Project no. 035003

u-2010

Ubiquitous IP-centric Government & Enterprise Next Generation Networks Vision 2010

Instrument: Integrated Project

Thematic Priority 2

D4.2.3 Report on the Mountain Rescue Service Trial

Due date of deliverable: 31st October 2009

Submission date: 15th December 2009

Start date of project: May 1st 2006 Duration: 36 months

Organisation name of lead contractor for this deliverable: Lancaster University

Revision: v1.0

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	✓
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Abstract

The document presents a report on the Mountain Rescue Service trial. Using the trial methodology defined in D4.2.2, each component of the Mountain Rescue Service prototype solution was tested the results are reported here. These results are analysed and the success or otherwise of the overall solution is evaluated against the original requirements from WP1.

Keywords:

Mountain Rescue, Mobile Networks, MANEMO, Backpack Routers, Presence Management, Video Service, Voice Service, VoIP.

History of Change

Issue	Status	Date	Details	Responsible
v0.1	Draft	16/09/09	General document skeleton and ToC.	Martin Dunmore
v0.2	Draft	25/09/09	Added voice service tests	Lee Howarth, Martin Dunmore
v0.3	Draft	05/10/09	Added Evaluation chapter and some Presence Management Tests	Panagiotis Georgopoulos, Martin Dunmore
v0.4	Draft	07/10/09	Added MANEMO tests, some backpack router tests and some camera tests	Panagiotis Georgopoulos, Ben McCarthy, Martin Dunmore
v0.5	Draft	27/11/09	Edited Trial Methodology, Added input in Presence Management System Tests, edited TOC and restructured document	Panagiotis Georgopoulos
v0.6	Draft	27/11/09	Edited Trial Methodology, Added input in Backpack Router Tests, edited TOC and restructured document	Ben McCarthy
v0.6.5	Draft	01/12/09	Completed the evaluation against requirements chapter	Martin Dunmore
v0.7	Draft	01/12/09	Completed presence Management System Tests and Command and Control Tests Chapters, added input on Chapter 11	Panagiotis Georgopoulos
V0.8	Draft	02/12/09	Completed Chapter 4 and Chapter 9	Ben McCarthy
V0.8.5	Draft	04/12/09	Completed Chapter 11 and Chapter 5	Panagiotis Georgopoulos, Ben McCarthy
V0.9	Draft version for internal Review	05/12/09	Writing of Introduction and Executive Summary. Revised whole document. Deliverable ready version for internal review	Panagiotis Georgopoulos, Ben McCarthy
V1.0	Final Version	14/12/09	Revised document based on internal review comments by Harold Linke and Kate Yeadon	Panagiotis Georgopoulos

Table of Contents

List of Figures

List of Tables

Executive Summary

Deliverable 4.2.3 'Report on the Mountain Rescue Service Trial' is the companion deliverable of Deliverable 4.2.2 'Prototype Mountain Rescue Service Trial'. Effectively the purpose of Deliverable 4.2.3 is to document the results and findings gathered whilst carrying out the tests involved in the mountain rescue service trial which was introduced in deliverable 4.2.2. The tests carried out and described in this document are designed to highlight the suitability (and possible failings) of every aspect of the software and hardware developed to support the Mountain Rescue service scenario.

As part of our efforts to support mobile network communications in this complex and difficult scenario we have developed a plethora of different hardware components, protocol designs and implementations and software solutions. In this deliverable we aim to provide a comprehensive review of all of these different components by testing each individual item and by comparing their properties and capabilities against the requirements of the mountain rescue teams they are designed to support.

Deliverable 4.2.2 'Prototype Mountain Rescue Service Trial' presents an overview of the various service components that form the prototype implementation of the Mountain Rescue Service trial. This document provides an overall picture of which components have been implemented, how they work together, the methodology of the trial and our findings. Specifically, we test and analyse the functionality of the following components:

- Presence Management System (PMS)
    - Software designed to transmit location updates from incident area to the headquarters.
- Backpack Routers
    - Hardware devices designed to automatically deploy communications infrastructure across rescue incident areas.
- Backhaul Internet Connectivity Options
    - Wide Area communication links to transmit data to and from the incident area over the Internet.
- Command and Control Software (CaC)
    - Software designed to help mission coordinators in all aspects of their role.
- MANEMO Mobile Networking Protocol
    - Protocol designed to automate all aspects of the communication network infrastructure setup and maintenance.
- Voice & Video Communication Service

Each of these separate components plays an important role in our overall mountain rescue communication package and collectively they piece together to offer a comprehensive solution to mountain rescue teams in general.

1. Introduction

Mountain Rescue is one of the most challenging emergency scenario for which to implement ICT solutions for the emergency workers. When utilising new concepts in mobile networking the communications network becomes a technology enabler for new applications such as real-time monitoring and management, VoIP, video streaming and telemedicine.

The u-2010 project has designed and implemented solutions to cater for widely dispersed and mobile emergency workers operating in remote geographical areas based on a rapidly-deployed dynamic communications infrastructure. The Mountain Rescue service scenario is described in the u-2010 deliverables D1.1.1 Reference scenarios based on user studies [1], D1.1.2 Functional requirements for networks and services [2] and D4.2.1 Report on the Mountain Rescue Service Concept [8].

Deliverable 4.2.3 'Report on the Mountain Rescue Service Trial' presents, in detail, the numerous different tests and experiments carried out in order to ascertain the suitability of all of the aspects of Lancaster University's Mountain Rescue solution designed to facilitate mountain rescue operations. These results cover the many different aspects we have had to consider, including the underlying networking solution and communications in general, the PMS client and the Command and Control (CaC) centre software with all the services that it can provide, in addition to specialised applications for use in this scenario such as "Push-to-Talk" VOIPv6.

The rest of this document is structured as follows; the chapter, that follows immediately, briefly describes the aims of the methodology that we followed in our trials of the different components of our solution. Chapter Three discuses the tests that we carried out for the Presence Management System (PMS) client application and provides results regarding its performance on-field and on-mountain rescue tests. It also mentions the intermediate actions that we took to improve the performance of the PMS whilst finalising its implementation. Chapter Four gives an overview of the backpack router tests and in particular describes the results from in-field operation, battery lifetime and reliability testing, as well as the results from range connectivity tests. Chapter Five discusses the results from satellite and backhaul links tests. Chapter Six illustrates results from our Command and Control software testing. It breaks down all the different services provided by CaC and provides the results of our additional tests as well as rescuers' feedback from using the software. Chapter Seven illustrates and discusses MANEMO tests on the different flavours of MANEMO. Chapters Eight and Nine describe Voice and Video tests respectively, first individually and then using our MANEMO protocol. Chapter Ten provides an evaluation of our Mountain Rescue Solution against the original requirements that were set at the beginning of the project. Finally, Chapter Eleven describes our recommendation for further work that could be carried out to improve the Mountain Rescue Solution devised for the u-2010 project.

2. Trial Methodology

The purpose of the Mountain Rescue service trial is to verify the operational working of the prototype implementation and to validate that the technical and user requirements, specified in D1.1.2 Functional requirements for networks and services [2], have been met.

The Mountain Rescue service trial is conducted in two 'flavours'. The technological aspects of hardware, software and network protocols are tested by technical staff in both laboratory and field environments. Meanwhile the operational aspects are tested in the field, first by technical staff and second by end users, that is, members of the Cockermouth Mountain Rescue Team (CMRT).

The high-level objectives of the trial can be summarised as:

- Establish and maintain successful network connections from the mountains to HQ
- Network connections can be deployed rapidly
- No network or device configuration required by users
- Reasonable mission lifetime across the system
- Connectivity maintained when roaming in mountains
- Sufficient voice service across network
- Sufficient video and image service across network
- Successful presence management / localisation service across network
- Suitable Command and Control (CaC) backend solution

We therefore conduct technical oriented tests of all the individual prototype systems before conducting technical and user oriented tests of the integrated systems.

2.1. Presence Management Service Tests

The main objective of testing the Presence Management Service (PMS) is to verify that location updates can be sent from user devices in the mountains to the software hosted at the Team HQ. Further objectives include the ability for the client software to use the best available communications medium and for the system to recover when periods of no connectivity are seen.

The testing of the PMS consisted of three phases comprising initial lab tests, preliminary field tests and on-mountain tests with the Incident Area Network (IAN) connected to the Team HQ. The on-mountain tests were undertaken in the region that the CMRT operates in and during those we have consolidated the test parameters from which the results will be analysed:

- PMS client module synchronisation with GPS satellites
    - average time taken from activation to give a location
    - differences between devices used
    - how stable are the GPS signals in mountains?
- Reported GPS locations verified for accuracy

  - checked against Garmin readings
  - checked against map readings
  - checked against Google maps (server end)
- Location updates sent using Wi-Fi when available
  - client monitors connection to server to verify it is reachable
  - client swaps to using SMS when Wi-Fi is unavailable or server is unreachable over Wi-Fi
- Client stores all locations and timestamps
  - client updates server after periods with no connectivity available
- Determine how many updates are lost
  - Wi-Fi vs. SMS reliability

The results undertaken from these tests are reported in Chapter 3 of this document.

2.2. Backpack Router Tests

The Backpack routers perform a crucial role in the IAN of the Mountain Rescue service trial. As well as effectively extending the coverage of the network on the mountainside, they also provide access to that network for individual devices. As the Backpack routers are a prototype design developed by in-house at Lancaster University, it is therefore important to test that they meet their design objectives and satisfy the requirements of the mountain rescue team in general. The requirements for the Backpack routers and their design objectives can be summarised as:

- Size. The router enclosure must be small enough to fit easily inside a backpack compartment.
- Weight. The router must be light enough to be carried by rescue workers on long search and rescue operations.
- Boot time. The time taken for the router to be usable after it has been switched on should be as fast as possible.
- Easy to use. The user should not have to perform any configuration nor need any significant training to operate the router.
- Battery lifetime. The router should be able to operate for several hours in order to be useful for search and rescue operations.
- Reliability. The router should not reset or hang.
- Vibration and shock. Shocks and vibrations from walking, running and climbing should not affect the operation of the router.
- Weather resistance. The router should be resistant to weather conditions i.e. wind, rain, frost and sunshine.
- Effective range. The range that the radio hardware inside the routers can cover.

To confirm the suitability of the Backpack routers we carried out a number of laboratory and field tests to verify these goals (which we performed throughout the summer of 2009) as well as ascertained feedback about the devices in general from mountain rescue workers. The results of these tests are reported in Chapter 4 of this document.

