easy to share your files quickly and easily, whether they contain video, audio, images, or other data. Such serivce can also provide file backup (i..e, safely and reliably). These services allow the user to use either a browser or client application to interact with the service. Today this is still a new service for mobile users. Although distributed files systems - including those for “disconnected” operation have been a research topic for several decades (see for example, Coda http://www.coda.cs.cmu.edu/ and http://citeseer.ist.psu.edu/kistler92disconnected.html) in this thesis. The Coda distributed file system have had a solution for the disconnected operation, it use CML [3] to help solve the disconnected problem when transfering the files. But the Pocket PC have two limits, one is the battery, the other is the storage, so these make us do not turn the wireless adaptor on all the time and do not make us to use the CML solution. We need the application can help Pocket PC to distribute the storage and the battery. We will focus on using an application to provide the mobile device with intelligent behavior - allowing the application to manage both the online storage and local storage.

Such intelligent behavior was a theme in the a project on Adaptive & Context-Aware Services[4]. For mobile users, wireless connectivity is an element of their context, as the signal strength and available bandwidth change when the user moves betweens different areas or when other people or objects move. These changes in link properties may lead to handoffs *within* a single network or *roaming between* different networks. While Mobile IP [5] can be used to maintain a single IP address despite roaming, it is not clear if this is necessary when performing transfers to/from an online storage service (since each session could utilize the current temporary IP address). However, the use of Mobile IP together with VPNs may be useful to avoid the need to rake the VPN connection because of the change in network attachment point. Thus a clear question in this thesis is “How long is a transfer likely to take”?

Rather than simply triggering one of two behaviors scheduling (as would be the case with Coda - i.e., connected or disconnected), this work attempts to go farther by scheduling transfers based on information about the current and predicted communication conditions. This work thus extends the earlier work of both Inmaculada Rangel Vacas [6] and Maria José Parajon Dominguez [7]. An important part of this work will be to find a model to describe the metrics to be used to decide which files should be transferred (i.e., uploaded or downloaded).

The following figure shows the general network architecture which we consider.

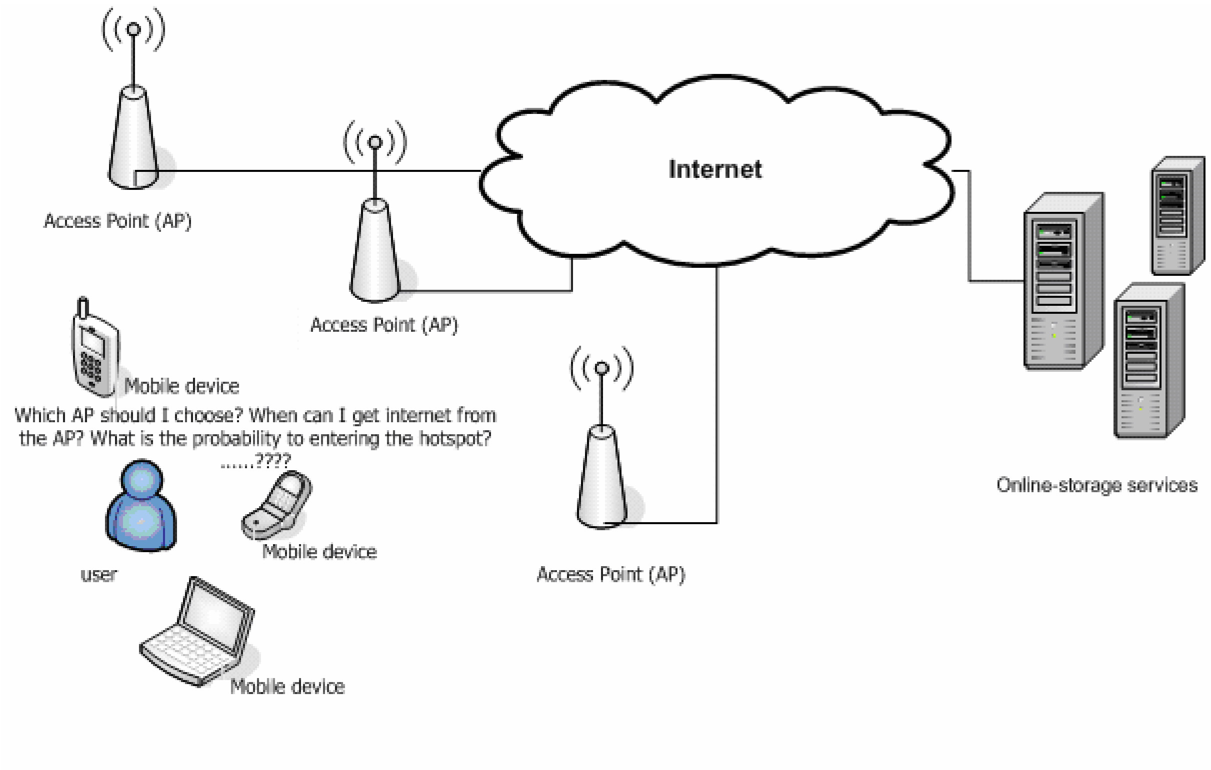

Figure 1: Overview of online-storage for mobile users

2. Background

Online storage technology has existed for many years. It has been particularily significant for small or medium sized enterprises (see for example firmx, firmy, and firmz). In addition, some companies have focused on providing such services to individuals (see for example Microsoft, Streamload, and Novell).

Currently fixed users typically do not have trouble storing and sending large files as they increasingly have "always on" broadband connectivity. If you want to share your home movies with your friends or relatives across town or overseas, but don't want to clog up their email inboxes, you can use an online storage service. Later others can download your movie based on a link that you sent to them (via e-mail, instant messanger, etc.) or they can watch them by streaming them from the online storage service. Similarily if you have taken a great high resolution photo and you need to get it to your production department on other side of the world in a hurry, but the file is too big to attach to your e-mail - your online storage service provider comes to the rescue! Simply upload the digital photo to their server and the photo can be downloaded almost immediately from anywhere on the internet. Similarily online storage can help you share your mp3s and other digital music with your friends or you can even download it yourself to listen to it directly or to store it into your mobile device. (Note that here we are not considering the legal issues surrounding digital rights, but assume that there is another infrastructure which provides this control and instead focus simply on the storage and transfer of objects - thus the control of access to the contents of these objects is orthogonal to this thesis work.)

Similar to online storage service providers, there are also application service providers who provide CPU cycles to users. So if you have developed a new software package, but your current host does not provide you with enough bandwidth to serve all your customers, you can host your program at a hosting service. The popularity of such on-line service provisioning is very apparent in the recent Microsoft annoucements of Windows Live, Microsoft Office Live, Windows Live Messenger [8].

The iPAQ 5550 Pocket PC running Microsoft’s Pocket PC 2003 operating system will be used in this project. The iPAQ 5550 Pocket PC is a mature Pocket PC product, the Microsoft’s Pocket PC operating system is a steady operating system. The wireless support and the Application Programming Interface (API) will help us to implement wireless signal survey, battery monitor, and storage card monitor. The API will help us to develop the application easier and faster. In the other side the Microsoft’s Pocket PC operating system also have a good market on mobile devices, the iPAQ is one and the typical product.

2.1 (Wireless) Network Access

In order to be able to access the remote online storage the mobile device must have network connectivity and the connectionty should be good enough. These depends on a number of factors:

- a suitable network interface and the presence of a compatible access point
- sufficient battery power to perform the operations necessary
- authorization to utilize this access point

- the available bandwidth of this connection
- the signal strength
- the distance of this access point range of a wireless network
- the number of the devices using this single access point
- the roaming technology if there is a need to roam

2.2 Battery power

Our application can find out the amount of battery power left by using the interface xxxx. Microsoft 's Windows CE platform provides a number of functions for managing battery. The Windows CE getSystempowerstatusEx2() can be called to return a SYSTEM_POWER_STATUS_EX2 structure. This SYSTEM_POWER_STATUS_EX2 structure contains information about the power supply of the system. There are many members can describe different informations aspect of the power: the battery level, voltage, estimate operating, etc.

In this project, we will need these members as follow:

ACLineStatus:

Value	Meaning
0	offline
1	online
255	unknown

BatteryFlag:

Value	Meaning
1	High—the battery capacity is at more than 66 percent
2	Low—the battery capacity is at less than 33 percent
4	Critical—the battery capacity is at less than five percent
8	Charging
128	No system battery
255	Unknown status—unable to read the battery flag information

BatteryLifePercent:
The percentage of a full battery charge remaining. The range is from 0 to 100 or unknown, if the status is unknown.

BatteryLifeTime:
How many seconds of battery life remain at the current utilization rate, or –1 if remaining seconds are unknown.

BatteryVoltage:
Amount of battery voltage in millivolts, This member can have a value in the range of 0 to 65,535.

To use the getSystempowerstatusEx2() function, we need to use the Microsoft pinvoke mechanism to call a function in a dynamically loaded library. Hence we must import coredll first, then use the getSystempowerstatusEx2 function.

2.3 Turning a (wireless) interface on and off

Fundamental to saving battery power is to turn off an interface when it is not needed. Microsoft has defined methods to turn on and turn off an interface either immediately or at some specific time. To perform these operation on HP iPAQ when running Pocket PC you must to use pinvoke to load the liberary iPAQUtil.dll first. Then you can call the function iPAQSetWLANRadio(UInt32 flag), where the instance of flag will be set 0 or 1; 0 is on, 1 is off. This controls the wireless network adapter of the Pocket PC by managing its power.

2.4 Detecting a (Wireless) Network Access Point

Assuming that you have turned on your (wireless) network interface, the next task is to to find a suitable wireless network. Fortunately, there exist functions in the .NET compact framework [9] to support wireless network client configuration. Microsoft .NET CF's Automatic wireless network configuration supports the IEEE 802.11 standard [10] and was designed to minimize the effort required to configure access to a wireless network.

The Microsoft Windows CE device driver interface control the IEEE 802.11 network interface card. The network user interface (UI) uses the WZCSAPI [11] interface to retrieve and set the relevant 802.11 parameter(s). The status monitor also uses WZCSAPI to get parameters such as signal strength, SSID, and MAC address. By using NDIS User I/O (NDISUIO)[11], the Network Driver Interface Specification(NDIS)[11] protocol driver exposes a generic interface for sending requests and receiving status from NDIS miniport drivers. The Windows API and NDIS API will enable us to get the desired wireless card information including the signal strength, the IP address and MAC dress of the wireless card have, and some other detailed information we want. We can even turn on and off the wireless card by setting some parameters. These APIs provide the interfaces we need to communicate with the underlying hardware.

The automatic wireless network configuration daemon configures the network interface when the device roams across from one wireless network to another, i.e., without any requiring the user to manually specifying connection settings for each local network. When the device moves from one location to the other, the automatic wireless configuration daemon searches for all the available wireless networks that the device can hear. Then it notifies the device when there is a new network which is availble and the device can connect to it. Generally, the device will select the network with the strongest signal strength to connect, but it also can select a specific network to which the user hopes to connect. Once the wireless network is selected, the configuration software updates the wireless network adapter to match the settings of that wireless network, then it will attempt to connect to that wireless network.

Figure 2 shows an overview of the components of the Windows CE 802.11 Automatic Configuration subsystem.

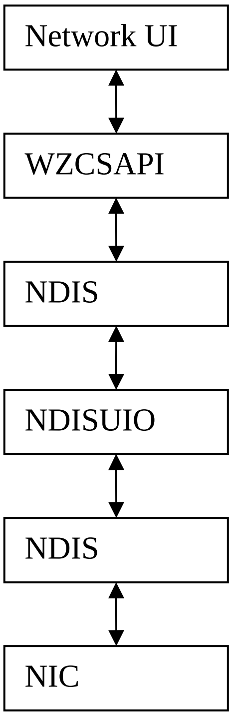

Figure 2 : overview of WinCE 802.11 configuration subsystem

The Smart Device Framework[12] is a very useful application framework which enriches and extends the .Net Compact Framework. There are many class libraries and controls along with all the existing class libraries and controls available from www.opennet.org . This framework is open-source, this help us to learn how a function works and also enable us to improve it. I have made extensive use of this framework. One of the functions I have used enables me to schedule a process to be run at a specific time. Using this function I am able to turn off the network interface to save power, but later turn it back on to see if there is network connectivity. The CeRunAppTime API can help us to run an application at a specified l time, it is implemented as part of the OpenNETCF WinAPI library and handles conversions between the SystemTime struct and the .NET DateTime structure.

We also should know what information can also use this framework to get information about the AP from the wireless network interface, to do this some classes defined in OpenNETCF are used. We can get SSID, the current infrastructure in use by the adapter, the hardware address for the network adapter, the current type of network in use(see section 4.12 for details), the privacy mask for the adapter, the strength of the RF signal being received by the adapter for the SSID, and the list of supported signaling rates for the adapter (where each value indicates a single rate). This information concerning the network interface and AP provides the basic information about the wireless network coverage which we experience at our current location from specific AP. Once we have this information we will choose the an AP, then we can connect to get internet. We save this WLAN coverage information along with our current location, so that we can use it the next time were are at this location. This could be used to speedup handoffs, since if we return to this location we don't have to search for access points, but could directly try to use one of the APs we saw before.

2.5 Global Positioning System (GPS)

In this project, GPS is very helpful technology – because it helps us to know where the device is when it makes a measurement, hence we can relate these measurements to an absolute location. As noted before, the device can remember all the networks detected in this location previously and hence use this to predict which APs might be available in the future.

The following basic knowledge about GPS is from [13]. GPS refers to satellite-based radio positioning systems that provide 24 hour three-dimensional position, velocity, and time information to suitably equipped users anywhere on or near the surface of the Earth (and in some cases off the earth's surface). The NAVSTAR system, operated by the U.S. Department of Defense, was the first global positioning system widely available to civilian users. Applications include hand-held telematics, fleet tracking, and vehicle management systems – the later involve wireless communication devices designed for automobiles providing drivers with personalized information, messaging, entertainment, and location-specific travel and security services. GPS technology is used in a wide range of applications, including maritime, environmental, navigational, tracking, and monitoring.

We used a GlobalSat Bluetooth equipped GPS receiver as a GPS signal receiver and this receiver uses Bluetooth to connect to the PDA. The PDA runs the Pocket PC operating system and an application to learn where the user is. HP has also released a Bluetooth equipped GPS receiver for their iPaq. Additional information about these devices are from [14]. The concept of transforming your PDA into a Global Positioning System (GPS) device isn't new; in fact, GPS add-on products have been available for several years. However, untill now, these GPS modules required the use of a CompactFlash slot or some sort of cabling to connect the GPS receiver to the handheld device, or they attached to the PDA via a daughter card (expansion pack). In most cases, the GPS units themselves were often bulky and unsightly, adding unwanted weight and girth to an otherwise sleekly designed PDA. Today due to highly integrated GPS receivers, this is no longer a concern.

Additionally, thanks to Bluetooth technology, you can easily turn your PDA into a full function handheld navigation system without using cables, expansion packs sleds, or in some cases, an expansion slot. Instead you can use the expansion slot to insert digital maps (which are memory intensive), thus you do not have to worry about consuming the capacity of your handheld's internal memory. Bluetooth technology really pays off when using these devices as vehicle navigation aids. Wireless connectivity allows you to place the GPS receiver in a spot that has a clear view of the sky, while the PDA can be loacated such that it provide optimal screen visibility and easy access to menu screens.

The application we have developed should run on any HP iPAQ with integrated Bluetooth® technology - running the standard factory released operating system and software.

2.6 Common online storage technology situation

Online storage technology has become more and more mature. Today it provides shared folders (via the internet). Such folder sharing is used by many firms and individuals. This technology allows users to create a private peer-to-pear network that helps them to synchronize multiple files across multiple devices; while enabling sharing of files with other users. Some companies have begun to support file sharing services, for example FOLDERSHARE [15] Users no longer need to send large files via email, burn them on

CDs/DVDs and physically mail them, or upload files to a website (-as a file sharing service is makes it easier to manage files and share files with other users than upload files to a website.). These companies allows users to share and synchronize important information almost instantly with anyone that the user invites, thus making it a nearly perfect file sharing for both personal or small business.

Online storage based on network storage is a generic term used to describe network based data storage. There are three major variants of network storage: Direct Attached Storage (DAS), Network Attached Storage (NAS), and Storage Area Network (SAN).

Direct Attached Storage(DAS) involves a storage device directly attached to a host system, such much as an internal hard drive can be attached to a server. DAS is, by far, the most common method of storing data on computer systems.

Network Attached Storage (NAS): uses special devices (storage server) connected directly to the network. These devices are assigned an IP address and can be accessed by clients via a server (that acts as a gateway to the data), or via an intermediary.

Storage Area Network(SAN): A SAN is a network of storage devices that are connected to each other and a server, or cluster of servers, which provide access to the SAN. In some configurations a SAN is also connected via the network. For high performance, SAN's use special switches as a mechanism to inter the storage devices. These switches, which look a lot like an Ethernet networking switch, act as the connectivity point for connect the multiple servers participating in a SAN. Such switched it possible for devices to communicate with each and they also provide many advantages.

The technologies and protocols used in network storage communications are SCSI, RAID, iSCSI [16] , and Fiber Channel. For many years SCSI has provided a high speed, reliable method for data storage.. Fibre Channel is a technology used to interconnect storage devices allowing them to communicate at very high speeds (up to 10 Gbps in future implementations). As well as being faster than more traditional storage technologies like SCSI, Fibre Channel also allows devices to be interconnected over a much greater distance. iSCSI is a technology that allows data to be transported to and from storage devices over an IP network. What it actually does is serialize and packetize the data from a SCSI connection to enable it to be transported over an IP network. Using iSCSI, network storage can be distributed anywhere that IP packets go, which as the Internet proves, is basically anywhere.