2.3. Satellite and Backhaul Link Tests

The satellite and backhaul links connect the Incident Area Network to the Team HQ, allowing communication from the mountains to the Team HQ. The main objectives of these links are:

- Satellite dish and receiver can be setup and configured rapidly.
    - time to setup dish, stand and receiver
    - satellite synchronisation time
    - time for first packet to be routed from arrival at location
- Size and weight of satellite equipment
    - storage requirements
    - problems with dish size (high winds)
- Location requirements
    - Line of Sight (LoS) to satellite
    - Estimated required distance from mountainside for LoS
- Data performance of satellite link
    - in different weather conditions
- Radio backhaul links can be setup and configured rapidly
    - time to setup antenna, stand and receiver
    - antenna pointing and link synchronisation time
    - time for first packet to be routed from arrival at location
- Size and weight of radio equipment
    - storage requirements
    - can the remote relay be easily carried?

The results undertaken from these tests are reported in Chapter 5 of this document.

2.4. Command and Control Software Tests

The command and control (CaC) software is located at the Team HQ and provides a central point for various services including the Alarm Service, the PMS, Video Service and GIS. The objective of these tests is to verify that these services are operating satisfactorily:

- AlarmTILT Integration
    - Emergency operations can be launched and terminated using AlarmTILT

    - Available members are contacted using AlarmTILT
    - Software monitors responses from members and notifies them upon closure.
- GIS
    - Mapping engine displays all OS details
    - Maps have zoom in/out, scroll and rescale capability
- Presence Management Service
    - Mapping and navigation using GIS is accurate
    - All rescue worker movements are logged
    - Missions can be replayed from logs
- Instant messaging
    - Messages can be sent and received using IPv6 and SMS
- Video and Picture service
    - Web service showing video streams and pictures from remote cameras controlled from CaC software

Discussion of the results undertaken from these tests are reported in Chapter 6 of this document.

2.5. MANEMO Tests

The MANEMO protocol suite we have implemented is the core technology behind the role of the backpack routers. The successful operation of MANEMO is therefore critical to the success of the Mountain Rescue service trial. In summary, the tests of the MANEMO software are:

- Ensure the MANEMO protocols operate without crashing or hanging the router
- Verify that MANEMO can operate without user configuration or intervention
- Verify that MANEMO is able to connect to networks that have not been pre-configured
- Verify that MANEMO can provide routing between the Mobile Command Post and the WAN via the IAN
- Handover management
    - Certify the handover manager can monitor all available connectivity options
    - Ensure the handover manager monitors Layer 3 connectivity to its Home Agent
    - Ensure the handover manager does not connect to incorrect, undesirable or sub-optimal networks
    - Examine how optimal the behaviour of the handover manager is
    - Conduct handover latency tests to determine impact on voice and video services
- Examine the effect of mobile chaining
    - What are the realistic possibilities for extending the IAN from the Mobile Command Post?

- How far away from the Mobile Command Post given n intermediate Mobile Routers?

- Identify any scenarios (however unlikely) that MANEMO cannot solve or where MANEMO is not the optimal solution.

MANEMO test results are reported in Chapter 7 of this document.

2.6. Voice Service Tests

The objectives of the VoIP voice service tests, results of which are reported in Chapter 8, are as follows:

- Identify how well the VoIP service performs using numerous wireless hops and different backhaul connectivity options.
- Identify the effect of signal degradation, channel interference and link utilisation on the VoIP service performance.
- Determine if it is possible to run a VoIP service in a MANEMO infrastructure and how feasible this is.
- Compare the bespoke voice service with a CoTS solution (Linphone).
- Identify situations where QoS may be suitable or even necessary in order to achieve an acceptable VoIP service.
- Determine whether push-to-talk emulation provides a more robust voice service than an open access, full duplex system. What is the optimal trade-off between user familiarity, system robustness and feature richness?

2.6.1. Test Procedures

All voice service tests are carried out using pre-recorded audio files in addition to unscripted user conversations. Using pre-recorded audio files allows a more exact measurement and analysis of data such as packet latency and loss, since the content and semantics of the data is known in advance. Repeating the tests with unscripted user conversations allows us to gain a more qualitative insight into the voice service performance based on user experience.

In order to make a comparison with a CoTS VoIP solution, we needed to find a non-commercial IPv6-capable VoIP application. This application has to be IPv6 capable in order to operate over MANEMO as required. Ideally, this application would run in the Windows Mobile environment so the same client devices could be used which would eliminate any hardware or OS performance differences. Unfortunately, no suitable application could be found that matched the criteria exactly. Ultimately, Linphone was chosen as the comparison application as it had all the desired functionality but only varied in the device and OS that it operates on.

This means all tests are performed with PDAs running Windows Mobile and the custom VoIP service and with Linux laptops using the Linphone application.

All tests are performed with mobility in mind. The devices are only connected via a wireless access point and varying scenarios of backend connectivity. They move in a pattern designed to push them in and out of connectivity to see what happens in each situation. At first, the devices start close to the access point and then are taken close to the edge of signal range. They are then taken out of range before being slowly brought back into range again. The final point is to attempt to place the device right on the cusp of the wireless signal to observe the effect on the VoIP service.

RTT measurements for different test scenarios are taken using a ping utility. Packet data at the endpoints is recorded using the Wireshark network measurement tool in order to compare data rates and packet losses. Ideally, this would be on the clients themselves but this is not possible with the Windows Mobile PDAs. Therefore, data is captured at the first and last router the data passes through in order to compare loss on the backhaul. Sequence numbers for voice data will be used to determine packet loss in the test scenarios. The tests are performed with as little network traffic as can be achieved in order not to affect the results. The only exception to this is when background traffic is injected into the network to see if the applications could benefit from QoS.

2.6.2. Test Scenarios

This initial testing stage consists of using a single Linux PC with a wireless network adaptor to broadcast an IPv6 enabled wireless network. The devices we have chosen to use, 2x Windows Mobile PDAs running the custom VoIP application and 2x Linux laptops running Linphone, are set up ready to connect to each other. On the Linux PC, Wireshark is used to log to all packets from the connected devices. Each pair of devices are then connected in turn and used to run the tests while the data is recorded. The purpose of this stage is just to test both applications function correctly with no errors and to give a base for data rate comparison for later tests. The topology of the network set up being used for this stage can be seen in Figure 1.

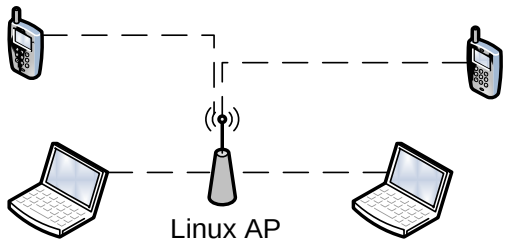

Figure 1 : Voice Service - Local Initial Testing

After the initial set of tests, the focus moves to a more complex sets of tests with gradually increasing phases of complexity. The next phase is to see how the applications handle a simple wireless network situation where there is an Ethernet connection as a backbone link. The setup consists of a Linux PC with a wireless network adaptor acting as an access point connected to an Ethernet backbone which has a connection to a Cisco wireless access point. The entire network is IPv6 enabled in order to support MANEMO that is used later. This setup can be seen diagrammatically in Figure 2.

In this stage a test using ping6 to determine RTTs between the devices is performed. This is followed by the basic audio tests which consist of the playing out of the audio files and users conversing using the applications. These are close to the access points so the effect of the wireless determination is not a considerable factor. This allows results to be taken about how the VoIP is coping with the backhaul medium and its effect. Once these are completed, mobility tests take place. Firstly with one device from each pair moving as described and then with both devices moving in this fashion. This will show the effect of the wireless signal and quality on the voice applications compared to the results they had with strong signal and no loss in the previous test. When these have been completed, the same set of tests are performed again, this time running IPERF over the backhaul connectivity medium. In this case, UDP packets are be generated in ever increasing frequency to simulate heavy traffic on the backhaul. This is done in increments until the VoIP degrades so much that it is unusable.

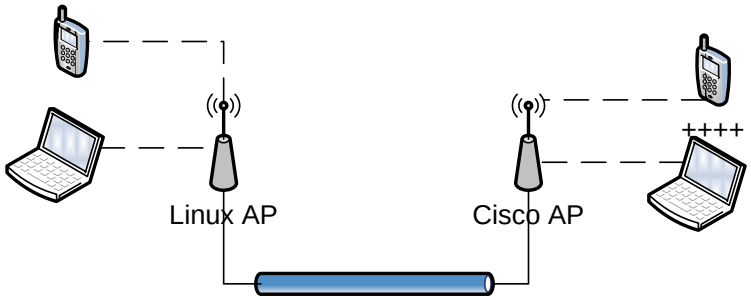

Figure 2 : Voice Service – Local Base Tests

In the following stage we run the same set of tests but with a more complex backhaul medium. Instead of an Ethernet backhaul link, the global Internet is used. In order to give us a suitable number of hops an IPv6-in-IPv4 tunnel is used with a provider that is a certain topological distance from the UK. This is done for two purposes. It introduces the random traffic element to the tests so they can be compared to previous tests where the traffic was controlled and increasing the number of network hops the traffic has to go through, increasing the traffic delay and the chance that the packets will be lost in transit. Firstly, a tunnel is established from Lancaster to London (or another nearby European city) in order to increase the hops and delay by only a short amount to emulate a standard VoIP call in the UK. After this, the tests are then repeated with a much longer tunnel (e.g. to Hong Kong) in order to get a long a delay as possible and a very high latency. This helps to emulate a long distance VoIP call. Finally, running the tests involving a satellite link, using the Astra2Connect service, gives us a valuable set of results for VoIP calls when a satellite backhaul is the only option (quite possible in a Mountain Rescue context). Again this set of tests is run without MANEMO present so we have a base comparison for these scenarios when the MANEMO tests are run using the same situations.

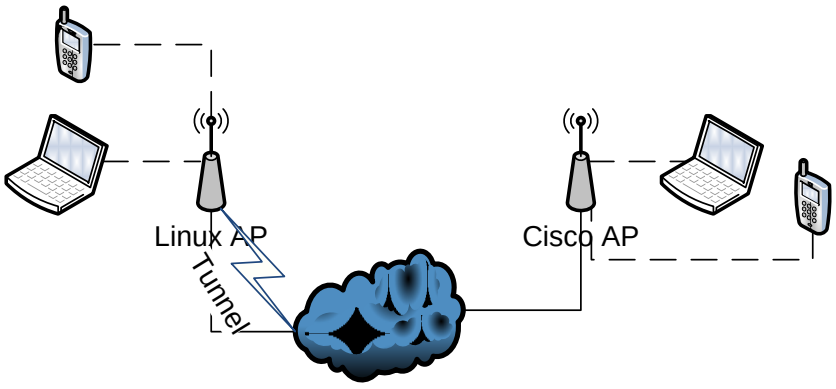

Figure 3 : Voice Service - Internet Tests

Once this stage has been completed, we should have a solid baseline for comparison when we introduce MANEMO. These tests are part of the systems integration tests and so are described in section 2.7.

2.7. Voice Service tests over MANEMO

Further to the elementary Voice Service tests, further tests over a MANEMO infrastructure are required to try to determine if a VoIP-like service is feasible in a MANEMO environment.

This consists of two separate backpack routers (representing separate mobile networks), with the user devices connected to these routers. In addition, a Home Agent is located on the network, which provides a mobility service for both backpack routers. The setup of this can be seen in Figure 4.

In this setup the same standard set of tests is run as described for previous Voice Service tests. The initial set of tests is compared to the tests from previous stages to see what effect the MANEMO protocol set has on the VoIP traffic and if any issues have arisen over the wireless connection. The extra tests are concerned with movement of the backpack router to which the devices are connected. For the first set of tests, the user devices will stay close to their backpack routers so wireless packet loss is limited and the backpack routers (not the user devices) will be moved in the mobility pattern described earlier. This allows us to see how MANEMO itself affects the voice connections, as the actual user devices (PDA's and Laptops), will maintain the same wireless connection to their associated backpack router. The final test for this stage is to get MANEMO to perform a network handover of a backpack router. This involves one backpack router moving out of range of the Linux AP and connecting to the Cisco AP.

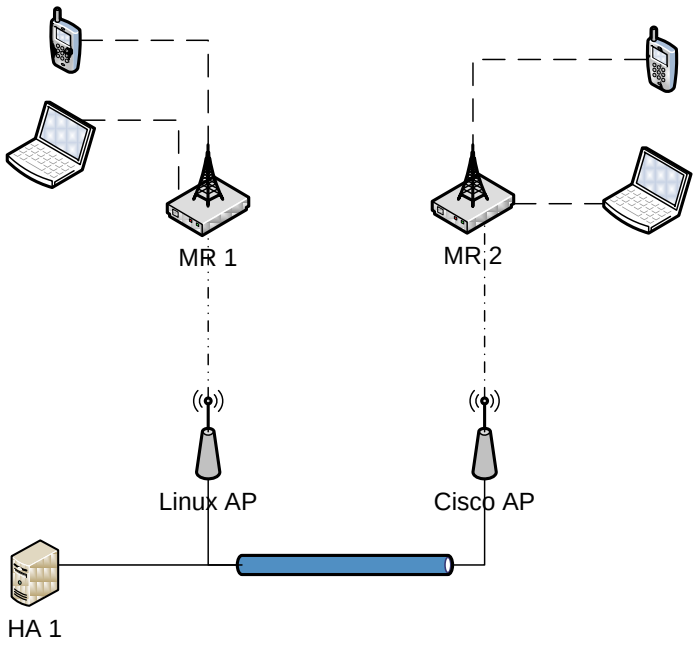

Figure 4 : Voice Service and MANEMO

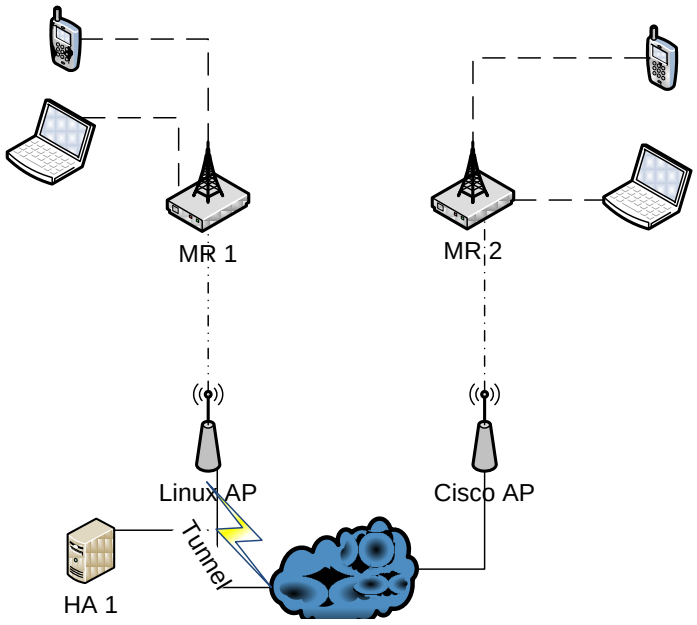

Figure 5 : Voice Service and MANEMO – Long distance

Additional complexity is introduced by adding a large delay link to emulate a VoIP call over a large distance (Figure 5). This is accomplished by adding a satellite link (and the necessary IPv6-in-IPv4 tunnel) at the egress of one access point. As before, this is to introduce a large number of hops in the backhaul connectivity medium. One backpack router is able to access the Home Agent locally, while the other has to traverse the tunnel to reach it.

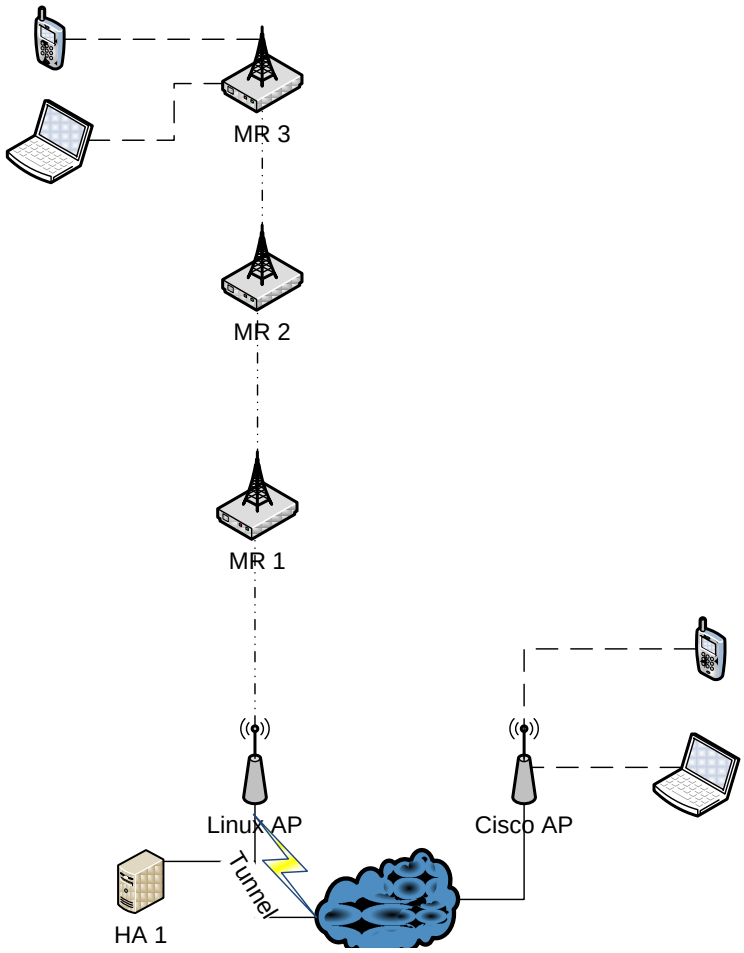

Figure 6 : Voice Service - Large Number of Wireless Hops

A final set of tests uses the same backhaul setup as in the previous tests but with a chain of mobile routers attached to one of the access points (Figure 6). This is to emulate the situation where the IAN is extended into a search region by the movement of the rescue workers carrying backpack routers. Thus, there will be multiple wireless hops that are introduced to the IAN for some end-to-end paths. Tests initially use a chain of three backpack routers, with a view to determining how long the chain can be before VoIP performance becomes intolerable.

Finally, a handover of the entire mobile router chain takes place to see what effect this has on the VoIP service. The effects of a handover with such a large amount of wireless hops should provide interesting results.

2.8. Video Services Tests

Before testing the Video Service in any complex mobile networking topologies it is first important to ascertain whether the streaming video equipment we sourced suitably support our more basic requirements, these requirements can be summarised as:

- Verify video server can transmit video over IPv6

- Verify that the observed images and video are of sufficient quality
- Observed data rates when using
    - Different encoding schemes
    - Different video resolution
- Streaming method
- Stability of the camera when attached to backpack
- Average battery uptime when streaming continuously
    - Wired vs. wireless camera
    - Video resolution vs. power consumption
    - Encoding method vs. power consumption
    - Streaming method vs. power consumption

2.9. Video Service tests over MANEMO

Similar to testing MANEMO with the Voice Service, the performance of the Video Service in a MANEMO environment will also be investigated. Although the latency requirements of the video streams are not as critical as with VoIP, there is still an open issue to investigate regarding the behaviour of the video streams in the presence of multiple wireless hops and long distance links (e.g. satellite).

Perhaps, more interesting from an evaluation perspective is the effect on video stream performance due to changes in available data rates. The unpredictable nature of multiple wireless hops in the IAN, long distance links across the WAN and handover events, all contribute to large variations in available data rates over time.

For these reasons, we use the same test infrastructures as we also use in the Voice Service tests (described in Section 2.8 above) to test the Video Service performance. As with the Voice Service tests, the MANEMO mobile routers will be arranged into the different configurations illustrated in Figures 4, 5 and 6 to test how the same network topologies (including multiple intermediary wireless hops and extended paths across the Internet) affected the capability to transmit video in general. However, the Video Service tests will concentrate on the effect that different stream resolutions and encodings behave in the presence of these network conditions. Each of these tests is then repeated for the different streaming options available (UDP, HTTP etc.).

3. Results of the Presence Management System Tests

The main objective of testing the Presence Management System (PMS) is to verify that location updates can be sent from user devices in the mountains to the software hosted at the Team HQ. Chapter 6 will then discuss the results of the Command and Control (CaC) software, operating at the team's HQ, which receives and process the location updates. Additional objectives for the PMS tests are to verify the ability for the client application to use the best available communications medium and for the system to recover when periods of no connectivity are seen. The delay of acquiring GPS readings in combination with their accuracy was also very carefully tested and evaluated during these tests, as the GPS precision in the search and rescue area is of critical importance. Furthermore, the logging functionality of the PMS was thoroughly analyzed as the recorded data allows the system to replay a mission and us to build coverage maps of the available connectivity options. Moreover, lab and on-mountain tests checked the robustness of the hardware and the software and verified whether the initial requirements of the system were met.