JBOD (for "just a bunch of disks," or sometimes "just a bunch of drives") is a derogatory term - the official term is "spanning" - used to refer to a computer's hard disks that haven't been configured according to the RADI (for "redundant array of independent disks") system to increase fault tolerance and improve data access performance.

The RAID system stores the same data redundantly on multiple disks that nevertheless appear to the operating system as a single disk. Although, JBOD also makes the disks appear to be a single one, it accomplishes that by combining the drives into one larger logical one. JBOD doesn't deliver any advantages over using separate disks independently and doesn't provide any of the fault tolerance or performance benefits of RAID. [17]

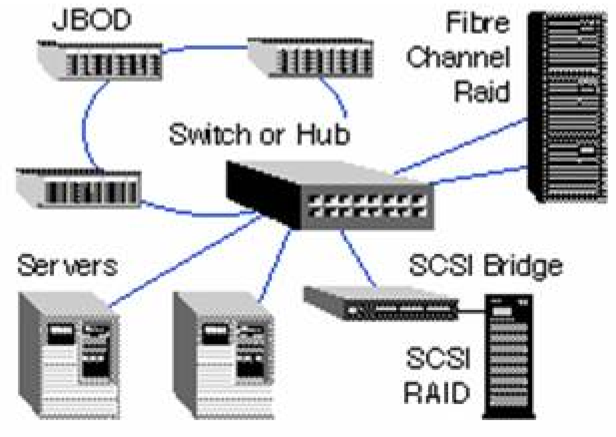

Figure 3 overview of online storage system

2.7 Microsoft's C# .NET Framework and .Net compact framework

C# is the newest of Microsoft's languages and makes use of the Microsoft .NET Framework. With its roots in the C and C++ programming languages, the chief architects of this new programming language, the chief architects of this new programming language. Anders Hejlsberg and Scott Wiltamuth, sought to deliver a product and would encompass the power of both languages while incorporating the simplicity and productivity of Microsoft Visual Basic. To create this language, Hejlsberg and Wiltamutch looked at all of the modern programming languages and adopted features that would make C# a language for the future.

The .NET Framework, upon which C# is built, is a comprehensive set of classes that provides functionality in every conceivable aspect of programming. This set of classes is known as the Common Language Runtime (CLR). Within the CLR there are graphics classes, data, XML, Directory Services, IO handling, and even classes that allow C# applications to examine their own metadata.[18]

The .NET Compact Framework is its equivalent for portable devices. The .NET Compact Framework, as it name states, is a compact version of the .NET Framework, it contains most of the features of the .NET Freamework, but some features are missing due to the differences between the architectures and operating systems of Windows for portable and non-portable devices.

2.8 Visual Studio .NET

Visual Studio .NET is not part of the .NET Framework. However, it deserves mention in an introduction of the .NET Framework. Visual Studio is an integrated development environment published by Microsoft for writing Windows programs. Visual Studio .NET can also be used to write managed applications in C#, C++, Visual Basic and any other language (such as Perl) that is integrated into the environment by a third-party.

Visual Studio .NET itself is a partially managed application and requires the .NET Framework to run. Visual Studio .NET is a very user-friendly and productive environment in which to write managed applications. It includes many helpful wizards for creating code, as well as useful features such as context coloring, integrated online help, auto completion and edit-time error notification.

Visual Studio .NET is a great product. But, it is important that you recognize Visual Studio .NET and the .NET Framework are different products.

The .NET Framework is the infrastructure for managed code. The .NET Framework includes Common Language Runtime (CLR) as well as other components that I discuss shortly. The .NET Framework also ships with an SDK (Software Developers Kit) that includes command line compilers for C#, C++, Visual Basic, and Inside Language (IL).

The bottom line is that the Framework is all you *need* to develop C# applications. That being said Visual Studio .NET can increase your enjoyment and productivity significantly.

2.9 Wireless Local Area Network (WLAN)

Wireless Local Area Networks (WLANs) are designed to cover limited areas such as, buildings and office areas. Today they are becoming more and more widely used not only in office and industrial settings, but also on the university campus and at users' homes.

Just as in an Ethernet LAN, every WLAN device has its own Media Access Control (MAC) address in order to be able to distinguish the link layer end points of the transmissions. IP addresses can be statically or dynamically mapped to these MAC addresses.

IEEE 802.11 is the family of specifications developed by IEEE for WLAN technology. Some of the members of this family includes 802.11, 802.11a, 802.11b, 802.11g,

802.11 wireless LAN up to 2 Mbps transmission in the 2.4 GHz band, ISM band; 802.11a is an extension to 802.11 providing up to 54 Mbps in the 5 GHz band; 802.11b is an extension to 802.11 providing up to 11 Mbps in the 2.4 GHz band; 802.11g provides 20+ Mbps in the 2.4 GHz band.

2.10 HP iPAQ 5550 Pocket PC

Some of the interesting features of this hand held device are the following:

- Increased memory capacity (128 MB RAM) enables the user to store many programs and files. With the iPAQ File Store, up to 17 MB Flash ROM, enables the user to store data in a safe place protected from battery discharge or device resets.
- An integrated IEEE 802.11b WLAN interface enables high speed wireless access to the internet or intranet.
- Integrated Bluetooth-wireless technology

Further detailed specifications of this PDA are shown in the following table:

Table 1: HP iPAQ h5550 specifications[1]

Operating system preinstalled	Microsoft Windows Pocket PC 2003 Premium
Enhanced security	Biometric Fingerprint Reader
Connectivity	Integrated Bluetooth wireless technology, WLAN 802.11b
Expansion slot	SD slot: SD, SDIO, and MMC support
Processor	Intel 400 MHz processor with Xscale™ technology
Memory,std.	128 MB SDRAM, 48 MB Flash ROM
Display	Transflective TFT LCD, over 65K colors 16-bit, 240 x 320 resolution, 3.8" diagonal viewable image size
Input type	Pen and touch interface
Audio	Microphone, speaker, and a four pole 3.5 mm headphone jack providing output and mono input to/from a headset
External I/O ports	USB slave and serial I/O
Dimensions (Lx Wx H)	13.8 x 8.4 x 1.6 cm.
Weight	206.5 g

[1] The information in this specification is from HP iPAQ 5550 User Manual (http://h10032.www1.hp.com/ctg/Manual/lpia8006.pdf, last access: 2006, 4, 9).

3. Method

We hope to get the available wireless networks connectivity probability by time increasing, position where user can heard the available networks, and energy of battery costs when transferring files. So to find the cumulative distribution function (CDF) of connectivity to available wireless networks and find the energy of battery cost curve will be our goals. To find the CDF by time, we need the time when we can heard the available networks and position, then use these data to calculate the time interval of connectivity, at last calculate the cumulation probability of these time intervals. The position where we can hear the available networks should be recorded for the future data analysis. Energy cost of battery should record the battery changing by time, choose some point to draw the curve of energy cost when transferring files, reading or writing files and so on.

To design the experiments, I divided the application to two parts. One is to survey the wireless signals and the current location, the other is to monitor the energy cost on battery. From these two parts, I can design application to get the data we need, then analyze these data to find the functions we need for the future work.

3.1 WLAN survey application

To find the function, we should collect the data concerning what WLANs are available. So a WLAN survey application for the HP iPAQ 5500 is needed. This application should collect the following information for us to use:

1. AP name: the name of the AP;
2. Mac address: the Mac address of the Adapter the device connect;
3. Signal strength: the strength of the signal from the adapter the device connect;
4. Nearby APs: the other APs the device can hear;
5. Signal strength of other APs: the signal strength of the APs the device can hear;
6. IP address: the IP address the device assigned;
7. Subnet mask: the subnet mask the device assigned;
8. Gate-Way: the gate-way of the connection;

This application collects data which wireless signals from nearby access points (AP's). For each AP additional time stamped information is collect, for example the AP's wireless interface MAC address and its signal strength. The program also notes which APs this mobile device (here an HP iPAQ 5550 Pocket PC) can connect to.

We will use GPS to detect our location, give the position of the device. Later we will use this same data to derive a cumulative distribution function describing the probability of having WLAN connectivity.

We collect the following information using GPS:
Latitude, longtitude, velocity (speed of moving and direction), and satellite signal status. As a side effect we also learn where we have good GPS coverage.

The user can also get GPS information, including a list of satellites - along with the user's speed, position, and quality of this position data. The user interface also shows the status of the Bluetooth connection to the GPS receiver (here a GlobaSat BT 338 Bluetooth equipped GPS receiver).

3.2 Battery monitor

For the energy cost of battery, the main energy cost is from four parts: the energy cost on files transfer (as a function of size); the energy cost on writing files to the SD card or reading files from SD card; the energy cost on media files online playing; the energy required to connect to APs. Design or use the exist applications to get all the useful data is what we should do. The applications should measure all the data we mentioned.

Collect the data of energy cost to draw the curves will help us to compare the energy cost in different situation. Energy cost will be display both by the data in the tables and curves in the planar X,Y axis. The functions should also be found for the different curves, these functions will be the foundation for implementing these results in the future.

4. Analysis

For our study, we want to utilize the survey system to collect data, then find fit functions to this data. The goal is to understand if the survey data enables us to predict future network connectivity.

The first survey system included the wireless signal survey and GPS survey, this system will focus on collecting data to be used to calculate the probability of finding an available wireless network in Kista. When we start this system, it will show the survey results both via a user on the interface and write them in a textual file.

The second survey system is designed to understand battery costs when transferring files. The cost is recorded when the transfer finishes.

We collect the data, then use numerical analysis methods to find functions to describe the changing of this data.

4.1 Wireless survey application design

The wireless survey application includes the display part, control part, and data recording part. The display part is used for displaying the survey information, both the wireless signal information and GPS information. The control part is used for controlling the start and finish of the application and displaying the survey via the display. Data recording is used for save the results of the survey. The results are saved in a simple text file.

4.1.1 Detail design

Initially we will collect data once per 1 second. This rate was chosen based on the maximum update rate of the GPS receiver which we used. The storage capacity needed to store all of the collected data for a full day would exceed the available storage of the device so a SD flash memory card was used to provide both additional capacity and to avoid the loss of data that might occur if the user inadverdently let the PDA's battery run too low. Using such a flash memory card, also means that the device does not need to use its WLAN link to communicate the collected values in real-time, but can upload the collected data later.

4.1.2 The wireless survey

Our survey program is based upon the openNETCF library.

Within the OpenNETCF.NET, there are some classes in this namespace we used. The '`Adapter`' class represents a single instance of a network adapter, which could be a PCMCIA card, USB network card, built-in Ethernet chipset, etc. We use the class '`AdapterCollection`' talk to the network adapters for the windows CE device. Then use '`Adapter`' class to create an instance of such a device. Then we check if this instance is a wireless interface or not. We use the function `AssociatedAccessPoint` to find the access point that this device has connected to. We can now use a function to get the signal strength and MAC address you of this access point adapter. Following this, we use the class '`AccessPointCollection`' to find the neighbouring networks and their functions in this class to get the signal strength and MAC address. These networks include not only the connected AP, but also all of the networks that we can hear in this location.

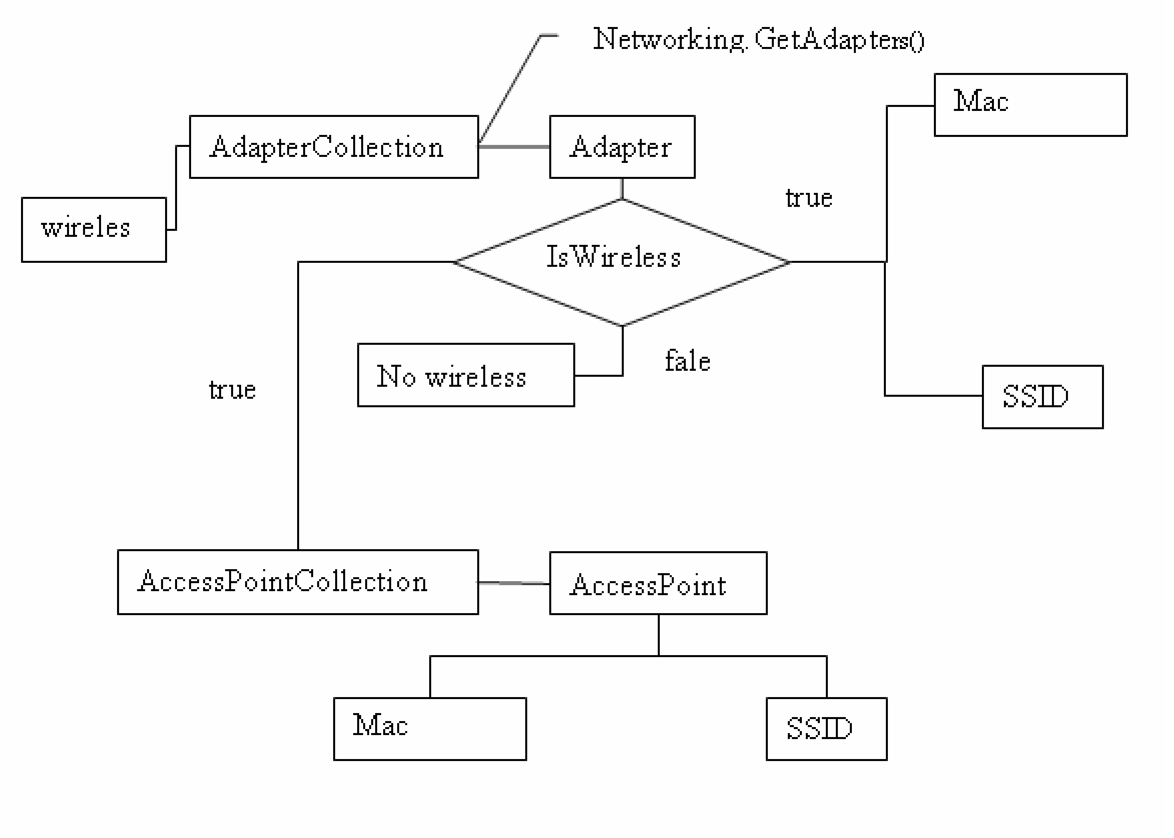

Figure 4 : Wireless survey design application

4.1.3 GPS survey

GPS can help us get the position learn where the device is. Given this information we can analyze the network situation at some position and know the networks changing in given area.

To utilize the GPS device, we use a dynamically loaded library (dll) file from Franson Technology AB. We use the interface they supply to create a virtual serial connection to the Bluetooth GPS receiver. It is based on their GPS ToolKit.Net. The serial port and baud rate can be set by the user or also can be set automatically. Users can see the GPS data in the list box if they wish clicking the button they want. The data includes the satellite information, speed information, quality, and the current position. The satellite information describes the satellites the receiver can find and get the signal. Speed will tell the user the moving speed of this device from some data. The quality is about the connection quality. The position is the exact position of the device, the longitude, the latitude and the altitude.

There are five parts in this application for describing the GPS information.

- GpsFix is for getting the basic GPS information including latitude, longitude, altitude, etc.

- In the quality, there are HDOP, VDOP, and PDOP. These three properties are for the precision of the GPS measurement.

- Movement is for measuring the GPS receiver moving information.

- Satellite is for us to get the Satellite information we can hear.

- ComStatus for us to get the physical connection for serial interface information.

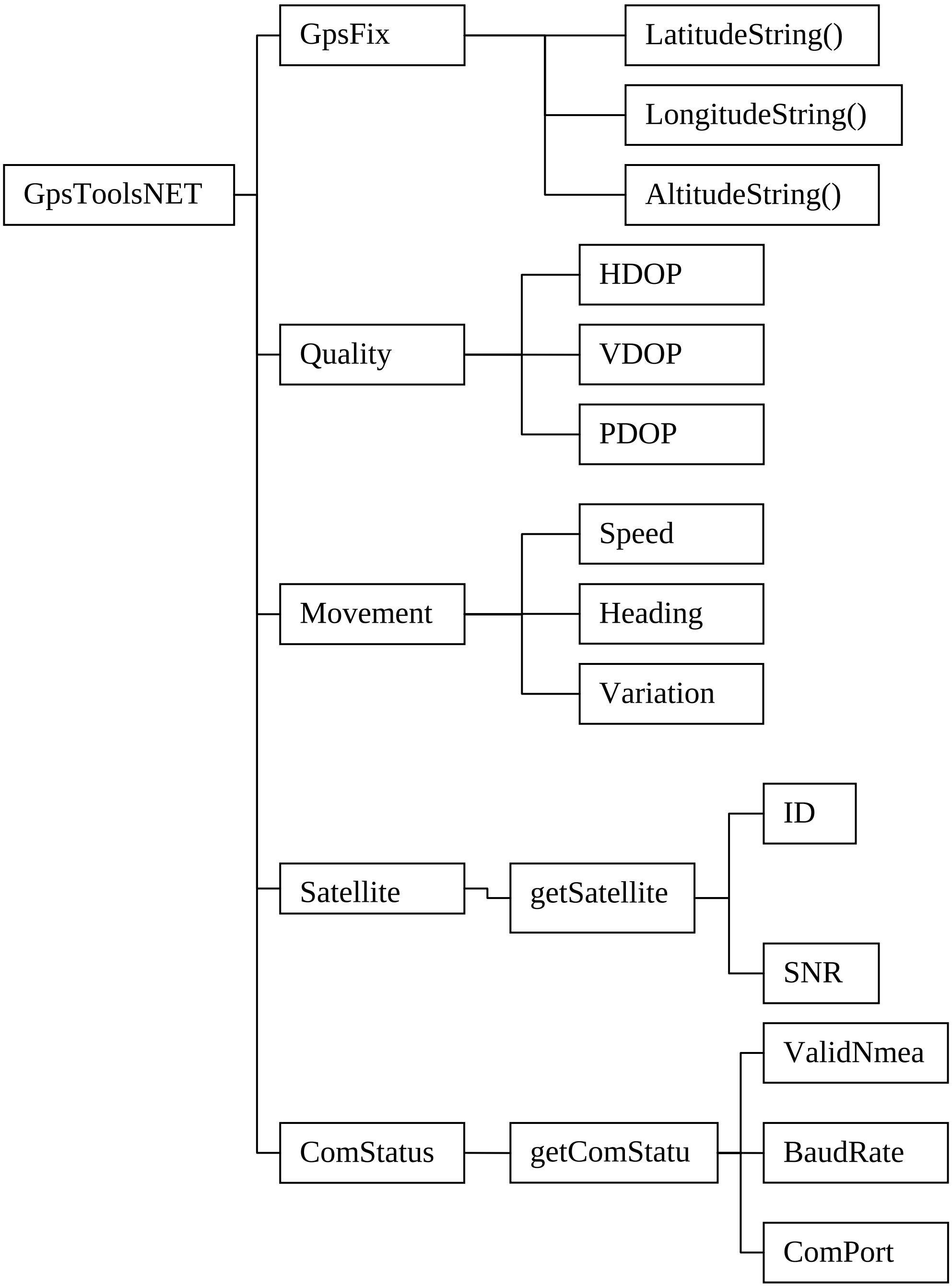

Figure 5: GPS works overview

From Figure 5, we can see clearly the structure of the GPS application. In five parts of this application, we will use different methods to get the different instances and then get the attributes of the instances. These attributes will be displayed by the application.