The testing of the PMS consisted of three phases comprising initial lab tests, preliminary field tests and on-mountain tests with the IAN connected to the Team HQ. The preliminary field tests and the on-mountain tests were undertaken in the region that the CMRT operates in, that is mainly around Buttermere, Loweswater and Lorton (Figure 7).

In all of our field/mountain tests, we disabled the GPRS functionality on the PDA devices for two reasons: 1) to avoid the GPRS to Wi-Fi swapping bug in Windows Mobile and 2) because GSM signals in the region are generally not good enough to establish and maintain GPRS connections.

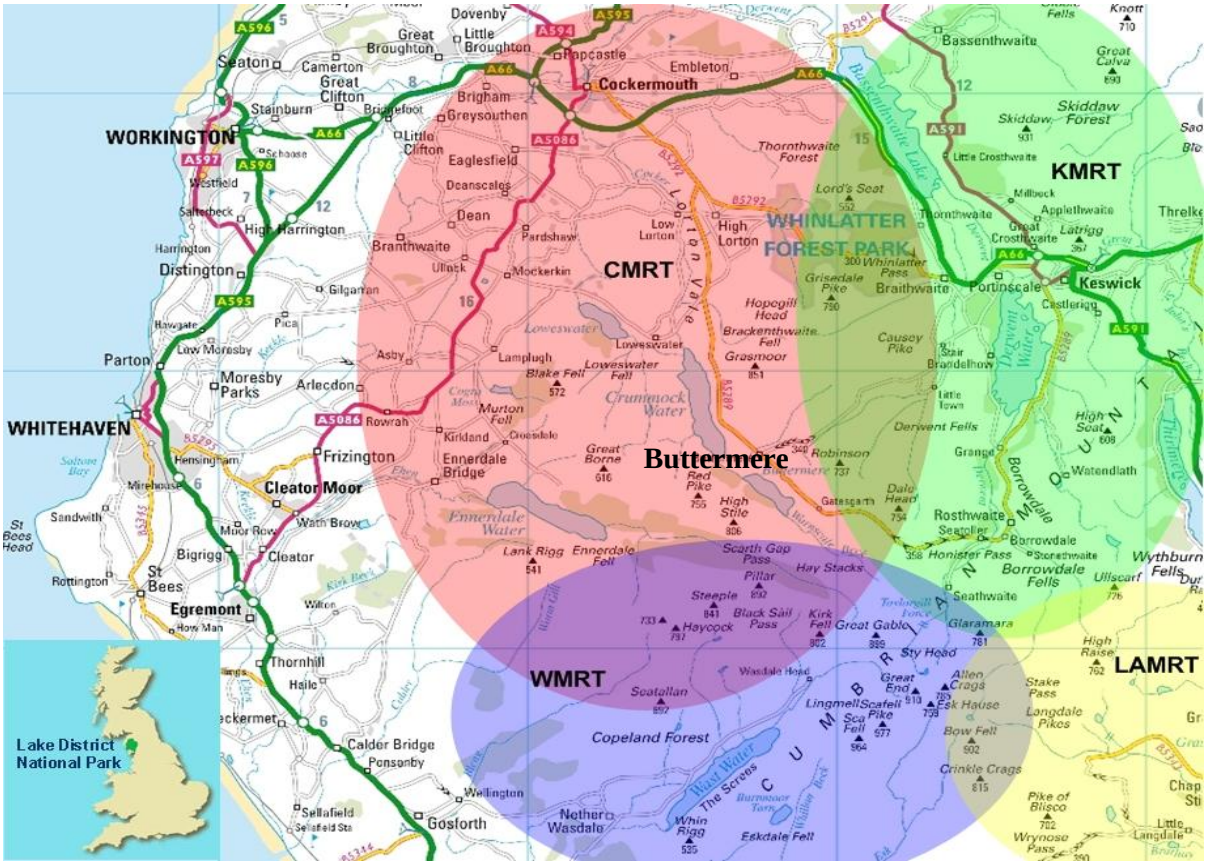

Figure 7 : CMRT Search Region

At the early stages of the project we conducted some preliminary tests around the area of Buttermere (both sides of the lake, Figure 8) and Scale Hill bridge to mainly check the GSM signal of the region, and identify good Points of Presence (PoP) that could be used to setup the Wi-Fi antennas that we acquired later in the project. In addition, we ran these tests to help us build some connectivity coverage maps and also check the GPS status around the region that the CMRT operates in. During these very important preliminary field tests the client application was initially used to log GSM signal, GPS status, GPS location and battery level and provided an initial foundation for evaluating later tests.

Figure 8 : Terrain view of the CMRT Search Region

Later in the project, and as the client application of the PMS was strenuously being implemented and tested technically in the lab, we conducted more field/on-mountain tests around the areas the CMRT operates in, highlighted in Figure 8. These tests, building upon the foundation provided by the initial tests, identified important bugs on the application that could not have been found in the lab. To be specific, the rural morphology of the terrain induced big fluctuations in both the GSM and Wi-Fi signal strength of the temporary Wi-Fi access points that we had set up and these caused the application to hang. Our exact findings were :

a) The client application was throwing unusual exceptions when it was trying to identify and log the O2 GSM signal strength around the region. As the region does not have permanent habitats O2, and in fact none of the other telecommunication providers, have invested much in telecommunication infrastructure in the region and especially around Buttermere, and thus the

client application was getting big fluctuation in the readings of the GSM signal. For example, we experienced cases when the GSM signal strength reading was 42% and when the person holding the PDA was doing a step forward, it was dropping down to 0% (no signal). This was a very useful finding, given the fact that we have not experienced it in the lab and on campus tests at Lancaster, where the GSM signal strength levels were varying between 70% and 98%.

b) Similar behaviour, although less severe, was also experienced in these field tests with the Wi-Fi signal strength fluctuating more than expected. Whilst we were successful in providing backhaul connectivity to the PMS client, by setting up temporary Wi-Fi access points in the region which were relaying data to the backhaul CLEO network, the client application was experiencing big fluctuations in the Wi-Fi signal strength readings from these local access points, when a person was roaming in the region, mainly due to the rural morphology of the terrain including cliffs, rocks, trees, lakes and rivers. The PMS client application was throwing exceptions that we have not seen before and it was "hanging" when the Wi-Fi network card of the PDAs (we had the two different IPAQ models at the time) was trying to poll and get the signal strength from nearby access points.

Both of the aforementioned findings were very thoroughly analyzed in the lab and solutions were found so that the PMS client application could handle the situations gracefully. Replicating the GSM and Wi-Fi signal strength fluctuations in the lab was of great difficulty and included various different tests and consultation of experts in communications. Our Wi-Fi tests included setting up access points with different equipment that we could tweak the level of their signal strength and propagation, varying from standard Linksys access points, to access points from atheros chipset Wi-Fi network cards handled by the madwifi Linux drivers and access points from Cisco MARs with and without their antennas. Our thorough testing and replication of on-mountain signal strength fluctuations included even the successful construction of a faraday cage (Figure 9), which is known to reduce and sometimes completely block the signal propagation. The result of all the aforementioned tests was that we were able to handle gracefully exceptions from GSM and Wi-Fi signal fluctuation readings in a way that they were not affecting the effectiveness and the overall goal of the PMS client application.

Figure 9 : Faraday Cage constructed in our laboratory

In our effort to test our signal strength fluctuation corrections on the PMS and also start testing the online and offline functionality of the PMS client (i.e. how it behaves when no connectivity is available and when connectivity is regained) two further tests were conducted in two different search and rescue regions of the Lake District. The first one was a 10 minute drive followed by a 70 minute walk around the area of Ullswater (Figure 10) and the second was a 125 minute walk and drive around the areas of Buttermere, Scale Bridge, Scale Hill and Lorton. The PMS client used both online and offline mode and managed to identify successfully the present or lack of connectivity options and send GPS updates using the most efficient connectivity medium to the CaC server backend which was located at the time at Lancaster University.