Useful data print: print the data we need for future data analyze. These data includes wireless network situation, the GPS situation and the time. In the wireless network there are the SSID of adapters, the signal strength for every adapter and their Mac addresses. In the GPS situation there are latitude, longitude and altitude. Record this information every one second. This we implement by timer function, set the timer function interval one second.

Application interface:

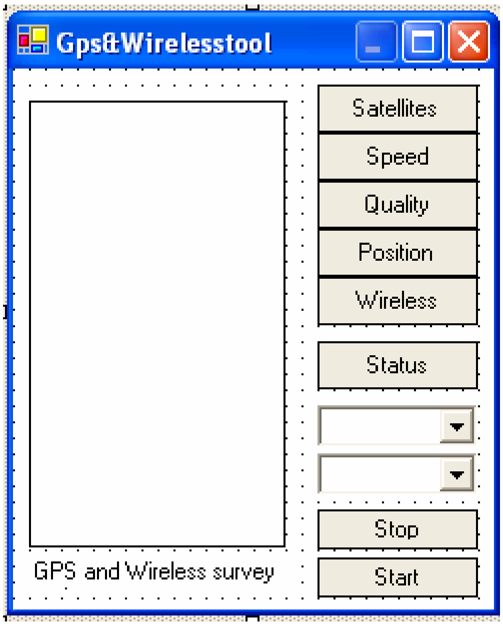

Figure 6: GPS & Wireless tool interface

From Figure 6, we can see the interface of this application clearly. There are some buttons in the right part of this interface and a text field in the left part of the interface. The “start” button and “stop” button is for controlling run or stop this application to measure. The “status” button is for getting the status of the application, including COM information and the other configuration status. The other buttons are for getting the measurement information, for example, if user press wireless button, the information of the wireless signal strength, AP’s name, MAC address and other information will be displayed in the text field. Press the other buttons, there are some relevant information in the text field.

4.2 Experiment 1

4.2.1 Method

Consider a user walking from their apartment in Arvinge to the Forum Building - part of the KTH’s campus in Kista, a suburb of Stockhom. Some possible paths are shown in red in figure7. We repeat to walk many times and get several groups of data.

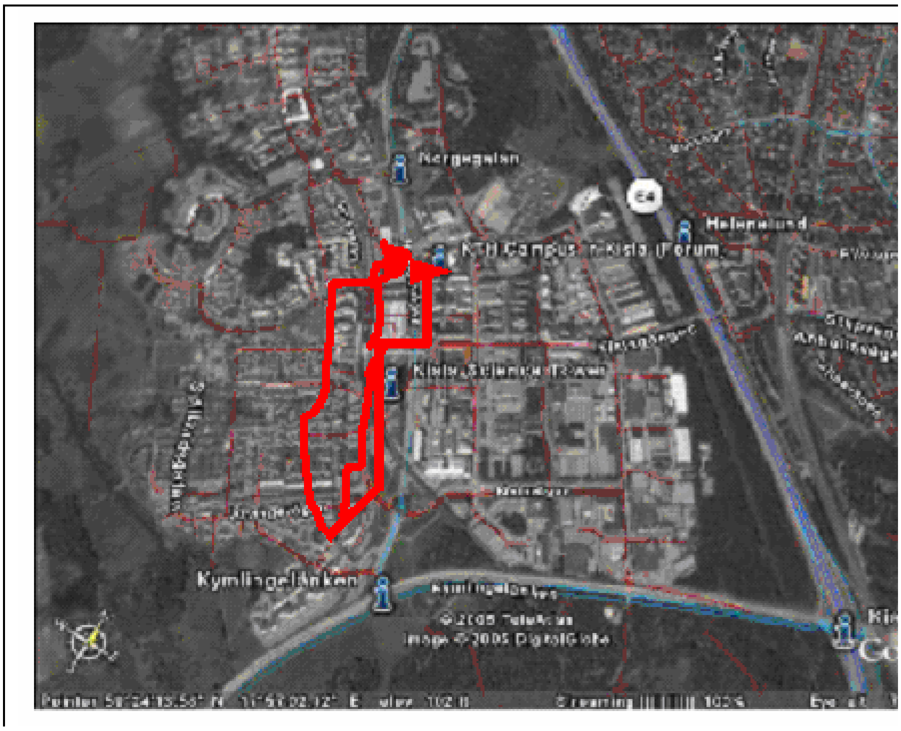

Figure 7: Experiment 1 several possible paths[2]
Path from user's apartment to campus on foot
(Figure from Google Earth on 2005.12.05)

Along these paths, we can hear wireless networks from several different networks, but most of these APs are not available for the user. The user records all the AP's names and MAC addresses as follow:

[2] This map is from Google Local, http://www.google.com/lochp?hl=en&tab=wl&q=, © 2005 Google. We have the permission to use it in this thesis from Google.

Table 2: Available AP MAC address

Network Name	AP MAC Address
Default	00-2-8A-A2-FE-C7
ACMilan	00-2-8A-A2-FE-C7
open	00-2-8A-A2-FE-C7
open	00-2-2D-2-88-A4
open	00-2-2D-2-89-50
open	00-2-2D-00-83-29
open	00-2-2D-00-87-34
open	00-2-2D-00-85-B3
open	00-2-2D-00-85-57
PH_LANDSCAPE	00-D-54-9E-14-DB
marakanda2	00-D-88-F2-8A-39
marakanda	00-D-88-F3-7B-88
hej	00-40-5-55-28-EB
Raukserver	00-11-95-F0-83-55
Home	00-D-88-3A-B5-1B
Paulsson	00-11-95-20-23-AD
dt_home	00-2-8A-A2-FE-C7-00-00
Stay_Away_,mac	00-D-88-C3-AC-37
SSET	00-9-5B-A9-D5-18
NETGEAR	00-F-B5-EB-22-C0
Bobby	00-9-5B-73-1-68
Bluestar	00-F-B5-51-F7-A2
Direct	00-2-8A-A2-FE-C7-00-00
NETGEAR	00-2-8A-A2-FE-C7-00-00
Nordrhein-Westfalen	00-11-24-61-4D-53
linksys	00-14-BF-48-76-98
default,mac	00-11-95-3D-86-91
Linnea	00-13-10-83-8F-59
PC_city	00-15-24-EF-7C-27
Home	00-9-5B-EF-4B-37
Linda’s home	00-15-24-EF-3B-30
Home1	00-9-5B-36-5D-47

In these AP’s, this user only can use the networks: “open” and “ACMillan”. Users want to know when they can access the available networks, we express this interms of the probability to connect to an available networks. Along each of several paths, we keep track of the network(s) we could used to connect to the Internet, but we do not attempt to connect continuously along the path. The connectivity available at each time step will help us to calculate the probability of access via these wireless network(s). The survey application I designed tries to listen for WLAN signals Access Points every 2 seconds. The user turns on this application when he leaving from his apartment, then turns it off when he arrives at his destination. The application records the network connectivity automatically, the user need not take care perform any other actions. The record includes the time, signal strength, GPS information, and each AP’s MAC address. Along these three paths, what time we can hear the available networks will be recorded and later used for calculating the probability of connecting as a function of time. We give a group of time result we surveyed which the times when we can hear the available networks. These data can be seen in Appendix. From the figure below, we can see the one of the survey results. It describes changing of connectivity status on these three paths by time.

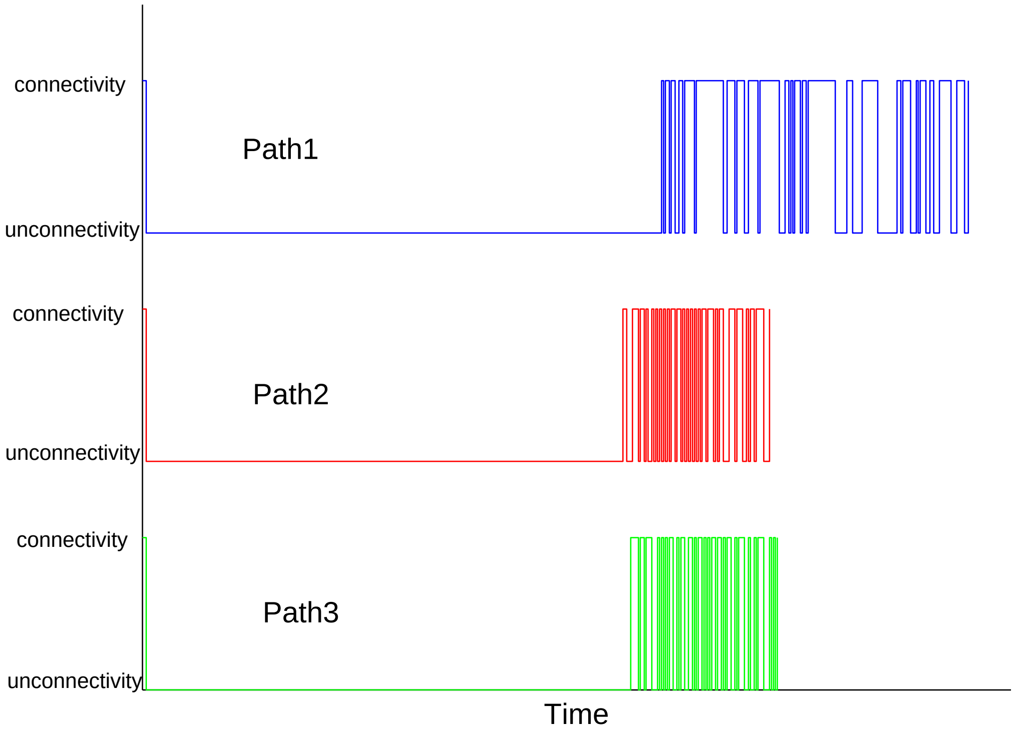

Figure 8: connectivity status by time on different paths

Figure 8 shows us a group of data we surveyed on these three paths. We can see at beginning we can connect to available network, then in a long time we can not connect to any available network. By the time passing, we can connect to available network frequently.

4.2.2 Analyze the result of the wireless survey experiment

To calculate the probability of connecting easily and directly, we consider the status for these as a function of time, we set the node an entry in the list to be “1” when we can have connectivity, set it to “0” when we can not have connectivity. Then translate all the time we can connect to available network to “1”, all the time we can not connect to available network to “0”. Here I give an example as below. It is one of five groups of data we measured. From this group of data, the status list which describes the available/potential network connectivity on each path at 2 second is shown below:

path1:

1100000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000010110110011
0111110111111111111110011110111100111110111111111100011010111011011111
1111111110000001110000011111111000000000011011110001011100110001111110
001111001

path2:

1100000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000011000111011010010101010110110
1010101010110111010110001110111001011011110001

path3:

1100000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000111101101110001010101100101
10011010110101011011010110010111001001011100010101

Via these status lists we can see directly that the connectivity status of the available networks for each time step and for path. Because we only survey the signal every 2 seconds, the smallest time interval is 2 seconds. The user hopes to know the probability in a certain time interval. Thus we consider we define the time window a of 2 seconds, 4 seconds, 6 seconds, and so on to the largest size which is the time we from start of the path to the destination. The figure below gives the time window partitions.

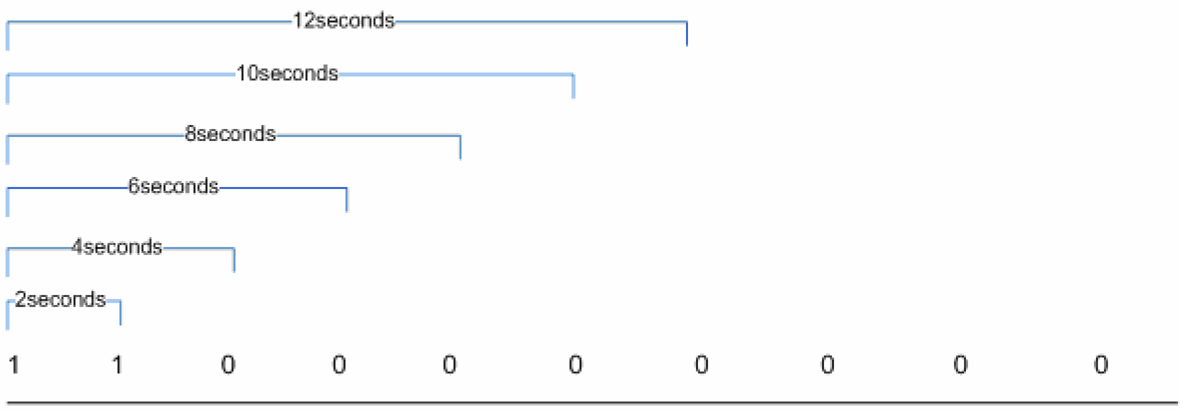

Figure 9: Time windows define

Figure 9 shows how we partition the total time into windows. For these three paths, the largest time window is 856 seconds on path 1, 664 seconds on path 2, and 672 seconds on path 3. As shown in the figure, all the time windows are multiples of two. After we defined the size of our time windows, we must decide how to using these time windows. We give a figure below to explain.

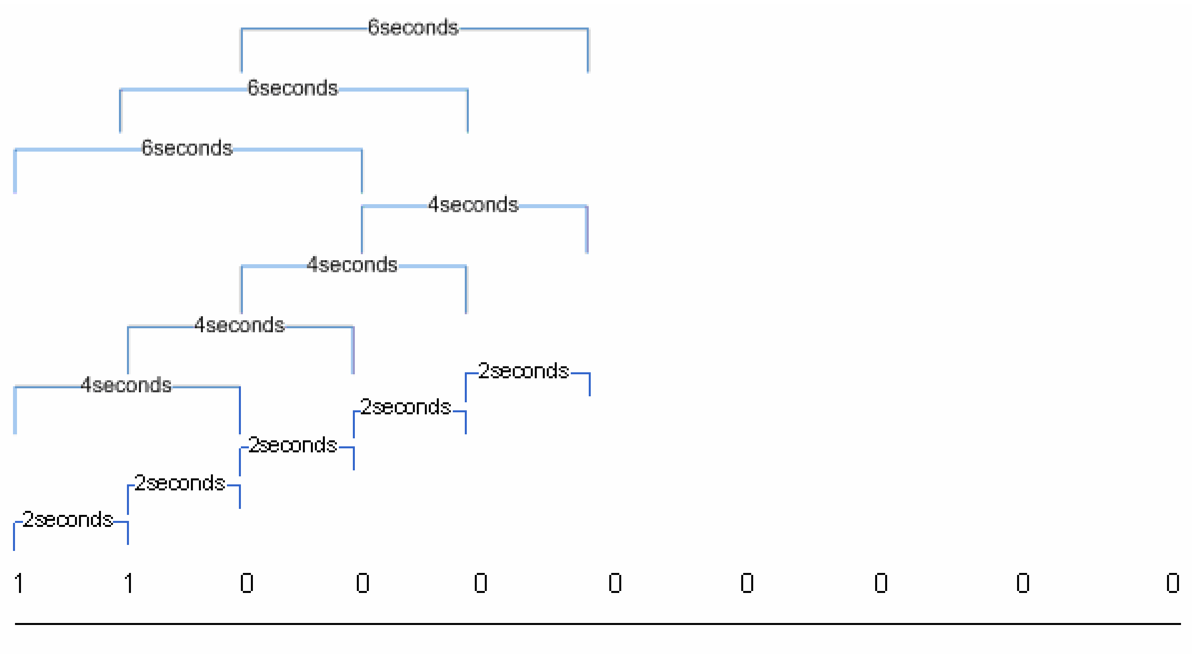

Figure 10: Sliding time windows

In figure 10, we can see a time window of a given size of it moves from the left to right unit by unit, when the time window cover the last unit, the time window will stop moving. We give an example of using a time window of 2 seconds. If there is the status "1" in the time window, we can say that we can connect to the available network for these 2 seconds. For every size of time window we need a list to describe the connectivity status when it moving at each offset. We also set "1" connected, "0" unconnected. We should redo this by the time window moving, when the time window moves one unit. For each time window we consider we record the connectivity status again, then slide the window to be considered to the right. The result is the list below which is the 2 seconds time window status list.