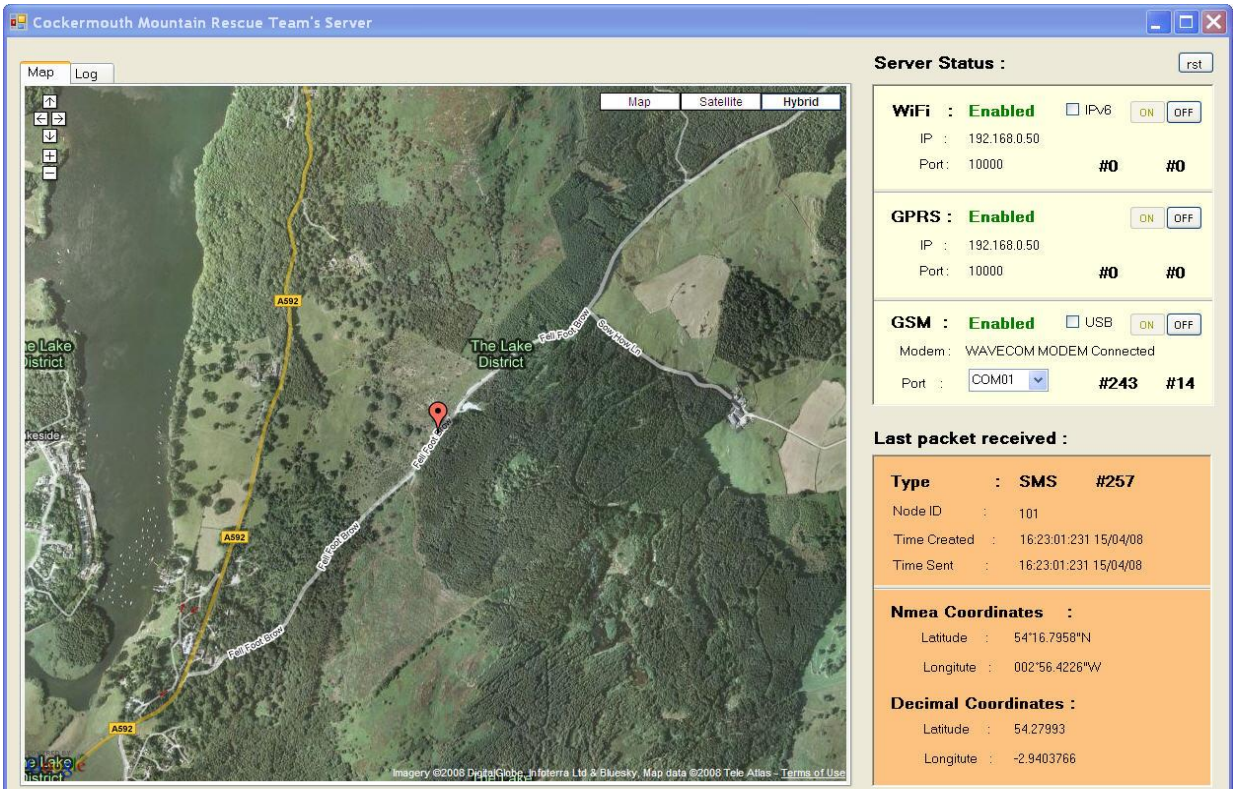

Figure 10 : CaC screenshot of the drive and walk in Ullswater

During these tests, three further important issues were identified and actions were taken to resolve them. The first one was that the standard IPAQ PDA batteries were not sufficient for further testing because sometimes they could not last for more than an hour and a half with the PMS client running at its full functionality. In other occasions, the batteries were not able to power the PDAs properly, leading to graceless turn off of the PDAs, even though the reported battery level seemed sufficient for further testing (e.g. 54%). This seemed to be a sign of "worn out" batteries. Therefore we replaced the 1200mA batteries that the IPAQ PDAs were equipped with 1800mA batteries which could last way longer and in fact they were more appropriate for real rescue missions. The second issue that we identified was about the interval SMS messages with GPS updates were sent to the CaC interface. Normally, it takes 3-4 seconds for the PDA devices to send an SMS message to the server when Wi-Fi connectivity is not available. However, in cases with low GSM signal strength or severe GSM signal strength fluctuations, which are the norm in the search and rescue region the CMRT operates in, the PDA needed up to 7-8 seconds to send and sometimes retry the transmission of some SMS messages. When the transmission interval for SMS location updates was 10 or less seconds this led to a lot of SMS messages being lost and getting loss percentages up to 45%. Further analysis of the PMS client and CaC server logs identified that although the system could cope with sending SMS updates every 10 seconds or less, the loss percentage was high enough and made it inappropriate for the mountainous region of the Lake District. Experiments identified that an interval of 20 seconds was balancing rightly our need to show location updates at the CaC whilst keeping the percentage loss to minimum (usually less than 8%). A third issue that was identified during these tests was the need to dynamically modify the interval of the transmission of GPS coordinates as there was a need of more frequent updates being sent when the rescuers were in a car driving towards a location, than when they were roaming and trying to find the casualty. This was also implemented and tested in future field tests.

As we were becoming more confident with the PMS implementation we continued doing tests around the region of Buttermere and Grasmoor Hill by complementing the connectivity options with Wi-Fi links relaying data to CLEO and with the Astra2Connect satellite link. The Wi-Fi connectivity to CLEO during these tests was realized with the use of "rapid-response" PoP at Rannerdale and Low Fell hill described in detail in Section 4.1 of D4.2.2 [9]. During these tests we were providing IPv6 Wi-Fi connectivity to the PMS clients using two 802.11b access points, which were relaying data to the Rannerdale car par PoP and from there to the Low Fell hill PoP. From Low Fell data were relayed to the Moota Hill mast, just north of Cockermouth, and then routed via the global Internet to our CaC. During these several hour tests we were able to stress the capabilities of the PMS client and check how well it behaved in regions with good, average and no Wi-Fi and GSM connectivity. Our tests showed that the client application managed to confront no Wi-Fi connectivity periods by sending SMS messages, and when GSM signal was very low or lost completely, to store GPS updates for later transmission. The PMS client managed also to transmit stored coordinates when connectivity was regained and at the same time flag the packets appropriately to inform CaC for these occasions. A very useful finding during these tests was that after 45 to 50 minute trials, the PDAs were becoming unresponsive and although the PMS application was running correctly in the background, someone could not interact very well with other PDA's functions due to low resources in Windows Mobile. This resulted in us coding a more lightweight implementation of the same functionality for the client PMS and carefully handle the threads that were created to aid Windows Mobile in gaining back their much needed memory and CPU cycles. These tests also resulted in carefully tweaking the way the PMS client application was swapping from Wi-Fi to GSM and back, in order to minimize packet loss as much as possible.

Figure 11 : Rannerdale Car Park PoP looking at Grasmoor Hill (left), Access Point at Rannerdale looking at the PoP (right)

Further on-mountain experiments were carried out on the basis of the previously described tests leading to refining more the PMS client application which had reached a very stable version and was also run on newer HTC devices (Touch Cruise) which were more powerful. Our tests showed that the PMS client application could run in any Windows Mobile 5.0 or higher device with the appropriate network configuration.

Battery life in the IPAQ devices with the 1800mA batteries when using the client application in its full potential, was generally between 3 and 4 hours. The battery life on the HTC devices was longer with observed lifetimes up to 5.5 hours of continuous use.

The logging functionality of the clients was observed to be excellent, not only for logging GPS updates, but also logging every other important detail that the PMS client could see. The very detailed GPS logging functionality allowed replaying the mission of a rescuer simply from the log of the client application, in addition to the logging and replaying functionality of the CaC. Generally, the PMS client logs timestamps with all the following, GSM and Wi-Fi signal strength, battery level, GPS coordinates, GPS status (number of satellites being seen and locked), packets being sent over Wi-Fi, GPRS and SMS, packets being recorded as offline and finally the payload of each packet.

Results from our tests also showed that the client application needed a locking functionality to prevent accidental tap on the screen of the PDA when placed in a pocket. Windows Mobile "locking functionality" was found to be very inadequate as it was very difficult for the rescuers to see the screen and find a way to unlock the device especially when being outdoors under the sunlight of extreme weather conditions. Thus, we implemented another "locking functionality" that prevents the application from accidental stop of its execution by needing to tap the stop button twice within 3 seconds. Tests were undertaken with rescuers having the devices in their pockets and the PMS client was found to run successfully even when buttons where being pressed/tapped accidentally.

The PMS client application was also tested at the Training Centre for Civil Protection and Disaster Relief, Ig, Slovenia, which resulted in a very successful demonstration with the aid of the URSZR Slovenian Rescue Team. These tests and results are not reported in this document as they are being described in D4.5.1 [10] and D4.5.2 [11].

3.1. PMS Client GPS Performance

This section focuses on the performance of the GPS module of the PMS client application run on PDAs and summarizes our results from our tests, described in the previous section. The PMS client application has been used in four different PDA devices, namely HP IPAQ 6915, HP IPAQ 914c, HTC Touch Cruise and HTC Touch Cruise T4242, with the last two having very minor hardware differences (see details in D4.2.2 [9]). Generally, the HP IPAQ devices were earlier manufactured and their GPS module appeared to be slower and view less satellites than the one on the HTC devices.

While using the PMS client on the PDAs we experienced an average of 50 second delay to get a GPS fix and start sending GPS updates from a cold start (the application is executed for the first time and the GPS fix is acquired from the first time after the device has booted). Of course, this time varies tremendously depending on the precise location and line of sight to the equator from that position. Generally, these times reduce sharply for hot starts (the application is running, the GPS module was activated before and is being enabled again) to an average of 5 seconds.

Once the GPS fix has been established in either of the different PDA devices we tested, it is rarely lost unless the device is taken indoors or in an area of steep rock outcrops blocking line of sight to the satellites. Even in dense woodland, the GPS fix tends to remain stable, albeit with less satellites synchronised. On average, we have observed between 7 and 11 satellites synchronised both around the University and in the Lake District. The PMS client can generally maintain an accurate GPS fix with 4 or more satellites synchronised. In wooded areas we have observed the number of synchronised satellites to vary between 3 and 6 (assuming the trees are the only blockage to line of sight) and we have verified that by analysing the log files.

Table 1 Time Taken for Localisation Client to Synchronise with GPS Satellites (Cold Start)

Time for Synchronisation	HP IPAQ 6915	HP IPAQ 914c	HTC Touch Cruise
Average (secs)	50	50	45
Min. (secs)	35	30	32
Max (secs)	300	180	200

Table 2 Time Taken for Localisation Client to Synchronise with GPS Satellites (Hot Start)

Time for Synchronisation	HP IPAQ 6915	HP IPAQ 914c	HTC Touch Cruise
Average (secs)	5	5	4
Min. (secs)	3	2	2
Max (secs)	32	30	18

Detailed hardware and software differences between the devices used are noted in D4.2.2 [9]. The main point to emphasize is that the newer HTC devices have better resources, are more responsive and behave better when running the PMS client. In terms of their GPS capability, the GPS module on the HTC devices outperforms the IPAQs' one, as it is faster, it provides more precise GPS coordinates and also sees and locks more satellites (we have observed up to 16 satellites being reported versus 14 on the IPAQs)

The accuracy of the GPS coordinates reported by the PMS client was verified in 2 ways:

1. Compared with the readings from Garmin GPS devices at the same location
2. Checked with Ordnance Survey Map references

The locations reported by the PMS clients were very accurate and the functionality of the client did not impose any burden to the acquisition or precision of the GPS coordinates. Assuming they were sufficient satellites, the readings reported from the PMS client corresponded with the Garmin devices and were correct according to map readings. These were verified with many methods such as running the CaC at the location where we did the described tests so that a person could see both the users roaming around a region and what was being shown on CaC at the same time, in addition to record the tests on CaC and studying them carefully later.

However, we did observe that some GPS hardware modules would attempt to give readings when the GPS signal was insufficient rather than report no (i.e. a null) position. This often caused problems with the PMS server component (CaC) as the incorrect readings would make the rescuers' tracks look unrealistic and impossible. A workaround for this was devised whereby any reading varies too much from its predecessors or is out of range of the search area is tagged as suspect and not included as a waypoint when drawing the rescuer tracks.

3.2. PMS Client Location Updates

The reliable transmission of the PMS Client location updates depends mostly on three key points. Firstly, the actual network conditions, secondly the PMS application's ability to scan the environment and identify correctly and efficiently these network conditions and thirdly, the PMS application's efficient

swap from one connectivity option to another and the actual data transmission function based on various run time criteria, described in D3.2.1 [5].

Regarding the transmission of location updates over Wi-Fi our repeated tests, both in the lab and on-mountain, aided in improving the functions handling the networking card of the PDAs, the efficient scanning of the environment for signal strength readings and also careful handling of the networking stack. To be specific, different threads were being used for scanning the environment for Wi-Fi Networks, to identifying the correct one and polling its signal strength. Additional thread is being used to maintain the TCP connection with the CaC backend and polling the socket to check if it is alive.