Sequence number		Status
1	:	1
2	:	1
3	:	0
4	:	0
5	:	0
.		
.		
.		
.		
.		
.		
856	:	1

In this list, we account how many "1" occur. The offset to the first "1" and the total number of the units the time window moved is the connectivity probability in the time window on this path. For this example list, if we there are x "1"s in this list, and the probability of access should be x/856. That means the probability to connect to an available network on this path in 2 seconds is x/856.

Continuing this, we can calculate the probability of connecting to the networks in 4 seconds, 6 seconds and so on along path1, path 2, and path 3. Based upon this analysis we derive the probability is also sequential, the probability we get will be fit for every where and every time you turn on the wireless adaptor to listen networks on the paths. We use the same method to calculate for different groups of data we surveyed. Then the probability of access curves on these three paths as follow:

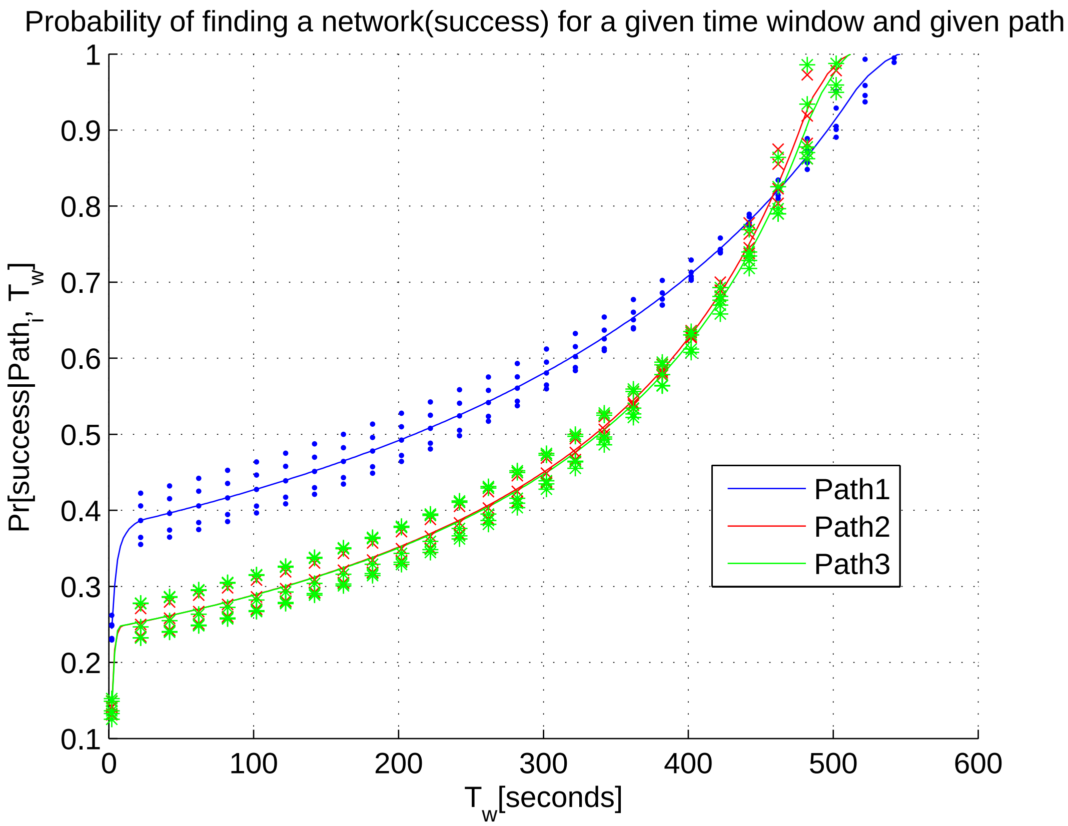

Figure 11: Probability of available networks connectivity on different paths

In this figure, Pr[Success/T_w] means the probability of success connecting to an available network. T_w[seconds] shows the size of the time window. The points which distributed around the lines are the probability on the given time on different paths. The three lines are the average probability curves for these three paths. The curves describe the probability of connecting to an available network within a given time. Three lines indicate the connectivity probability changing on these three different paths, the shape of these curves are nearly the same, at the beginning these curve go up rapidly then the shape changes gently but continue to increase, the probability increases continually with time (as expected - since there is connectivity at the end of the path).

.

We assume that users choose the path to go to follow equal probability. Thus the average connectivity probability curve from the user's apartment to the destination is as follows:

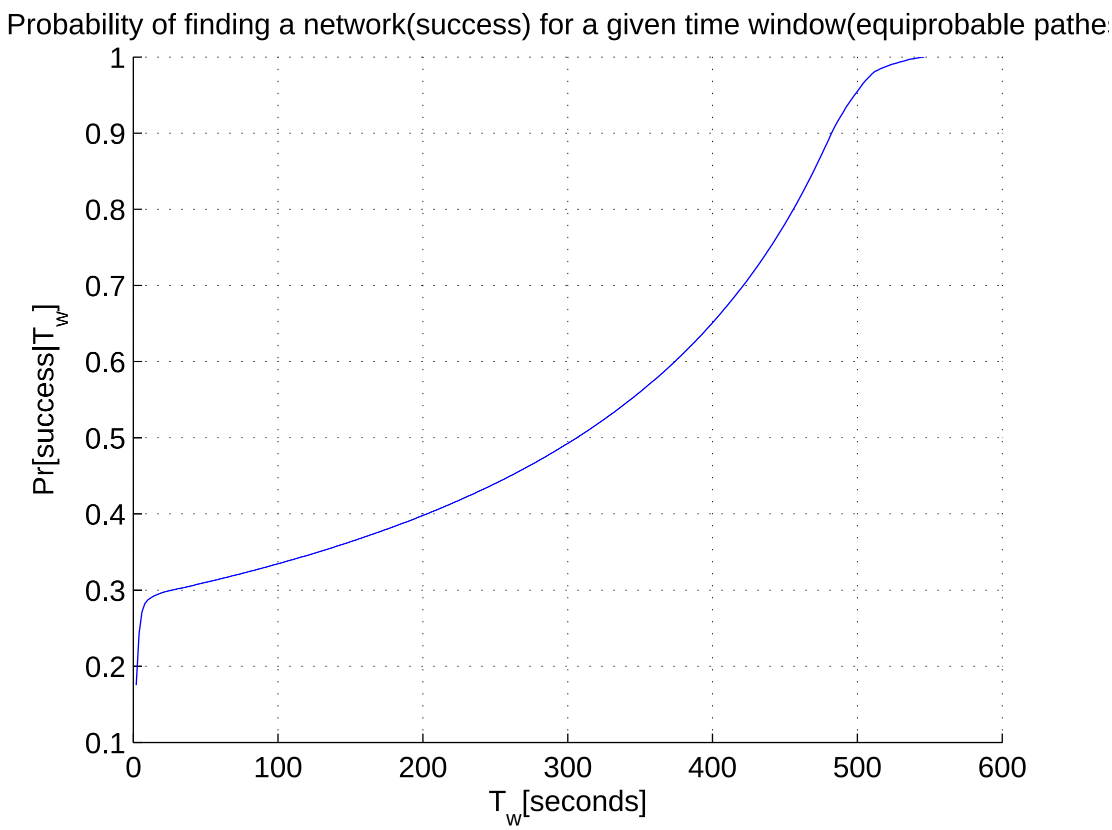

Figure 12: Average Probability of finding an available networks offering connectivity as a function of time

In figure 12, the curve describes the average probability of connecting to the available networks. This curve has a mostly similar shape to the curves of each individual path. It goes up rapitly at the beginning, then increase slowly. Users can only connect to an available network with some difficultly in any interval of 0 to 10 seconds, because we can see that from 0 to 10 seconds the probability is less than 30%. But for intervals of longer than 10 seconds, the probability increases very fast, it rises 10% in only 10 more seconds. From 10 seconds to 500 seconds, the probability rises does not increase as fast as in the earlier intervals of 0 to 10 seconds. After 500 seconds, the probability rises more slowly.

On these three paths, the time users have to wait for connecting to the available networks is also very important to users. The following data tell us how many instances of time to wait on the path.

Path 1
75 instances for 2 seconds
67 instances for 4 seconds
40 instances for 6 seconds
24 instances for 8 seconds
20 instances for 10 seconds
15 instances for 12 seconds
13 instances for 14 seconds
11 instances for 16 seconds
10 instances for 18 seconds

10 instances for 20 seconds
5 instance for 22 seconds
5 instance for 24 seconds
.
.
.
1 instance for 540 seconds
(The numbers of instance from instance for 22 seconds to the end are all 1 instance)

Path 2
130 instances for 2 seconds
34 instances for 4 seconds
20 instances for 6 seconds
7 instance for 8 seconds
.
.
.
1 instance for 500 seconds
(The numbers of instance from instance for 8 seconds to the end are all 1 instance)

Path 3
136 instances for 2 seconds
47 instances for 4 seconds
45 instances for 6 seconds
27 instance for 8 seconds
13 instance for 10 seconds
.
.
.
1 instance for 508 seconds
(The numbers of instance from instance for 8 seconds to the end are all 1 instance)

From these data, we can know the distribution of these instances. It also means we can see the distribution of the waiting time. So we can get the CDF for every path like the picture below:

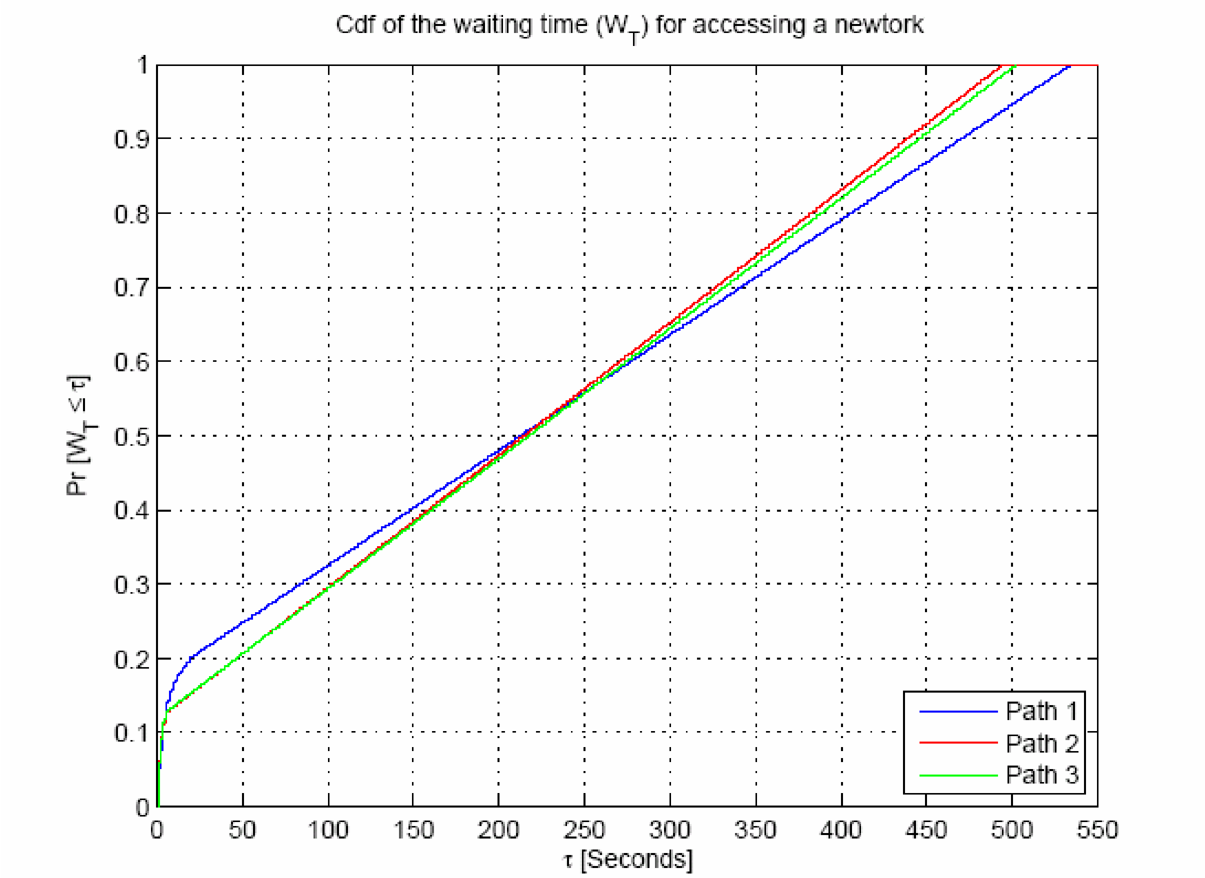

Figure13: Distribution for users have to wait for connecting to the available networks

From this figure, we can see the distribution of the time users have to wait directly. If users have the same probability for the path choosing, the average CDF should be below:

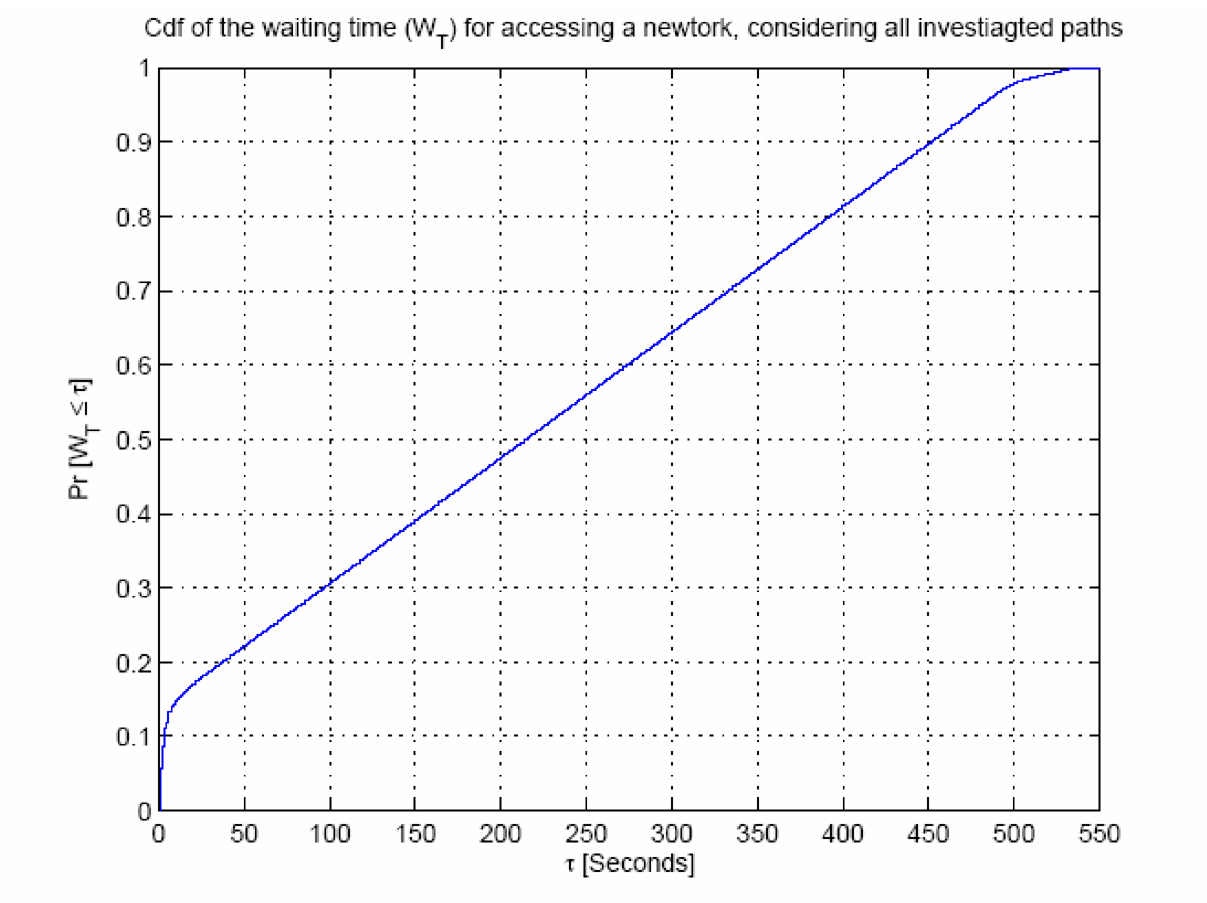

Figure 14: Average distribution for the time user have to wait

4.3 Experiment 2

Later in the day the user walks from the Forum building to the Wireless@KTH center where they enter and go to the Computer Communications lab, as shown in the figure below.

Figure 15: Subsequent Path from Forum building to CCSlab[3]

Since the user is equipped with their HP iPAQ 5550 PDA and would like to access some content, such as reviewing the audio from the latest lecture they just attended, their user can either download all the data before leaving WLAN coverage within the Forum building or take advantage of WLAN access points along the way.

Along this path the user hears several networks via many different AP's. From the data we collected, we can find that along this path, the user can utilize the OPEN network anytime. We know the position of the AP's along this path, so we can record the signal coverage of every AP by measurement.

We use the application we developed to survey the wireless signal along this path. Along this path to CCSLab the, we could access the OPEN network from 12 different AP's. The speed of the user's movement is about 1.5 m/s. The user walked from Forum building to the CCSLab in Wireless@KTH, the total time is 3'12''= 192 seconds, the distance is about 288 meters. We can deduce the coverage of the AP's based up their signal strength, when the user walk far away from the AP and the signal strength will be lower and lower till can not hear any signal. Then we also walk around the 2IT building, Electrum building, and Wireless@KTH building. With the help of the [19] we will get the relative coverage of every AP we have can connect to. We draw the AP distribution as below. We also can get the MAC addresses of the AP's we can connect and how many AP's we can hear at a given time.

[3] This map is from Google Local, http://www.google.com/lochp?hl=en&tab=wl&q=, © 2005 Google. We have the permission to use it in this thesis from Google.

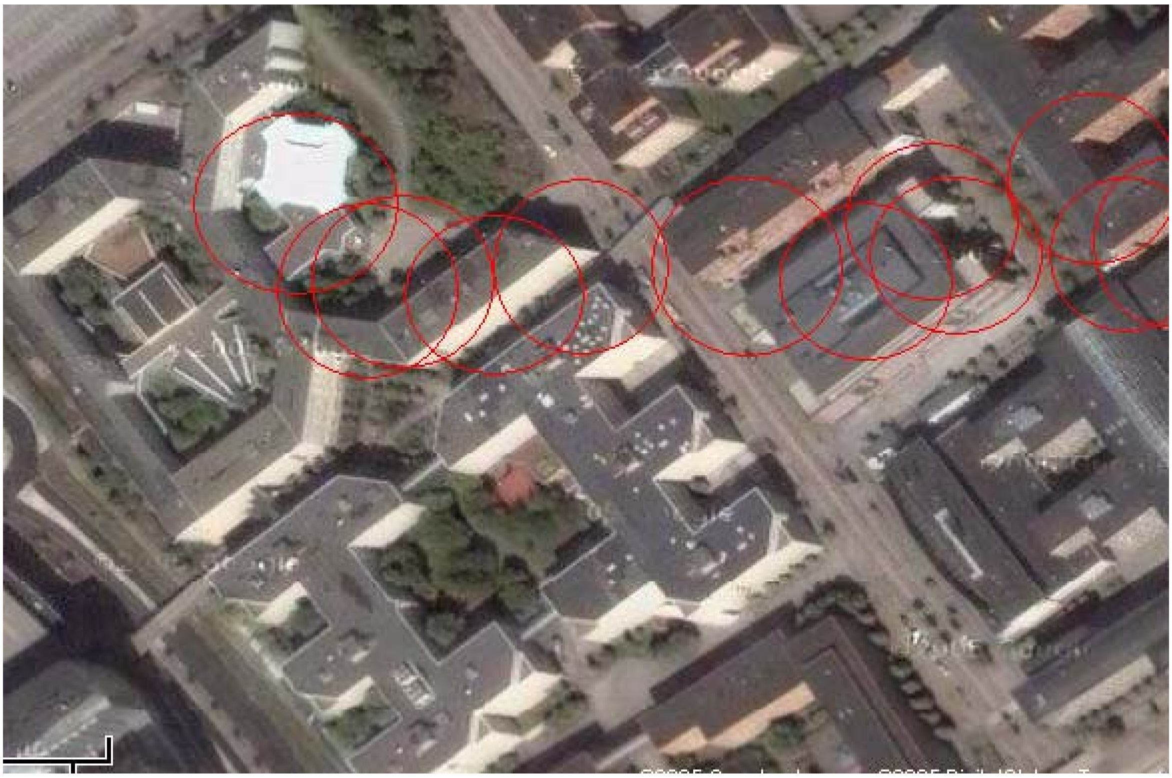

Figure 16: General coverage of AP's[4]

This figure shows the general coverage of every AP, the centre of the circle is the general position of the AP. But the circles are not base upon the measurements of the coverage area, but are just an approximation based on a uniform coverage from an AP located at the centre of the circle. In this figure, we can see that, the signal covers the whole path. Thus in practice, when we walk from the Forum building to the Wireless@KTH, we can have uninterrupted wireless accessibility everywhere on this path. That also means the user can transfer the file(s) as they walk along this path. On this path, we can connect to 12 different access points, we also can hear the signal from other different access points, the MAC addresses of these access point we can connect and we can hear are as the below.

[4] This map is from Google Local, http://www.google.com/lochp?hl=en&tab=wl&q=, © 2005 Google. We have the permission to use it in this thesis from Google.

Table 3: Networks and AP’s distribution

Time	Number of heard AP’s	MAC address of the connected AP
9:24:26-9:24:28	5	00-2-3A-4-4C-47
9:24:30-9:24:36	5	00-3-4C-32-B8-37
9:24:38-9:24:46	4	00-2-8D-56-A9-2B
9:24:48-9:25:02	4	00-3-2D-00-46-23
9:25:04-9:25:40	2	00-2-5B-8-B4-5F
9:25:42-9:26:12	3	00-2-7C-00-A8-34
9:26:14-9:26:56	4	00-2-2B-00-A6-C7
9:26:58-9:27:26	3	00-2-2D-3-7C-54
9:26:28-9:26:48	3	00-2-8A-A2-FE-C7
9:26:50-9:27:00	4	00-2-2D-2-88-A4
9:27:02-9:27:14	4	00-2-2D-2-89-50
9:27:16-9:27:28	5	00-2-2D-00-83-29

From table 3, we can see when we can hear the available networks and how many we can hear, also can see the AP’s address which we can connect to. Although we can hear the available networks all the time on the path, the signal strength changes along the path. On this path, the signal strength levels we can hear are “very low”, “low”, “good”, and “very good”. We define four numbers to indicate these four levels, “1” means very low, “2” means low, “3” means good, and “4” means very good. We also use the application we used in experiment 1, we also to compute the resulting of the network situation every 2 seconds. The records are as follow:

444443333222222111111111111111111111222222222222222222444444444444333333333
333333333333333334444444444444

These data describes the signal strength at every 2 seconds. So the probability of very good signal strength is the ratio between the times we can hear very good signal and total times we were on this path. After calculating, we determined the probability of very good signal strength is 30/105 = 28.6%, the probability of good signal strength is 30/105 = 28.6%, the probability of low signal strength is 24/105 = 22.9%, and probability of very low signal strength is 21/105 = 20%. From the figure below we can see the probability we calculated on this path for every signal strength level.

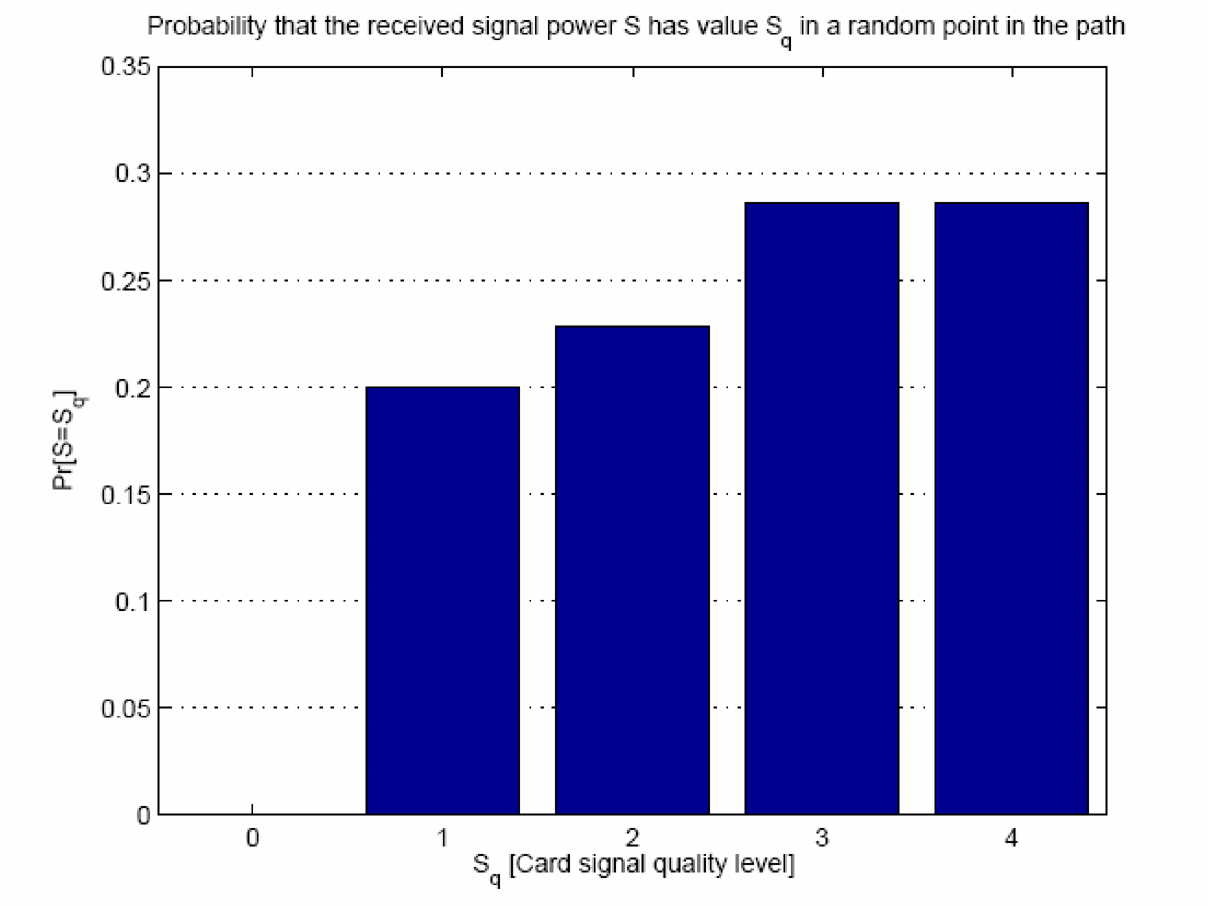

Figure 17: Probability that the received signal

The figure below gives us the level of signal strength changing on this path. In figure 14, we can see directly the signal level changing by time, we also can find that level 1 and level 2 appear between the 20th seconds and the 110th seconds. In this period, the user just in the middle of the path and far away from Forum building, 2IT building, and Electrum building. Geographically, the user is in an area with fewer available network AP's.

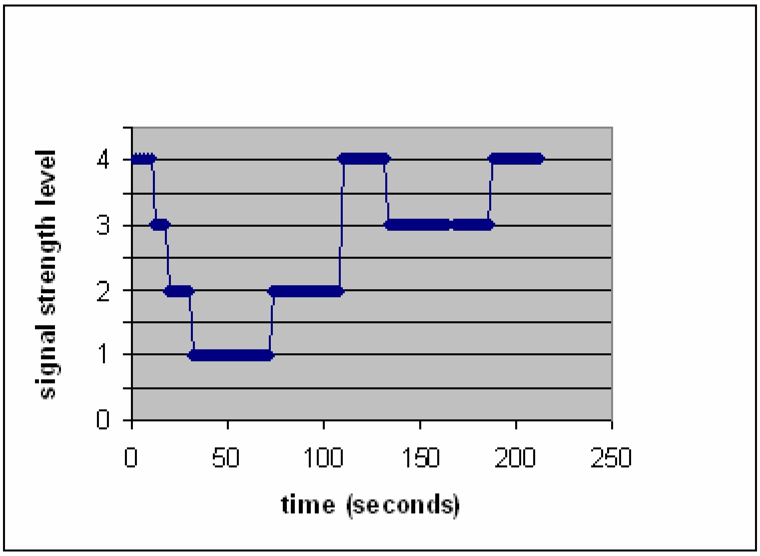

Figure 18: Signal strength level distribution on the path

After these two experiments, I learned that the available wireless AP's are located in Galleria, Forum building, Wireless@KTH building, Electrum building, and 2IT building. So these places and around these places are the area the user can connect to the available wireless network with a high probability. When users are in this area, they can turn on their device's wireless adaptor if there are some files that should be transferred or will be transferred. Therefore, determining the user's location is the first step in deciding if processing files transferred or not. After determining the user's position, then we will consider the other factors, before using the function to determine the probability of wireless connectivity base upon this we can turn the wireless adaptor on/off. Therefore knowing the coverage areas of available wireless networks with high probability is an important parameter of the whole solution.

The area we can connect to the available wireless network with the high probability is in the red circle as bellow:

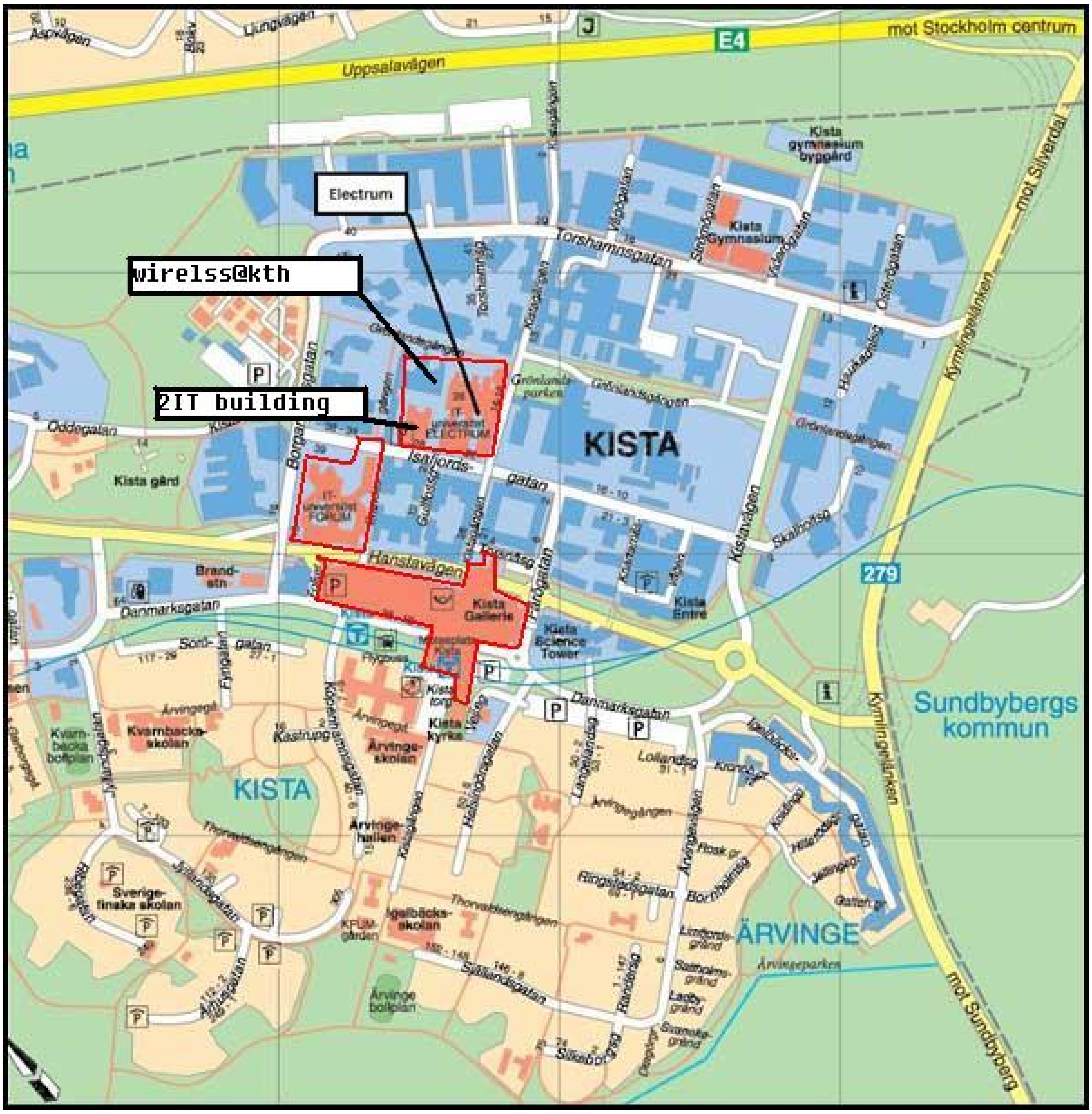

Figure 19: Available wireless main coverage in Kista[5]

5 This map is from http://www.hitta.se, © T-Kartor Sweden AB. We have the permission to use it in this thesis from T-Kartor Sweden AB.

5. Battery monitor experiment

Battery power is one of the limitations of the Pocket PC or even all the other mobile devices, users can not give their mobile devices a continuous power out doors, they must charge the battery when there is an available power source. The wireless LAN adapter often expends much more power than the other transfer medium, something like wireless link, such as Bluetooth and infrared.

WLAN offers advantages as a fast and steady technology when users transfer large files. Of course, while users hope to transfer all the files without packet lost, there are of course some reasons makes why a transfer may experience some problems which results in the file transfer taking longer than expected to finish. Sometimes, this problem is due to discontinous wireless network connectivity, fortunately this is not the worst, we may be able to wait and take advantage of the wireless network connectivity later, then we can continue to the transfer. Sometimes the problem is due to running out of battery, power to support the transfer when this is true we have to turn the wireless adapter off. The probability of finding a place to charge the battery is much lower than hearing a network in many places. If the user wants to download a file, there is not enough power to support downloading it, then this user need to charge the battery and transfer the file later. That means it is not possible to download this file at this moment, as a side effect, we save the battery power by delaying the transfer.

We would like to find a function which describes a relationship between battery utilization and the file transfer size is a method, so that we can use this information to avoid the situation we mentioned running out of power during a transfer. We designed and implemented the software to check and record the battery status change when users transfer a given file. The figure below describe the percent of the battery content curve remaining as over time, we can see how it changes we transfer different sizes of files. There will two situations we should consider: one is when the wireless adapter is off before starting a transfer, the other one is the wireless is already on.

For these experiments we used the following storage device:

Kingston 1.0 GB SD card

There are three transfer modes supported by SD card
-SPI mode, separate serial in and serial out
-One-Bit SD mode, separate command and data channels and a proprietary transfer format
-Four-bit SD mode, uses extra pins plus some reassigned pins to support four bit wide parallel transfers.

IEEE 802.11 wireless LAN interface in the HP iPAQ 5500 has several modes:
-Receive mode
-Transmit mode
-Sleep mode
-Sleep to receive/transmit mode

In this project, we use the high speed Kingston 1.0 GB SD cards which support up to 100 Mbit/s data rate in four-bit mode and 0-25 Mbit/s in SPI and one-bit SD modes. Similarily we used the following modes of the WLAN interface: x and y.