Results from our on-mountain tests showed that even in occasions when the Wi-Fi signal was very low (-80 to -90 dbm), the PMS client was still able to maintain the connection with the server without needing to swap to SMS. Increased reliability of the transmitted packets and minimum packet loss was also due to the TCP protocol being used after careful consideration and experimentation with UDP implementations. Results also showed that when the Wi-Fi signal is lost, there is usually no more than one GPS update packet being lost (and often there are no packets lost) as the client swaps to using SMS. This usually depends on how high the update interval is and when the connection with the server is lost. For example, if we use a 15 second interval and the Wi-Fi connection is lost on the 3rd second, the client can very easily send an SMS update on the 15th second without losing any packets. However, if the connection is lost on the 14th second, the application may not detect this on time and could still try to send the packet on the 15th second over Wi-Fi and then swap to sending SMS.

When the PMS clients lose all connectivity to the server, they store all the location updates for that period and update the server when connection is re-established. The initial results from preliminary tests of this functionality were very problematic due to the signal strength fluctuations and the client application swapping back and forth the different connectivity options according to these readings. However, thorough analysis of the logs and re-implementation of certain functionalities led to a very refined implementation of transmitting coordinates over the most available connectivity option and then storing them when no connectivity was available. When connectivity was regained, even after long periods of no available communication with the CaC, the PMS clients updated the server with all the ‘missed’ location updates. Results of testing this functionality also showed that we could update the CaC interface with a batch of stored GPS updates when Wi-Fi connectivity was regained and then split them carefully at the server backend. This functionality was also implemented in a very resource lightweight manner and tested thoroughly leading to a more efficient use of the connectivity options than transmitting stored coordinates one by one.

It is without doubt that we could not minimize packet loss down to zero, due to the very extreme networking conditions being apparent in rural terrains such as hills and mountains. In spite of the recovery capabilities of the PMS client, there are often some packets lost due to the demanding nature of the application environment.

Table 3 Packet Loss of the PMS in the Lake District

Packet Loss	Wi-Fi (TCP)	SMS
Average	1.5%	3%
Max.	2%	15%

In various tests in the Lake District, we have observed an average of 1.5% packet loss when using TCP over Wi-Fi and 3% loss when using SMS (after the refinement of the PMS implementation) which are considered excellent for the scope of our system. The PMS client was working perfectly during our latest

on-mountain tests, identifying connectivity loss, and storing packets for later transmission. Packets were transmitted successfully when the client regained connectivity. However, packets can still be lost in the network (Wi-Fi and GSM) even when the client identifies successful transmission on its end. In the worst case, we have observed an SMS packet loss of 15% due to the severe fluctuation of the GSM signal making SMS transmission unreliable. Thorough analysis proved that this behaviour is attributed to the GSM provider's network (O2) and that we could not further improve that in regions with very bad GSM signal.

3.3. Feedback From Rescuers

In general, the rescuers were delighted with how easy it was to use the PMS application. They were very happy that no device or application configuration was required by them mainly because the PMS client reads all the settings from a configuration file. A simple tap of the on button is required for the application to run, and twice tapping to stop its execution. The rescuers particularly commented on the "execute the application, place the PDA in your pocket and forget about it" fashion that we decided to maintain for the PMS client.

Rescuers from the URSZR Slovenian team, reported having problems seeing the PDA screens in strong sunlight. This meant that, as simple as the GUI was to use, the rescuers simply could not see it properly. Another comment that we received from UK rescuers was that when they wanted, they could not use the PDA easily due to the small screen and buttons they had to press. Usually rescuers wear big waterproof gloves which made use and navigation on the PDAs very difficult. The use of gloves did not present any problems in them executing the PMS client application, but presented difficulties in browsing its tabs if they wanted to see additional GPS information. Another strong requirement that they had was for the PMS devices to be waterproof and ruggedised as usually mountain rescue missions develop under very bad weather conditions, which we did not manage to meet during the timeline of the project.

The battery life of the PDAs is seen as acceptable but the rescuers would feel more confident in using the equipment permanently if battery life could be extended to 8 hours or more of continuous usage. This is not a problem in the majority of search and rescue missions since they mostly last between 2 and 4 hours. However, there are occasionally missions that can last more than 24 hours. Having PDAs that could last for 8 hours would allow them to be distributed to search parties on 8 hours shifts. Long search missions also require the PDA batteries to be recharged more rapidly or having additional fully charged batteries on stand-by to use them when required.

4. Results of the Backpack Router Tests

The Backpack router is designed to be carried by individual mountain rescue workers (as well installed into rescue team vehicles) and is a key component in the proliferation of the mobile network that is projected in the field of operation. It is therefore important to ascertain the capabilities of the prototype device we developed and determine its suitability for use in everyday mountain rescue cases. In this section we detail the testing we performed specifically to confirm the capabilities of the backpack router and the feedback we received from mountain rescue workers about the device in general.

4.1. In-field Operation

One of the primary requirements for the backpack router is to be unobtrusive in the mountain rescue worker's day-to-day operations, which ultimately means it must be very straightforward for them to use and require very little input during operations. The backpack router was designed from the beginning with this very requirement in mind and is therefore almost as simple to use and unobtrusive as it can possibly be. The physical device simply has one switch which can either be set to "On" or "Off", after switching the router on it can then be dropped into a backpack and doesn't require any further interaction. From a cold start, once the router is switched on it automatically boots up all of its appropriate system resources and then starts to automatically configure its networking interfaces. Once all hardware configurations are completed the backpack router then initialises the MANEMO protocol and its "Intelligent Handover" software, which immediately starts scanning the surrounding area for appropriate external connections to the Internet. This handover software then continues to run autonomously for the duration of an operation, constantly updating the router's understanding of its surrounding wireless networking environment and making handover decisions as and when it is necessary.

It is unlikely that the boot time of the backpack router would be of importance to a mountain rescue worker under normal circumstances (as the router could easily be started at the HQ during briefing or in a vehicle on the way to rescue mission). However in certain situations a rescue worker may be hastily added to an existing search and in this situation, the time it takes the backpack router to configure itself may be crucial. To determine the overall time it takes for the backpack router to complete its boot phase we developed a simple application that recorded the router's system time at the moment data is able to be successfully transmitted back to a HQ. By configuring the router to reset its system clock and therefore begin its start up phase with a system time of zero, we were then able to obtain an accurate boot up time. On average, in situations the initial hardware configuration took around 26 seconds to complete. At this point the router is ready to start trying to establish a global connection, then the MANEMO protocol and the handover software's start up process were seen to add a further 10 seconds to this value.

After starting up the router and placing it into a rescue worker's backpack, the next important in-field considerations become its resilience to the environment conditions it will encounter. For this purpose the device needs to be fully water resistant and also be able to resist shocks and vibrations. Our backpack router hardware is contained inside a water resistant plastic enclosure that ordinarily would be sufficiently weatherproof, apart from our current switch mechanism. At present we use a metal power switch mechanism that protrudes from the enclosure and could therefore facilitate water to enter. In future iterations of the enclosure we will attempt to solve this problem by have incorporating a completely seamless, waterproof method of switching the router on and off. The backpack router's shock and vibration resistance is good mainly because there are no moving parts used anywhere in its design. This ensures that even persistent vibrations and knocks will not affect the operation of the routers main board and interfaces, however the main concern in this respect is the stability of the internal cabling and fixings. As the router we currently use is still in the prototype stage the cabling and fixings we use are

unspecialised, off the shelf computer components. In order to properly fulfil this requirement however we would probably need to use more robust, better secured cabling and stronger internal fixings

4.2. Battery Lifetime and Reliability Testing

Once operational, the backpack router must continue to operate autonomously for as long as possible. The operational time of the backpack router is therefore inextricably linked to the lifetime of its internal battery and also to the reliability of the software running on the router. To test the battery life of the router we produced a simple application that constantly transmits traffic over each of the router's wireless interfaces for as long as it can. The application was started at the end of the boot phase and recorded the exact time that every packet was transmitted until the point it powered down. On average the current battery we use gave us 3hrs 30mins operating time under these circumstances. This is long enough to support some shorter rescue missions and is also perfectly suitable for performing demonstrations and trials, but for prolonged missions this would not be long enough. In addition to the operational time, the time it takes to then recharge the Li-ion batteries was also recorded to take an average of 3hrs 30mins. The battery that we currently use is again only a non-specialist, inexpensive, off-the-shelf product, if the backpack router were to be taken beyond the prototype stage then this power solution would be given further consideration.

In addition to determining the operating time of the backpack routers, we also tested the reliability of the MANEMO protocol that runs on the routers by transmitting prolonged data streams and recording whether the data was lost or temporarily broken at any point. In each of the tests we performed, with varying different MANEMO network topologies the data stream was seen to continue unbroken for over 48hrs. This is sufficiently long enough to support any possible scenario (and more likely, any possible battery technology we can obtain). These tests were performed solely on the MANEMO protocol however and did not incorporate the operation of the "Intelligent Handover" software. This software is still at an early stage of research implementation and is therefore still very much a work in progress at this time. Once completed, its operation could certainly represent a further possibility for software instability and these tests will therefore need to be performed again at a later date to include the handover software.

4.3. Effective Range Testing

Since the Backpack router devices interconnect with each other in order to expand the reach of the mobile network effectively when it is projected onto a mountain rescue search area, a fundamental consideration is the wireless range that each individual router can add to the network. The Backpack router can be considered as both a node in the wider mobile network and also as a hub to which individual devices connect. In the mountain rescue scenario this equates to the Backpack router connecting to other Backpack routers carried by other mountain rescue workers within range and at the same time allowing rescue workers' streaming webcams, PDA devices, GPS device, etc to connect directly to it. The more Backpack routers that can be deployed in a rescue situation the better, as each one will further proliferate the level of network connectivity available at any given time. However it is important to note that it is not necessary for each individual rescue team member to carry a Backpack router and therefore a situation may arise where only a half or a quarter of a given rescue team are carrying Backpack routers. In this case the Backpack routers deployed will provide connectivity to the individual devices carried by the rescue team member that the Backpack router is associated with and at the same time, it may also provide connectivity to the devices of another team member. For these reasons, it is therefore important to determine both the range capabilities for individual devices connecting to Backpack routers and for Backpack routers connecting to other Backpack routers.

4.3.1. Handheld Device to Backpack Router Range Tests

Measuring the capability of an individual device to remain connected to a Backpack router's Wi-Fi Access Point effectively indicates the range of coverage of a Mobile Network provided by a Backpack Router. To test this capability we connected Windows Mobile Smart Phones (HP iPaqs & HTC Touch Cruises) to the Backpack router and recorded the distances that could be roamed before the connection began to break up. To allow us to gather accurate measurements we developed a simple application that recorded a GPS coordinate log of all the area covered whilst the handheld device was able to remain connected the Backpack router and also, log exactly when and where that coverage was lost. In Figure 12 we present some of the testing we did in this area (illustrated in the familiar GoogleMaps interface, which was chosen to provide us with accurate distances between waypoints). What this figure shows is the ability for a handheld device to remain connected to a Backpack router at up to over 300 metres away from the Backpack router. It is also important to point out that in this specific test illustrated there was no clear line of sight either as there are significantly tall buildings present in this area pictured. Overall the results in these tests were very encouraging, as we had initially expected the handheld device to only be able to transmit to the Backpack router over much smaller ranges because they are relatively low power devices.