5.1 Monitor Application design

In this experiment we developed an application to monitor the remaining battery capacity. There are two situations we surveyed:

1, the battery utilization WLAN interface simply is on.
2, the battery utilization a transfer takes place continuously underway.

This application includes two parts. One is for transferring files and the other is for monitoring the battery.

5.1.1 Battery monitor module design

To get the battery status, we use two classes: SYSTEM_POWER_STATUS_EX and SYSTEM_POWER_STATUS_EX2.

We use the methods GetSystemPowerStatusEx( ) and GetSystemPowerStatusEx2( ) to get the instance of this class SYSTEM_POWER_STATUS_EX and SYSTEM_POWER_STATUS_EX2. Then can access the members we will need. These members include: BatteryFlag, BatteryLifePercent, etc. The structure of this application is as follow:

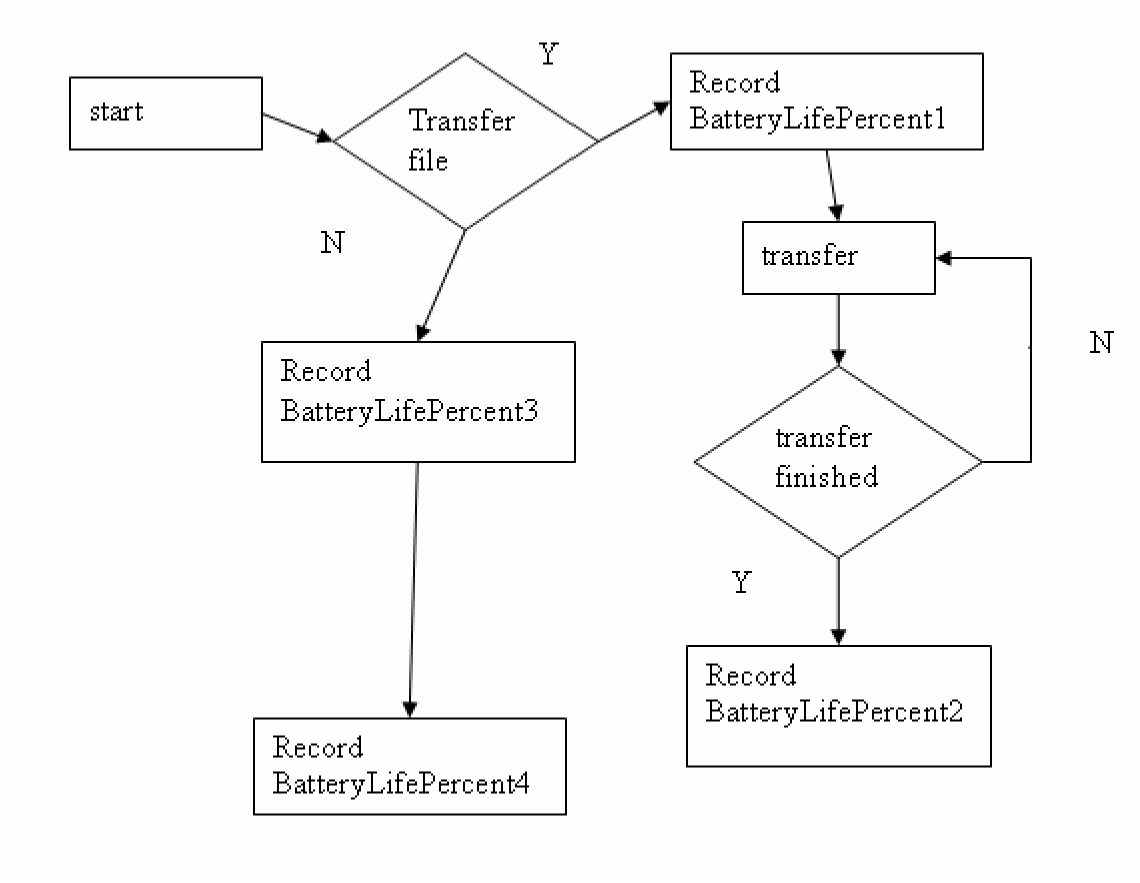

Figure 20: Battery monitor module design overview

From this figure, we can see that, the status: record batterylifepercent 1 and 2 are for calculating the battery utilization when we transfer the files, the record batterylifepercent 3 and 4 are for calculating the battery utilization when only the interface is simply turned on. Make two values be reduced, the result is the cost. The difference between these two rates of utilization gives us the power utilized as a function of the amount of data being transferred.

5.1.2 File transfer module

Different types of files we will use different transfer methods. For the audio or video files which will require real time play, a real time streaming protocol [20] should be used. For the files transfer directly to the SD card or upload, the File Transfer Protocol (FTP) [21] will be used. The users who have online storage space use FTP to management their files. The objectives of using such on-line storage are to promote users sharing files, to encourage indirect or implicitly, to shield a user from variation in file storage system among host and to transfer data reliably and efficiently.

Consider of the stability of the network, multiple-threads are used to resume broken download/upload. The structure of the file transferring is as follow:

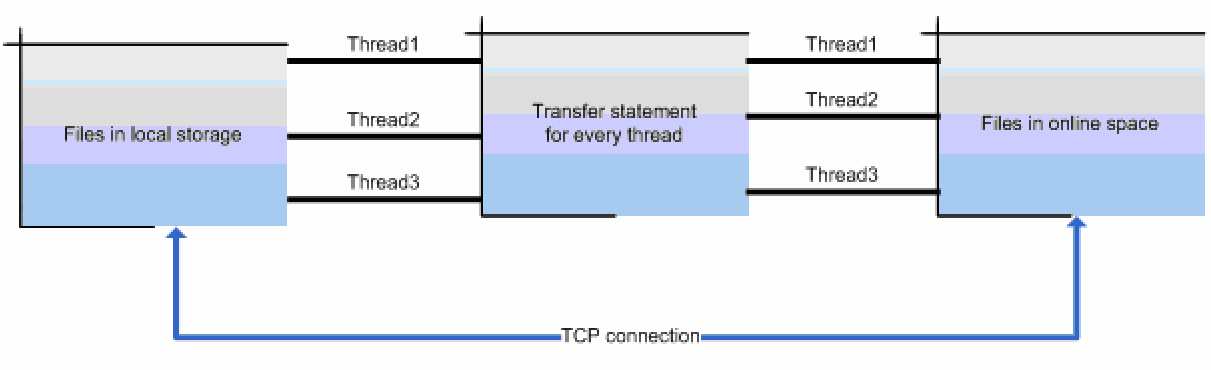

Figure 21: Transfer module overview

A configuration file saves the transfer state of every thread. This file stores the thread number and how many bits have been successful transferred. When the network activity is broken, this configuration files will immediately save the transfer status information. When the device has the network again, it first reads the configuration file to find where it should to continue to each transfer.

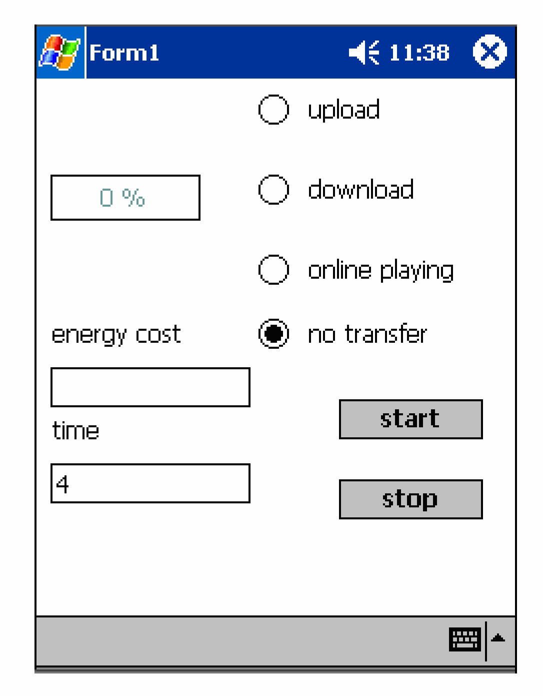

Figure 22: Battery monitor application interface

5.2 The result of battery monitor

5.2.1 Energy cost when transferring files

When we start to transfer files, we start the application and choose upload or download files. Actually the energy cost includes two parts, one is from the cost of the WLAN interface; the other is from writing or reading the file to or from the SD card. The results are as the table below.

Table 4: Sending files (read from SD card)

File size (Mbits)	2	4	6	8	10	14	20	60	100
Fraction of the total battery power that is used	1%	2%	3%	4%	5%	7%	11%	33%	55%

This table describes the energy cost on the process for reading files from SD card and sending these files via the WLAN interface. We also can find the relationship between file size and the fraction of the total battery power that is used in this table.

Table 5: Receiving files (write to SD card)

File size (Mbits)	2	4	6	8	10	14	20	60	100
Fraction of the total battery power that is used	1%	2%	3%	4%	5%	7%	11%	32%	53%

This table describes the energy cost of reading files from the SD card and sending them via WLAN interface. We can see some experiment data of file size, when we read files from SD card and then send out them, the relevant energy cost can be also found in this table.

From the data in Table 4 and Table 5, we can draw the curves as bellow.

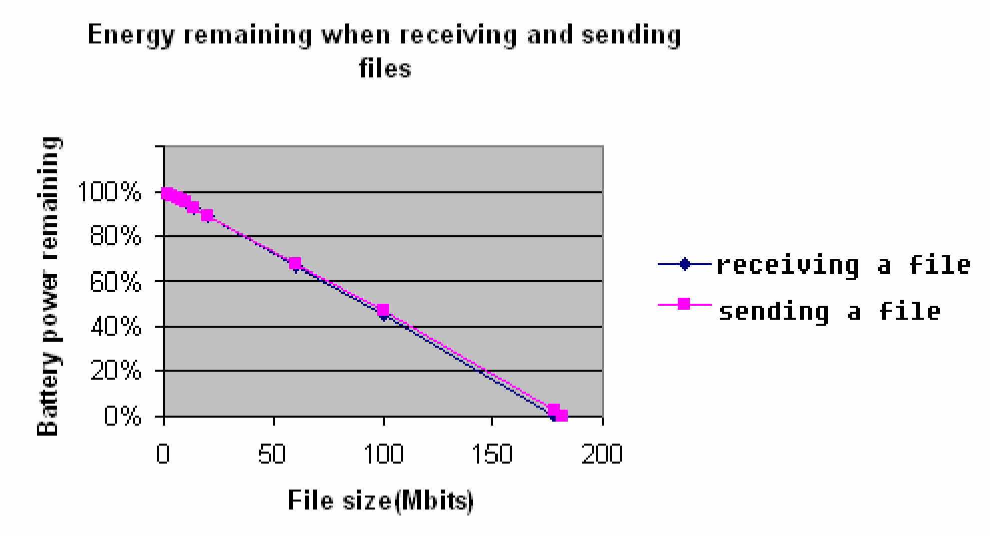

Figure 23: Energy remaining when receiving and sending files

The battery power remaining function of receiving a file is y1= 100 - 0.55x
The battery power remaining function of sending a file is y2= 100 - 0.56x

Figure 23 is from the table 3 and table 4, we can see clearly, at the beginning of the curve the energy cost of receiving and sending process are nearly the same. But as file size increases, the energy cost becomes increasingly different. The receiving process requires lower power than the sending process. That means the receiving process can transfer a larger file than sending process for a given amount of energy.

When the WLAN interface is simply turned on, we turn on the monitor application and start to survey. We repeated them 9 times for different durations (before starting each survey the battery was recharged). The results are as bellow:

Table 6: Energy cost of only WLAN interface simply on

Time(seconds)	90	180	360	540	1080	1440	1800	3600	5400
Fraction of the total battery power that is used	1%	2%	4%	6%	12%	16%	20%	40%	60%

This table shows us the relationship between energy usage and time. We can see the energy cost of simply leaving the WLAN turned on.

From these data, we can get the energy cost changing curve when only the interface simply on as bellow:

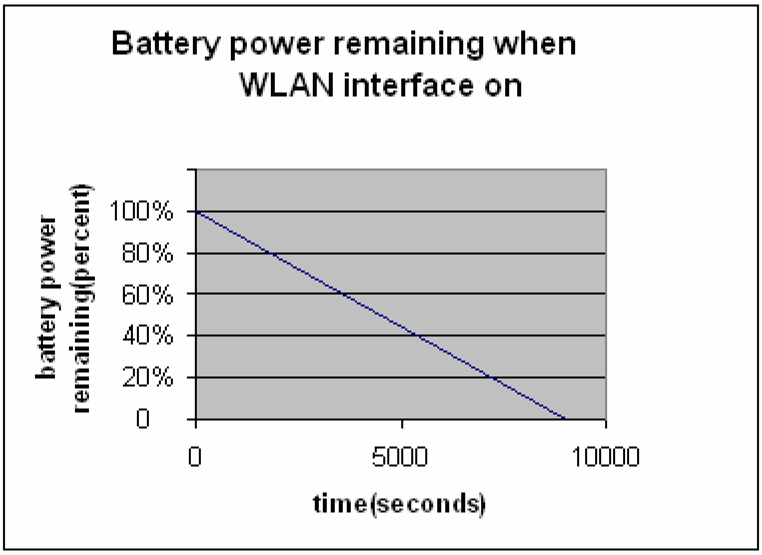

Figure 24: Battery power remaining when WLAN interface is on, but not actively sending or receiving user data

Figure 18 shows us the energy usage when costing curve the WLAN of only simply interface is turned on, but not actively transmitting a user data. In this figure we can see the energy use is linear with time.

The function of this curve is: $y = 100 - t/90$

When users walk along the path we mentioned in 4.2.1, they can not get a continually available network when transferring and there is a long time they have to wait to connect to an available network again. If we keep the WLAN interface on all the while, there will cost much unnecessary battery power. To save the battery power, we should turn the WLAN interface off. When users go through an area with high probability of connectivity, we can turn on the WLAN interface again.

To compare the energy on turning on/off wireless interface and energy cost on wireless interface keeping on, we did the other experimentation to find the energy cost on turning on/off wireless interface. We turn on and off WLAN interface continually till the energy costs

1%. The results of the experiment are below:

Table 7: Energy cost on adaptor on and off

Experiment number	Duration	Times
1	1’10’’	10
2	1’13’’	11
3	1’11’’	10
4	1’10’’	10
5	1’09	10
6	1’10’’	11
7	1’12’’	11
8	1’08’’	10
9	1’09’’	10
10	1’09’’	10

Table 7 shows us the results of these experiments. From this table we can calculate it will cost about 0.1% energy of the battery power on simply turning on and off the WLAN interface. In additional, the average time for power simply cycling is 6.8 seconds. When the WLAN interface simply on costs 0.1% energy of the battery power, it will take 9 seconds. That means turn one and off WLAN interface costs more energy than keeping WLAN interface on in 9 seconds. So when the time users’ wait is more than 9 seconds we should turn on/off the interface to save energy.

The total energy cost of the user’s mobile device will be from upload/download files (includes reading from/writing to SD card), turning on/off WLAN interface (We set the times we turn on/off the WLAN interface to n), and the WLAN interface simply on (we set the time WLAN interface simply on to t). So the energy cost should be the sum of these parts.

For uploading, the battery power remaining is:

$$Y_u = y2 - 0.1n - t/90 = 100 - 0.56x - 0.1n - t/90$$

For downloading, the battery power remaining is:

$$Y_d = y1 - 0.1n - t/90 = 100 - 0.55x - 0.1n - t/90$$

5.2.2 Energy cost of online playing

Because of the different type of the files, we divide the files to media files and un-media files, for the media files, when the wireless and the file size is possible, we can play them online with streaming. So To find what size of file is fit for playing online and what size and what situation should be downloaded.

We use windows media player as the test software to playing the streaming media files and the local source for local playing is from the SD card.