Figure 12 : Range Tests between Backpacks and Individual Device

4.3.2. Backpack Router to Backpack Router Range Tests

In addition to testing the range of the Mobile Network coverage projected around an individual Backpack router, it is also important to understand their range capabilities when interconnecting to other Backpack routers to form the on-mountain wireless network infrastructure. In this case the routers connect to each other using their onboard Wi-Fi interfaces through a 5dB omni directional antenna. Figure 13 again shows an image reconstruction of the effective Wi-Fi coverage test range, but this time for the Backpack router to Backpack router connection. On average we have been able to establish communication between

two backpack routers at up to 400 metres with near line of sight. This level of coverage, combined with the additional range achievable by handheld devices and the proliferation of network coverage that interconnecting the Backpack routers can provide is considered very positive overall. With the MANEMO approach and these levels of effective range, large search and rescue areas could be provided with high throughput wireless network infrastructure.

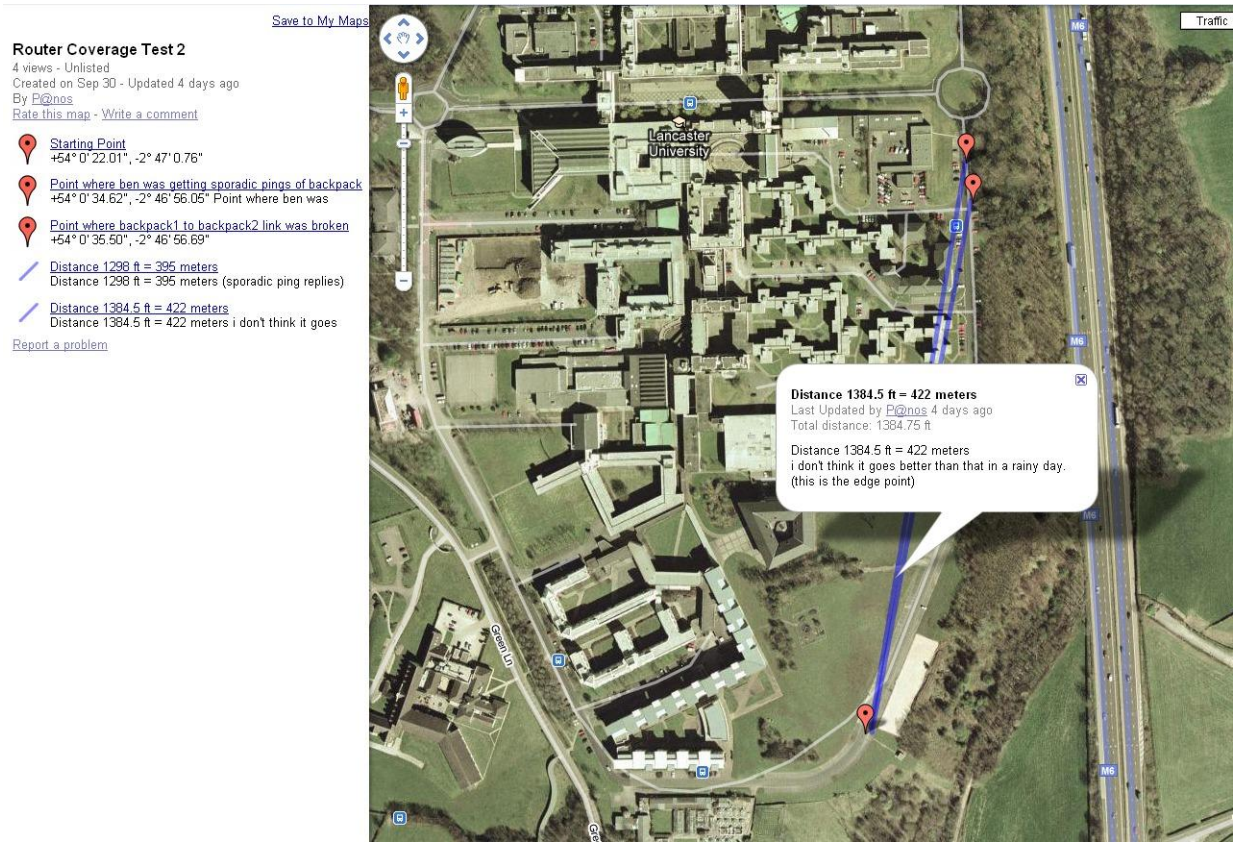

Figure 13 : Range Tests between Backpack Routers

4.3.3. Dense Woodland Range Tests

As well as straight forward, line-of-sight and near line-of-sight testing, we also carried out tests in densely wooded areas. It is obviously of importance that for any solution that is designed to operate in a mountainous area, tests should be performed at some point over this type of terrain. In dense woodland, we witnessed that the range of the Backpack routers is reduced to a certain degree (as it would be expected) but overall we were again very impressed with the actual results we achieved. In our tests we observed an average connection distance between Backpack routers of around 160 metres and an average connection distance between the handheld devices and the Backpack routers of around 100 metres. An example test configuration is shown here in Figure 14, in this area tree density was such that people become no longer visible after only around 20 – 30 metres and yet whilst out of visible range, communication could still continue over our networking infrastructure. In addition to the individual range of a Backpack router, this particular test area very succinctly highlighted the strengths of the connectivity

chaining approach of MANEMO. In this area the forest very steeply drops away (this is the point at which the first Backpack router would start to go out of range. However, since one Backpack router remained at the top of this land feature it meant that the second Backpack router (and thus its connected devices) could continue to communicate back to the Backpack router at the starting point, at the most southern point of the woodland. In particular, in this test we also carried 2-way Motorola personal radios for comparison and whilst the 2-Way radio signal broke up, the MANEMO connection stayed in place, because of its ability to forward data through the intermediary Backpack router.

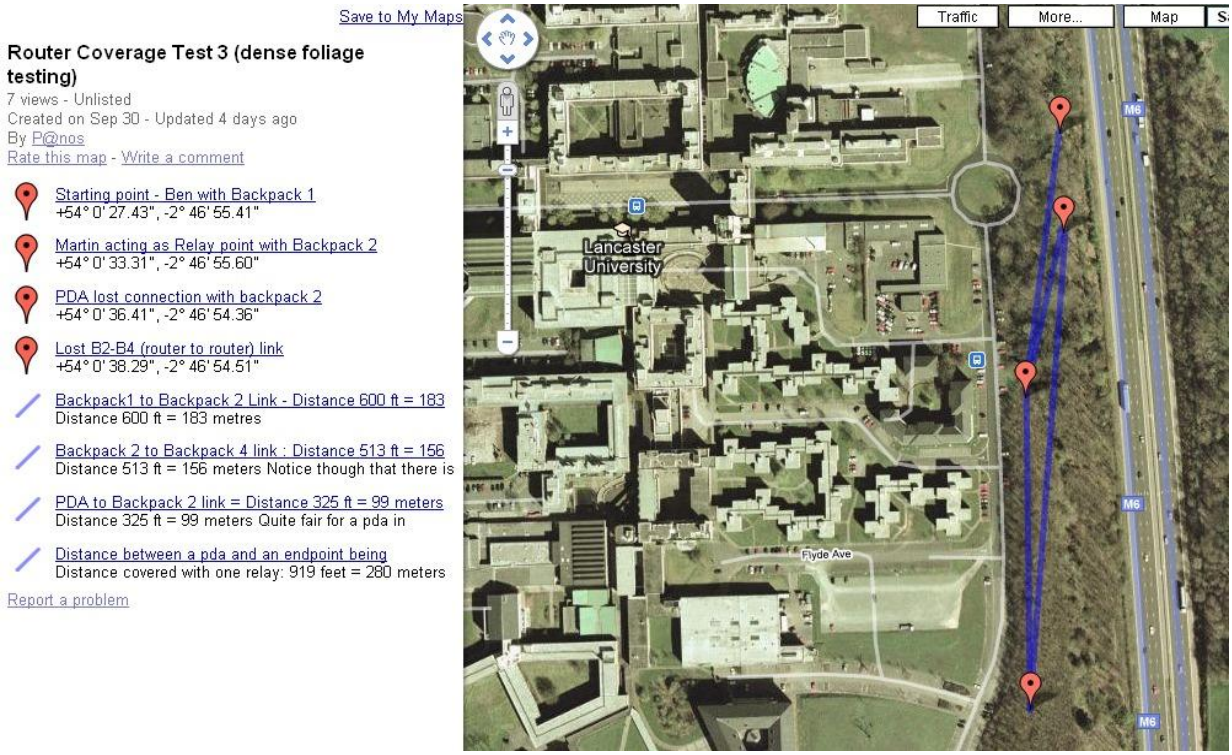

Figure 14 : Range Tests between Backpack Routers in Woodland

4.4. Feedback From Rescuers

To determine the suitability of the physical attributes of the Backpack routers we solicited the opinion of mountain rescue workers, since they are obviously best qualified to comment on these requirements. Of particular interest to us were their opinions on the size and weight of the device with a particular emphasis on their willingness to carry these devices. A unanimously positive aspect was the weight of the device, all rescue workers that responded remarked that the weight of the device was insignificant, relative to the weight they normally carry. Of greater concern was the size and physical dimensions of the device. In its current form the Backpack router is housed in a weatherproof thermoplastic container that is 190mm long by 140mm wide by 70mm deep. To better suit the requirements of the mountain rescue workers (i.e. fit in their backpacks more easily) the footprint of the Backpack router should ideally be reduced. As the footprint of the main board in the current prototype is almost 140mm wide, the width of the housing would not be changeable, however both the length and the depth of the housing could be greatly reduced through the use of a better designed battery and cabling set up within the housing. The use of a flat Li-ion battery with a similar footprint to the main board, layered on top of it would allow us

to drastically reduce the length of the housing (this extra space houses the rectangular battery currently used). Layering a new battery in this manner would then add to the depth of the device, but the current unused space in this dimension is significant and the space could be used even more efficiently with a better cabling solution.