Table 8: Energy cost of online playing and local playing

File size	Time	Energy consumed (online playing)	Energy consumed (playing on local storage)
5.6 Mbits	4'00''	2%	1%
11.2Mbits	8'00''	3%	2%
16.8Mbits	12'00''	5%	3%
22.4Mbits	16'00''	6%	5%

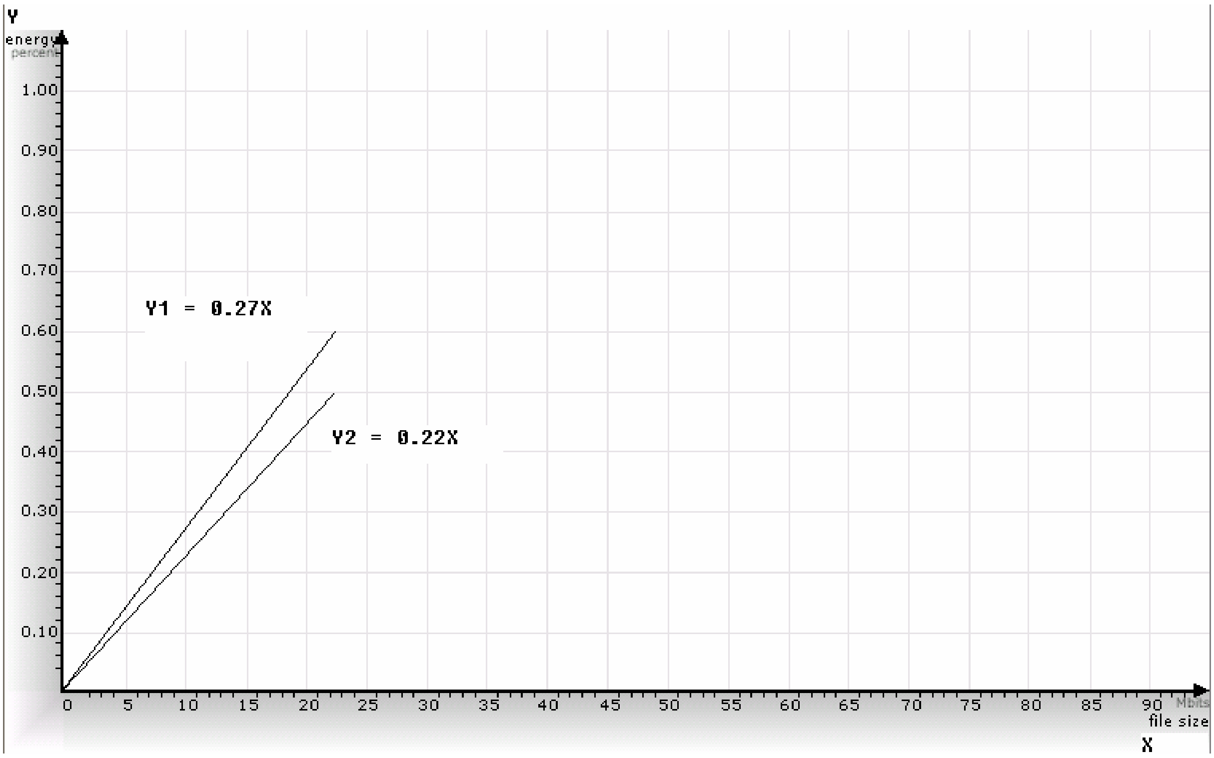

Figure 25: Energy cost on online playing and local file playing

Y1 is the curve of the energy cost of online playing
Y2 is the curve of the energy cost of playing local files

Table 8 shows the energy cost on online playing and local playing, it shows relationship between playing file size and energy usage. Figure 25 shows the energy cost curves of these two situations.

Actually, for the online playing the media files usually to be compressed by before playing, if we compress the media files before transmitting for local playing, the energy cost on transmitting will be saved. So the compression ratios will significantly affect the whole energy cost of local playing. It is necessary for us to find the influence of these compression ratios on the energy cost. To compare the energy cost by time on online playing and local file playing we should add the download energy cost to the local playing energy cost. Thus combining y2 and y1 (in figure 23), we get the function Y= 100-1.8x.

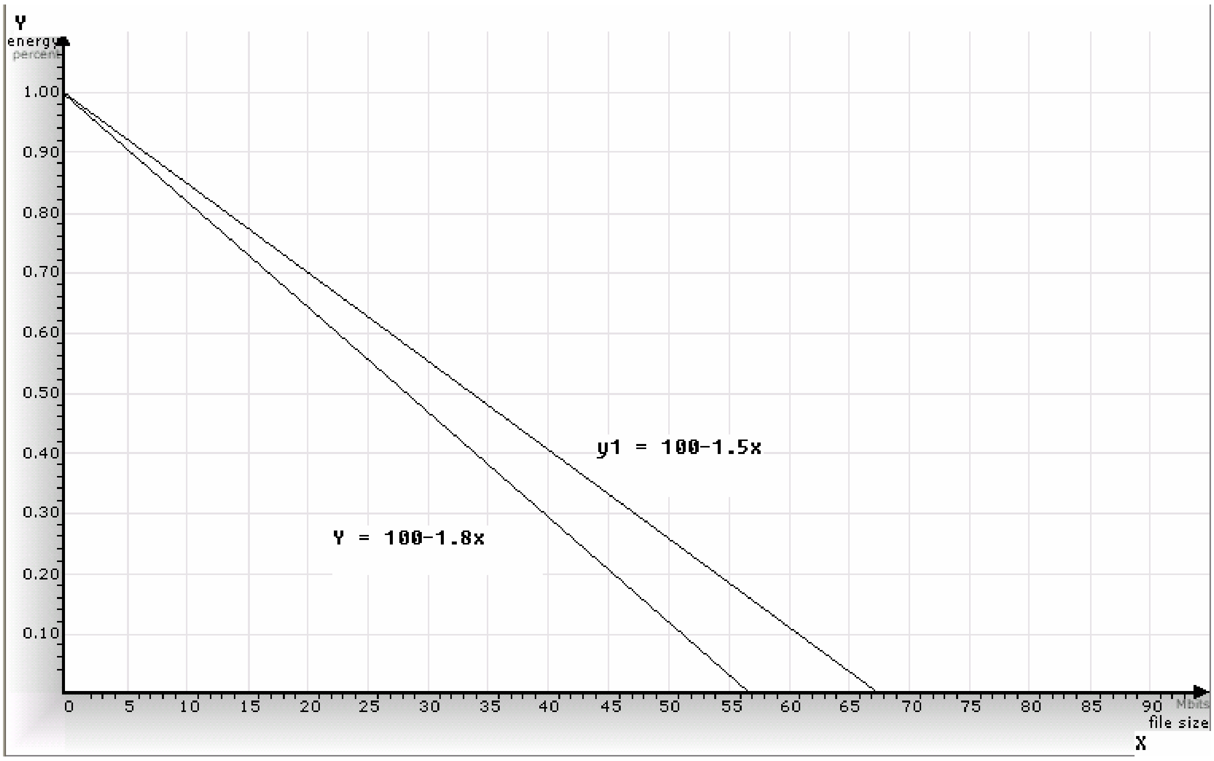

Figure 26: Battery power remaining comparison of online playing versus local playing

But if we can compress files before transfer it will cost less energy to transfer them. Then by different compression ratios, the energy cost will be different. We set the x =1 Mbits, when we playing this 1 Mbits file online, the energy cost is 1.5 percent of the battery capacity; if we play this file locally, the energy cost will be from two parts. One is from downloading it, the other is from playing it, playing the file will cost 1.25 percent of the battery capacity. So to find the compression ratios, we can make an equation as follow:

$$0.55x + 1.25*1 = 1.5*1$$

The result is x = 0.455. The compression ratios is 0.45/1 = 45.5%, so that means if the compression ratio is less than 45.5%, then local playing will cost less energy than online playing. So the compression ratio determines when we play files online or locally.

5.3 Suggestion for using these results

By the help of the GPS we can know the user’s position and the moving direction. If the position has a high probability of connection to available wireless network area, then the wireless adaptor should be on, or referring to the probability curve when the probability is high then turn the wireless adaptor on.

Given our experimental results we can derive a model for the energy which will be consumed to transfer a given file. We must also keep in mind that when the available battery power falls below 14% that the WLAN interface is automatically turned off by the Pocket PC operating system.

More formally, the cost of these processes can be settle for

C1>14%   C-C2=C1   C2<C

C: the current battery
C1: the battery after translate the files
C2: the power of the battery will cost when translate the files

```
if (C > 14%)
{
  if(C1 > 14%)
  {
     if(in the area with high connectivity of available wireless network)

            {set the wireless card state to "on"}

     else{
            set the wireless card state "off"
          }
   }else{
           set the wireless  card state "off"

          }

}else{
         set the wireless card state "off"
      }
```

By the energy cost curves, we can calculate how much energy the device will cost on use transferring the files; then based upon the file's compress ratio, we can decide if the media files should be played online or locally. We will know how many files can be transferred or how many media files to play by given the current energy. This we can manage the energy logically and intelligently.

6. Conclusions

6.1 Conclusion

The main objective of this master thesis was to understand the trade offs between having continuous connectivity and local vs. remote storage. To do so we performed the following studies:

- Survey the wireless signal on the path from a user's apartment to Wireless@KTH building and from Forum building to Wireless@KTH building. Find the probability of connecting to the available wireless network.
- Monitor the changing of the battery energy in different condition on iPAQ HP 5550 Pocket PC.
    1. the relationship between common files transferring and battery energy cost.
    2. the relationship between online file transferring and battery energy cost.
    3. the relationship between interface simply on and battery energy cost.
    4. energy cost of media files playing online and locally.
- Propose a solution scheme to consider with energy cost and wireless connectivity.

To evaluate the probability of connecting to the available wireless network on the path from user's apartment to Wireles@KTH, we conducted an experiment to test whether we can get the available wireless connection within a given time. We walk along on this path and make the wireless adaptor on every 30 seconds from anywhere and any time. The result of this experiment shows us that 9 times we can get the wireless connection out of a 30 times we turned the adaptor on. That means, the probability of connecting to the available network is 9/30 = 30% and this value is almost the same as the value we can get from figure 11. The areas when we are most likely to hear an available network are also shown in the red area in figure 19. A source of error in our survey is that we did not walk at a constant velocity, hence there is some added variance which may affect our results.

The other, we did not determine the exact position and the other parameters of AP's, for example their actual coverage, we can simply guess the general position of the AP's. There is some error in these AP positions and their coverage area and this error will affect the precision of the survey results, hence accuracy of our decision.

To evaluate the energy cost of file transferring, we measured battery power usage in a set of experiments and fitted a set of functions to these values. When we upload a 24M file to the server, the energy cost is 12%, but the function predicts that the cost should be 13%. these errors arise from two sources. Once source, if the limited precision of the measurement data, as we can only get the remaining battery power as a percentage and not in terms of Amp-hours. The other error is caused by concurrently running applications, these often affect the data. Luckily after some experiments, we found most errors are approximately 1%. Fortunately 1% is an error we can accept, because 1% of battery energy can not support the transfer of a single a mp3 file (usually 4-5 Mbits).

Although errors exist, but they are in the range we can accept and it will not make a large effect on our results. We have found a model to solve the problem we described in the beginning of this thesis. Base upon these results and the solution of our experiments, we can use these data and the solution to develop an application for describing the probability of access in the whole Kista or the other area.

6.2 Suggestion for others

This area is a new and a challenge area, particularly if we want to develop it in country or for use all over the world, we should have to build a large database for describing the wireless coverage situation (the different functions of connecting to an available wireless network) for the whole area we want. This will be a long and inconvenient process. On the other side, the available WLAN coverage actually is not sufficient for all mobile users, so GPRS will be a good complementarity for a long time.

6.3 Different design if do it again

If I were to do this thesis again, I would to get the help of some operators, get the exactl position and other parameters of every AP, then drawing an exactly map of the available wireless coverage. This would help to increase the precision of predicting when it is possible to connect to the available AP's and we could where we can get wireless directly from the coordinated obtained by GPS.

6.4 Future work

The results of this thesis are only a beginning to understand trade-offs of online-storage for mobile users. The results should be explored to be used in applications which can transfer files intelligent.

6.5 Work left to do

1. Combine the small parts to a big integrated system for the mobile device. Because the time limited, I only finished the researching part, solution design part, and some monitoring module testing. To combine these modules, developing control module and make them work together will also need some more detail working, including the detail design, coding and testing.

2. The next obvious thing to be done should be the research on file transfer module, battery and storage adaptive system and the IP automatic distribution. As noted earlier (see section 4.2), the actual transfer rates which could be achieved at each location along the paths has not been examined. This is clearly something which must be done in order to understand the actual performance versus the potential performance which this thesis explored.

The file transfer module should be focus on researching which kind transfer protocol will benefit for the different condition and whether we should compress the file before transfer.

An adaptive system will be design to analyze the different data from the different condition to help the device adapt the current condition of battery and storage. By the battery cost function we supplied, the next should develop the system for distinguishing the different kinds of files to manage set the priority for managing them.

An automatic distribution system should connect a user's device to the best AP (the AP with the smallest number of competing and with a good signal). Such an application will help the user can get the best connection and to achieve faster transfers.

References

1. Jens Zander, et al, Low Cost Infrastructure Systems, http://www.wireless.kth.se/AWSI/ , last access 2006,3,5.

2. Johan Hultell, Pietro Lungaro, and Jens Zander, Service Provisioning with Ad-Hoc Deployed High-Speed Access Points in Urban Environments, Wireless@KTH, The Royal Institute of Technology, PIMRC 2005, 15 September, 2005.

3. Yui-Wah Lee, Kwong-Sak Leung, and Mahadev Satyanarayanan. Operation Shipping for Mobile File Systems, IEEE Transactions on Computers, vol. 51, no. 12, December 2002.

4. Adaptive & Context-Aware Service, Wireless@KTH, http://psi.verkstad.net/acas . Last access: 2006,3,5.

5. JD Solomon, Mobile IP: The Internet Unplugged, Prentice Hall, Upper Saddle River, NJ, 1997, ISBN: 0-13-856246-6.

6. Inmaculada Rangel Vacas, Context Aware and Adaptive Mobile Audio, MS thesis, Department of Microelectronics and Information Technology, KTH, Stockholm, Sweden, Apirl 2005. ftp://ftp.it.kth.se/Reports/DEGREE-PROJECT-REPORTS/050418-Inmaculada-Rangel-Vacas.pdf . Last access: 2006,3,5.

7. María José Parajón Domínguez, Audio for Nomadic Users, MS thesis, Department of Microelectronics and Information Technology , KTH, Stockholm, Sweden, December 2003. http://www.imit.kth.se/info/GRU/Events/Old-Seminars/2003/20031203-11.00:LCN-Exjobb-presentation.html . Last access: 2006,3,5.

8. Antone Gonsalves, Microsoft Rivals Say ’Live’ Is ’Dead’, TechWeb News, November 01, 2005.

9. .NET Compact Framework, Microsoft Corporation, http://msdn.microsoft.com/netframework/programming/netcf/default.aspx , Last access: 2006,3,5.

10. 802.11, the Institute of Electrical and Electronics Engineers, Inc. (IEEE), http://www.ieee802.org/11/ . Last access: 2006,3,5.

11. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wceddk40/html/cxori80211ZeroConfigurationSubsystem.asp , last access 2006,3,5.

12. OpenNETCF, OpenNETCF.org, http://www.opennetcf.org/CategoryView.aspx?category=Home . Last access: 2006,3,5.

13. GPS-Global Position System, http://www.libanphone.com/gsm-glossary-of-terms.htm#n_9 . Last access: 2006,3,5.

14. Hit the road with a Bluetooth GPS receiver, CNET Networks, http://reviews.cnet.com/4520-6460_7-5135545.html?tag=txt . Last access: 2006,3,5.

15. FolderShare, ByteTaxi.Inc https://www.foldershare.com/ . Last access: 2006,3,5.

16. Define iSCSI, http://whatis.techtarget.com/definition/0,,sid9_gci750136,00.html . Last access: 2006,3,5.

17. JBOD, TechTarget, http://searchstorage.techtarget.com/sDefinition/0,,sid5_gci343350,00.html . Last access 2006,3,5.

18. Ted Coombs, Programming with C# . NET , Publisher: Thomson Learning – Published, May 1st, 2002 - Paperback , ISBN: 0766850080.

19. WiFiFoFum, AspectoSoftware, http://www.aspecto-software.com/Main/Products.htm . Last modified: 21 December, 2005. Last access: 2006,3,5.

20. H. Schulzrinne, , A. Rao, and R> Lanphier , RFC 2326 - Real Time Streaming Protocol (RTSP), IETF April 1998, http://www.ietf.org/rfc/rfc2326.txt . Last access: 2006,3,5.

21. J. Postel & J. Reynolds , RFC 959 - File Transfer Protocol, IETF October 1998 , http://www.faqs.org/rfcs/rfc959.html . Last access: 2006,3,5.

22. Norris Weimer, DOP, the non-technical description, http://gpsinformation.net/main/dopnontech.htm . Last access: 2006,3,5.

23. Steve Nix, Heading to Initialization, http://forestry.about.com/od/mappinggis/g/gps_hijk.htm . Last access: 2006,3,5.

24. GPS Glossary, 1998 - Center for Global and Regional Environmental Research University of Iowa, http://www.cgrer.uiowa.edu/cgrer_lab/gps/gpsdefs.html . Last modified: 6 November, 1998. Last access: 2006,3,5.

25. GPS data – HDOP, The Cetus GPS book, http://book.cetusgps.dk/content/hdop.html . Last access: 2006,3,5.

Table of Acronyms, abbrevations, and terms

Bit Rate	How many bits are communicated per second.
ComPort	Identifies the serial interface.
DOP	Often divided up into components. These componets are used because the accuracy of the GPS system varies. [22]
FTP	File Transfer Protocol.
HDOP	Horizontal Dilution Of Precision.
Heading	the direction in which a person or vehicle is moving [23]
ID	The number of the satellite.
IP	Internet Protocol.
NMEA	NMEA sentence is use for storing and translating the information from the GPS receiver. The function `ValidNmea` is use to judge if the device can get this sentence. If it return “true”, then it can get.
PDOP	Percent Dilution Of Position - Measure of the geometrical strength of the GPS satellite configuration. The amount of error in your position. PDOP less than 4 gives the best accuracy (under 1 meter). Between 4 and 8 gives acceptable accuracy. Greater than 8 gives poor accuracy.[24]
SNR	Signal-to-Noise Ration
Speed	Speed describes the GPS reported speed.
TCP	Transport Control Protocol.
Variation	The change of the signal from the satellite.
VDOP	Vertical Dilution Of Precision - are measure of how the satellite geometry influences the latitude and longitude data accuracy. [25]
WLAN	Wireless Local Area Network

Glossary

Wireless LAN: A WLAN is a type of Local Area Network (LAN) that uses high frequency radio waves rather than wires to communicate and transmit data among nodes. It is a flexible data communication system implemented as an extension to, or as an alternative for, a wired LAN within a building or campus. It may also be known as Wi-Fi.

Hot Spot: A hot spot is a place where you can access WLAN service. This can be for free or for a fee. Hot Spots can be inside a coffee shop, airport lounge, train station, convention center, hotel or any other public meeting area. Corporations and campuses are also implementing Hot Spots to provide wireless Internet access to their visitors and guests. In some parts of the world, Hot Spots are also known as Cool Spots.

Network infrastructure mode: A client setting providing connectivity to an Access Point. As compared to Ad-Hoc mode, whereby devices communicate directly with each other, a Profile set in Infrastructure Mode must pass data through a central Access Point. The Access Point not only mediates wireless network traffic in the immediate neighborhood, but also provides communication with the wired network.

PCs communicate directly with each other; clients set in Infrastructure Mode all pass data through a central AP. The AP not only mediates wireless network traffic in the immediate neighborhood, but also provides communication with the wired network.

Bluetooth technology: Bluetooth technology is a short—range wireless[1] technology that allows connection between devices without requiring cables. So, for example, you can connect your iPAQ to a cellular phone, a notebook computer, a printer, etc.

Appendix

Each path was traversed five times, we indicate below the times for each of these measurements when suitable access points could be heard.

On path 1

The first results:

07:34:46, 07:34:48, 07:43:48, 07:43:52, 07:43:54, 07:44:04, 07:44:06, 07:44:12, 07:44:14, 07:44:18, 07:44:20, 07:44:22, 07:44:24, 07:44:26, 07:44:30, 07:44:32, 07:44:34, 07:44:36, 07:44:38, 07:44:40, 07:44:42, 07:44:44, 07:44:46, 07:44:48, 07:44:50, 07:44:52, 07:44:54, 07:44:56, 07:45:02, 07:45:04, 07:45:06, 07:45:08, 07:45:12, 07:45:14, 07:45:16, 07:45:18, 07:45:24, 07:45:26, 07:45:28, 07:45:30, 07:45:32, 07:45:36, 07:45:38, 07:45:40, 07:45:42, 07:45:44, 07:45:46, 07:45:48, 07:45:50, 07:45:52, 07:45:54, 07:46:02, 07:46:04, 07:46:08, 07:46:12, 07:46:14, 07:46:16, 07:46:20, 07:46:22, 07:46:26, 07:46:28, 07:46:30, 07:46:32, 07:46:34, 07:46:36, 07:46:38, 07:46:40, 07:46:42, 07:46:44, 07:46:46, 07:46:48, 07:46:50, 07:46:52, 07:47:06, 07:47:08, 07:47:10, 07:47:22, 07:47:24, 07:47:26, 07:47:28, 07:47:30, 07:47:32, 07:47:34, 07:47:36, 07:47:58, 07:48:00, 07:48:04, 07:48:06, 07:48:08, 07:48:10, 07:48:18, 07:48:22, 07:48:24, 07:48:26, 07:48:32.

The second results:

09:34:58, 09:35:00, 09:43:48, 09:43:52, 09:43:54, 09:43:56, 09:44:02, 09:44:06, 09:44:08, 09:44:14, 09:44:16, 09:44:18, 09:44:22, 09:44:24, 09:44:26, 09:44:28, 09:44:30, 09:44:32, 09:44:36, 09:44:40, 09:44:42, 09:44:44, 09:44:46, 09:44:48, 09:44:50, 09:44:52, 09:44:54, 09:44:56, 09:45:00, 09:45:02, 09:45:04, 09:45:08, 09:45:12, 09:45:14, 09:45:16, 09:45:18, 09:45:22, 09:45:24, 09:45:26, 09:45:30, 09:45:32, 09:45:36, 09:45:38, 09:45:40, 09:45:42, 09:45:44, 09:45:46, 09:45:48, 09:45:50, 09:45:52, 09:45:54, 09:46:00, 09:46:02, 09:46:08, 09:46:10, 09:46:14, 09:46:16, 09:46:20, 09:46:22, 09:46:24, 09:46:28, 09:46:30, 09:46:32, 09:46:34, 09:46:36, 09:46:38, 09:46:40, 09:46:42, 09:46:44, 09:46:46, 09:46:48, 09:46:50, 09:46:52, 09:47:08, 09:47:10, 09:47:12, 09:47:18, 09:47:20, 09:47:24, 09:47:26, 09:47:30, 09:47:32, 09:47:34, 09:47:36, 09:47:58, 09:48:00, 09:48:04, 09:48:06, 09:48:08, 09:48:10, 09:48:16, 09:48:22, 09:48:24, 09:48:26, 09:48:32, 09:48:34, 09:48:42, 09:48:44, 09:48:46, 09:48:48, 09:48:50, 09:48:58, 09:49:00, 09:49:02, 09:49:04, 09:49:06, 09:49:12.

The third results:

08:25:40, 08:25:42, 08:34:34, 08:34:38, 08:34:40, 08:34:44, 08:34:46, 08:34:48, 08:34:52, 08:34:54, 08:34:58, 08:35:00, 08:35:02, 08:35:08, 08:35:10, 08:35:12, 08:35:14, 08:35:18, 08:35:22, 08:35:24, 08:35:26, 08:35:30, 08:35:36, 08:35:42, 08:35:50, 08:35:52, 08:35:54, 08:35:56, 08:36:02, 08:36:04, 08:36:06, 08:436:08, 08:36:12, 08:36:14, 08:36:16, 08:36:18, 08:36:24, 08:36:26, 08:36:28, 08:36:30, 08:36:32, 08:36:36, 08:36:38, 08:36:40, 08:36:42, 08:36:44, 08:36:46, 08:36:48, 08:36:50, 08:36:52, 08:36:54, 08:37:02, 08:37:04, 08:37:08, 08:37:12, 08:37:14, 08:37:16, 08:37:20, 08:37:22, 08:37:26, 08:37:28, 08:37:30, 08:37:32, 08:37:34, 08:37:36, 08:37:38, 08:37:40, 08:37:42, 08:37:44, 08:37:46, 08:37:48, 08:37:50, 08:37:52, 08:38:04, 08:38:06, 08:38:10, 08:38:22, 08:38:24, 08:38:26, 08:38:28, 08:38:30, 08:38:32, 08:38:34, 08:38:36, 08:38:58, 08:39:00, 08:39:02, 08:39:04, 08:39:06, 08:39:08, 08:39:12.

The forth results:

09:34:27, 09:35:29, 09:43:51, 09:43:53, 09:43:55, 09:43:57, 09:44:05, 09:44:07, 09:44:09, 09:44:13, 09:44:15, 09:44:17, 09:44:21, 09:44:23, 09:44:25, 09:44:29, 09:44:31, 09:44:37, 09:44:39, 09:44:41, 09:44:43, 09:44:45, 09:44:47, 09:44:49, 09:44:51, 09:44:53, 09:44:55, 09:44:57, 09:45:03, 09:45:05, 09:45:07, 09:45:09, 09:45:13, 09:45:15, 09:45:17, 09:45:19, 09:45:23, 09:45:25, 09:45:27, 09:45:31, 09:45:33, 09:45:35, 09:45:39, 09:45:41, 09:45:45, 09:45:47, 09:45:49, 09:45:51, 09:45:53, 09:45:57, 09:45:59, 09:46:03, 09:46:05, 09:46:07, 09:46:09, 09:46:15, 09:46:17, 09:46:21, 09:46:23, 09:46:27, 09:46:29, 09:46:31, 09:46:33, 09:46:35, 09:46:37, 09:46:45, 09:46:47, 09:46:49, 09:46:51, 09:46:53, 09:46:55, 09:46:57, 09:46:59, 09:47:09, 09:47:11, 09:47:13, 09:47:21, 09:47:25, 09:47:27, 09:47:29, 09:47:31, 09:47:33, 09:47:37, 09:47:39, 09:47:51, 09:47:53, 09:47:55, 09:48:03, 09:48:05, 09:48:07, 09:48:09, 09:48:21, 09:48:23, 09:48:25, 09:48:27, 09:48:29, 09:48:33, 09:48:35, 09:48:37, 09:48:39, 09:48:43, 09:48:47, 09:49:01, 09:49:03, 09:49:09, 09:49:11, 09:49:15.

The fifth results:

07:24:32, 07:24:34, 07:33:46, 07:33:48, 07:33:50, 07:34:00, 07:34:06, 07:34:08, 07:34:10, 07:34:14, 07:34:16, 07:34:18, 07:34:24, 07:34:26, 07:34:30, 07:34:32, 07:34:34, 07:34:36, 07:34:38, 07:34:40, 07:34:42, 07:34:44, 07:34:46, 07:34:48, 07:34:50, 07:34:52, 07:34:54, 07:34:58, 07:35:00, 07:35:04, 07:35:06, 07:35:08, 07:35:12, 07:35:14, 07:35:18, 07:35:20, 07:35:24, 07:35:26, 07:35:28, 07:35:30, 07:35:32, 07:35:36, 07:35:38, 07:35:40, 07:35:42, 07:35:44, 07:35:46, 07:35:48, 07:35:50, 07:35:52, 07:35:58, 07:36:00, 07:36:02, 07:36:08, 07:36:12, 07:36:14, 07:36:16, 07:36:20, 07:36:22, 07:36:26, 07:36:28, 07:36:30, 07:36:32, 07:36:34, 07:36:36, 07:36:38, 07:36:40, 07:36:42, 07:36:44, 07:36:46, 07:36:48, 07:36:50, 07:36:52, 07:37:06, 07:37:08, 07:37:10, 07:37:22, 07:37:24, 07:37:26, 07:37:28, 07:37:30, 07:37:32, 07:37:34, 07:37:36, 07:37:58, 07:38:02, 07:38:04, 07:38:06, 07:38:08, 07:38:10, 07:38:14, 07:38:16, 07:38:18, 07:38:24, 07:38:26, 07:38:30, 07:38:34, 07:38:36, 07:38:40, 07:38:44, 07:38:46, 07:38:48, 07:38:50, 07:38:54, 07:38:56, 07:38:58, 07:39:00, 07:39:04, 07:39:06, 07:39:12. 07:39:04, 07:39:06, 07:39:14.

On path 2

The first results:

10:25:20, 10:25:22, 10:33:36, 10:33:38, 10:33:46, 10:33:48, 10:33:50, 10:33:54, 10:33:56, 10:34:00, 10:34:06, 10:34:10, 10:34:14, 10:34:18, 10:34:22, 10:34:26, 10:34:28, 10:34:32, 10:34:34, 10:34:38, 10:34:42, 10:34:46, 10:34:50, 10:34:54, 10:34:58, 10:35:00, 10:35:02, 10:35:04, 10:35:06, 10:35:10, 10:35:14, 10:35:16, 10:35:24, 10:35:26, 10:35:28, 10:35:30, 10:35:34, 10:35:36, 10:35:38, 10:35:44, 10:35:48, 10:35:50, 10:35:54, 10:35:56, 10:35:58, 10:36:06.

The second results:

09:15:10, 09:15:12, 09:23:30, 09:23:32, 09:23:34, 09:23:38, 09:23:42, 09:23:44, 09:23:48, 09:23:56, 09:23:58, 09:24:04, 09:24:06, 09:24:10, 09:24:14, 09:24:16, 09:24:20, 09:24:22, 09:24:28, 09:24:30, 09:24:32, 09:24:38, 09:24:42, 09:24:46, 09:24:48, 09:24:54, 09:24:56, 09:25:04, 09:25:06, 09:25:10, 09:25:14, 09:25:16, 09:25:24, 09:25:26, 09:25:28, 09:25:30, 09:25:3, 09:25:36, 09:25:38, 09:25:42, 09:25:44, 09:25:50.

The third results:

07:25:11, 07:25:13, 07:33:33, 07:33:35, 07:33:41, 07:33:43, 07:33:45, 07:33:47, 07:33:51, 07:34:03, 07:34:05, 07:34:07, 07:34:09, 07:34:13, 07:34:17, 07:34:19, 07:34:21, 07:34:25, 07:34:27, 07:34:33, 07:34:39, 07:34:45, 07:34:47, 07:34:49, 07:34:51, 07:34:53, 07:34:57, 07:35:03, 07:35:05, 07:35:09, 07:35:13, 07:35:15, 07:35:21, 07:35:23, 07:35:27, 07:35:31, 07:35:33, 07:35:35, 07:35:39, 07:35:43, 07:35:45, 07:35:47, 07:35:51, 07:35:53, 07:35:55, 07:36:03.07:36:05, 07:36:07, 07:36:09, 07:36:13, 07:36:19.

The forth results:

10:45:14, 10:45:16, 10:53:36, 10:53:38, 10:53:42, 10:53:48, 10:53:50, 10:53:54, 10:53:56, 10:54:00, 10:54:06, 10:54:10, 10:54:14, 10:54:18, 10:54:22, 10:54:26, 10:54:28, 10:54:32, 10:54:34, 10:54:38, 10:54:40, 10:54:44, 10:54:50, 10:54:54, 10:54:58, 10:55:00, 10:55:02, 10:55:04, 10:55:06, 10:55:10, 10:55:14, 10:55:16, 10:55:22, 10:55:26, 10:55:28, 10:55:30, 10:55:32, 10:55:36, 10:55:38, 10:55:44, 10:55:48, 10:55:50, 10:55:54, 10:55:56.

The fifth results:

11:25:32, 11:25:34, 11:33:30, 11:33:34, 11:33:38, 11:33:40, 11:33:42, 11:33:48, 11:33:54, 11:34:00, 11:34:06, 11:34:10, 11:34:14, 11:34:18, 11:34:22, 11:34:26, 11:34:28, 11:34:32, 11:34:34, 11:34:38, 11:34:42, 11:34:46, 11:34:50, 11:34:54, 11:34:58, 11:35:00, 11:35:02, 11:35:04, 11:35:06, 11:35:10, 11:35:14, 11:35:16, 11:35:24, 11:35:26, 11:35:28, 11:35:30, 11:35:34, 11:35:36, 11:35:38, 11:35:44, 11:35:48, 11:35:50, 11:35:54, 11:35:56, 11:35:58, 11:36:06, 11:36:08, 11:36:10, 11:36:14, 11:36:16, 11:36:18, 11:36:26.

On path 3

The first results:

09:15:18, 09:15:20, 09:23:44, 09:23:46, 09:23:48, 09:24:00, 09:24:04, 09:24:06, 09:24:10, 09:24:14, 09:24:22, 09:24:26, 09:24:30, 09:24:34, 09:24:36, 09:24:42, 09:24:46, 09:24:48, 09:24:54, 09:24:56, 09:25:00, 09:25:04, 09:25:06, 09:25:10, 09:25:14, 09:25:18, 09:25:20, 09:25:24, 09:25:26, 09:25:30, 09:25:34, 09:25:36, 09:25:42, 09:25:46, 09:25:48, 09:25:50, 09:25:56, 09:26:02, 09:26:06, 09:26:08, 09:26:10, 09:26:18, 09:26:22, 09:26:26.

The second results:

08:36:12, 08:36:14, 08:44:46, 08:44:48, 08:45:00, 08:45:02, 08:45:04, 08:45:06, 08:45:10, 08:45:14, 08:45:22, 08:45:26, 08:45:30, 08:45:34, 08:45:36, 08:45:42, 08:45:46, 08:45:48, 08:45:54, 08:45:56, 08:46:00, 08:46:04, 08:45606, 08:46:10, 08:46:14, 08:46:18, 08:46:20, 08:46:24, 08:46:26, 08:46:30, 08:46:34, 08:46:36, 08:46:42, 08:46:44, 08:46:46, 08:46:48, 08:46:54, 08:47:00, 08:47:02, 08:47:04, 08:47:08.

The third results:

09:17:10, 09:17:12, 09:25:38, 09:25:40, 09:25:48, 09:26:00, 09:26:04, 09:26:06, 09:26:12, 09:26:14, 09:26:20, 09:26:24, 09:26:30, 09:26:34, 09:26:38, 09:26:42, 09:26:46, 09:26:48, 09:26:52, 09:26:56, 09:27:00, 09:27:04, 09:27:06, 09:27:10, 09:27:14, 09:27:18, 09:27:20, 09:27:24, 09:27:26, 09:27:32, 09:27:34, 09:27:36, 09:27:42, 09:27:46, 09:27:48, 09:27:50, 09:27:56, 09:28:04, 09:28:06, 09:28:08, 09:28:10, 09:28:14, 09:28:26, 09:28:26 09:28:30, 09:28:36, 09:28:40, 09:28:42, 09:28:50, 09:28:52, 09:28:56, 09:29:04.

The forth results:

08:13:13, 08:13:15, 08:21:41, 08:21:43, 08:21:47, 08:22:01, 08:22:03, 08:22:05, 08:22:11, 08:22:13, 08:22:21, 08:22:23, 08:22:25, 08:22:27, 08:22:31, 08:22:33, 08:22:37, 08:22:41, 08:22:43, 08:22:49, 08:23:01, 08:23:03, 08:23:05, 08:23:09, 08:23:13, 08:23:15, 08:23:19, 08:23:23, 08:23:25, 08:23:27, 08:23:31, 08:23:35, 08:23:41, 08:23:45, 08:23:49, 08:23:51, 08:23:55, 08:24:03.

The fifth results:

09:15:23, 09:15:25, 09:23:41, 09:23:45, 09:23:47, 09:24:01, 09:24:05, 09:24:07, 09:24:13, 09:24:15, 09:24:21, 09:24:27, 09:24:31, 09:24:37, 09:24:39, 09:24:43, 09:24:45, 09:24:47, 09:24:57, 09:24:59, 09:25:01, 09:25:07, 09:25:09, 09:25:15, 09:25:17, 09:25:19, 09:25:23, 09:25:27, 09:25:29, 09:25:31, 09:25:35, 09:25:37, 09:25:43, 09:25:47, 09:25:49, 09:25:51, 09:25:55, 09:26:03, 09:26:07, 09:26:09, 09:26:11, 09:26:19, 09:26:21, 09:26:27 09:26:43, 09:26:47, 09:26:49, 09:26:51, 09:26:57, 09:27:03, 09:27:05, 09:27:13.

COS/CCS 2006-8

www.kth.se