In addition to the dimensions of the Backpack router housing itself, a number of the rescue workers also noted the prominence of the wireless antennas as they are currently connected to the housing. At present the antennas just attach directly to the box and protrude at right angles from it. In the opinion of the rescue workers these would be quickly broken off in a prolonged rescue mission and would therefore be unsuitable. To solve this problem we intend to attach the antennas to the outside of the rescue workers' rucksacks and then connect them to the Backpack router via cable extensions, this approach will potentially give the added benefit of improving the effective wireless range of the Backpack routers since the antennas will no longer be contained within the Backpack itself.

Figure 15 : Backpack Router and Backpack

5. Results of the Satellite and Backhaul Links Tests

When an IAN has been established and localised network connectivity has been deployed using the backpack routers, a further Wide Area Network (WAN) connection is required to ensure data can be transmitted to and from the Internet. In the mountain rescue scenario access to these types of WAN connections, that can provide Internet connectivity, can be limited. It is obvious that local area connectivity options such as public Wi-Fi hotspots will not be available in these scenarios but this is also true of more widely available networks too. Cellular networks, which are now very prevalent in most urban areas and offer ever improving connections to the Internet, are often entirely unavailable across large parts of the mountainous areas that rescue teams operate in. For this reason more specialist solutions must be explored for connecting mountain rescue IANs to the Internet. In this section we document our testing and analysis of two potentially feasible approaches, satellite network connectivity and a custom long range wireless approach for providing backhaul links in the UK Lake District mountainous area.

5.1. Satellite Network testing

When communication is required in a remote environment then one of the most widely available options is satellite technology. Satellites cover extremely large geographical footprints and recent advances have seen satellite network operators installing equipment to support IP communication over their networks. Mountain rescue is an example of one of the areas that can potentially benefit from this technology and therefore we carried out testing of a satellite service that was provided to us by one of the U-2010 partners (SES Astra). For this phase of testing we have conducted numerous experiments both at the campus of Lancaster University and in the Buttermere area of the CMRT search region, in dry and in rainy conditions. Specifically we focussed on the data performance of the satellite link when transmitting IP traffic. The satellite service provided by SES Astra is only IPv4 enabled at present, so also in order to support our IPv6 solution we tested the links performance whilst using an IPv6-in-IPv4 tunnelling technique.

Our satellite on-mountain tests were undertaken at Rannerdale Car Park, on the East shore of Crummock Water (Figure 16 gives an impression of the terrain operations are performed in) and consisted of using an 80cm SES Astra satellite dish to create an uplink to the global Internet. The satellite connection used the Astra2Connect satellite service and connected to the Astra 1E satellite at 23.5°E. We initially had some problems in establishing the connection with the satellite due to high winds and the morphology of the ground making the synchronisation with the satellite more difficult and time consuming. High-winds can often disrupt the synchronisation with the satellite as the dish has a tendency to move regardless of how securely it is fixed to its mounting pole. However, one interesting observation we noted was that even though initial synchronisation can be problematic, once the satellite receiver is synchronised, the connection remains stable even in the presence of those high winds. Our testing also demonstrated that the satellite dish we used is waterproof, however, this cannot be stated for the receiver that comes with it and is mandatory for its use. Therefore, some extra consideration must given as how best to weatherproof this specific piece of equipment. During our tests, we powered the receiver of the satellite dish with a portable generator. This was an acceptable solution for undertaking our tests, however carrying and setting up a generator during a real rescue mission might be considered not feasible, and therefore power is another important concern.

Figure 16 : Astra2Connect Link with Grasmoor in Background

In our tests we found that satellite connection could be established from scratch in an average time of between 5 and 10 minutes, depending on the actual location that the IAN is being setup. This variation depends mainly on the satellite footprint of the area and also whether there are any obstacles (such as high trees or protruding rocks) blocking the line of sight of the dish to the satellite.

Results from our tests in regard to the connection provided from the satellite dish are very promising. Although the Lake District is on the edge of the satellite coverage footprint, we achieved an average downlink rate of around 990Kbps and an average uplink rate of around 244 Kbps in our tests. Round-trip times between the mountain location and Lancaster University (thus traversing Luxembourg and GEANT) averaged at around 600ms. When using IPv6, an IPv6-in-IPv4 tunnel was established using the Hurricane Electric tunnel broker service, and round-trip times increased to around 1000ms.

Figure 17 : Astra2Connect Link tests at Rannerdale Car Park

5.2. Backhaul Link Provided by the CLEO Network

To provide an alternative approach than using satellite connectivity to connect the IAN to the Internet we also experimented with the possibility of utilising the existing infrastructure of the Cumbria and Lancashire Education Online (CLEO) IP network that Lancaster University provides and maintains in the areas surrounding where the Lake District mountain rescue teams operate. As this infrastructure is already in place, the process of extending its reach to support the mountain rescue teams required introducing long range wireless links that could temporarily project network connectivity across the areas the teams operate in. For the purpose of this project the coverage would have to be temporary because stringent restrictions in the area prevented us from considering permanent extensions to the existing infrastructure. However, in essence this requirement ensured that the solution we experimented with was more applicable to other mountainous regions around the world since it did not impose a reliance on a high performance network already being in placed, but rather explored the possibility of extending any available connection into the remote environment of the mountain rescue IAN.

Although Cockermouth (the location the mountain rescue team are based) itself is part of the backhaul CLEO network, the actual search and rescue region of the CMRT has little CLEO connectivity. D4.2.2 describes a full analysis of our actions to set up permanent links to extend CLEO in the region that CMRT operates in and explains why we had to resort in deploying “rapid response” Points of Presence (PoP) in the search area. The idea behind these PoPs was that either a rescue vehicle or a couple of rescue workers could quickly establish a temporary PoP by setting up wireless point-to-point or point-to-multipoint links with known permanent PoPs elsewhere. For this reason, the equipment that comprises the rapid-response PoP needed to be as compact and lightweight as possible so that it could be easily carried by one or two rescue workers.

During our preliminary PMS client tests in the region we have identified technically feasibly and strategic locations to establish our temporary “rapid response” PoP, and we carried out tests in the area to evaluate

our decisions. Figure 18 illustrates the locations to set up PoP which could relay data and provide backhaul connectivity to the IAN. By setting up PoP at the designated locations, we could provide backhaul connectivity to a big part of the search and rescue region which, of course, could be extended with wireless Access Points and Mobile Routers.

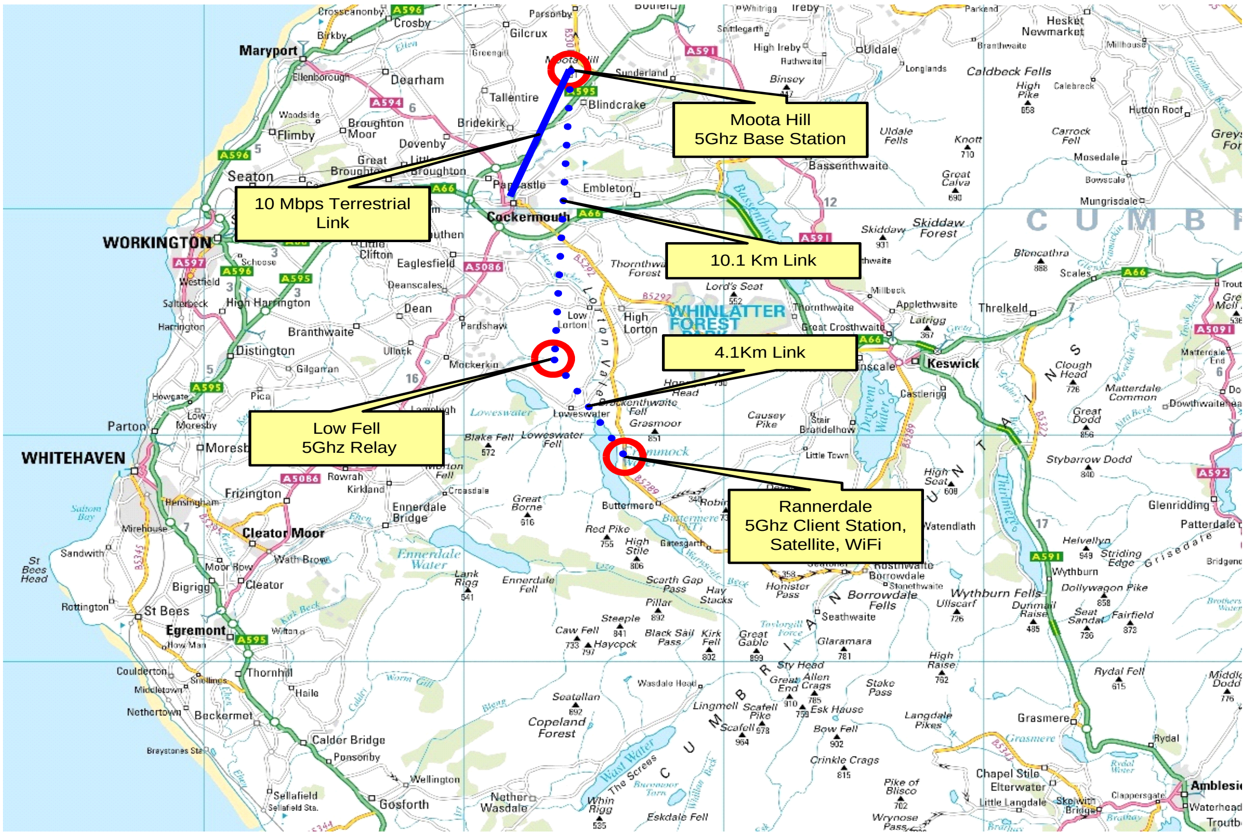

Figure 18 : Testing Locations within the CMRT Search Region

For the CMRT search and rescue missions, the rapid-response PoPs would quickly establish a 5 GHz point-to-point link with the Moota Hill mast, just north of Cockermouth. We identified several key locations where good line-of-sight to Moota Hill offers excellent relaying possibilities to search bases. Perhaps the best location for this is Low Fell to the north of Crummock Water, as Low Fell offers good LoS to Moota Hill in addition to key car park locations and popular tourist locations from which relay points can be placed (Figure 18).

Testing this backhaul link setup in the area has not been easy mainly due to bad weather in the region throughout the year. In addition, we had to engage with communications experts in the field both from academia and from the Information System Services of Lancaster University and ask their help in choosing the right wireless equipment. Balancing certain criteria of the wireless antennas such as their propagation and the distance that they can cover, in addition to their size and weight, needed careful consideration. Furthermore, a serious concern was how to power all this equipment at these rural locations since portable generators are too heavy to be carried to a reasonable distance both for our testing and for the actual rescuers.

Our backhaul link tests in the region were done by again using the car park at Rannerdale Farm as the base for a potential search and rescue operation, which very frequently happens for CMRT missions. At this location, we constructed the IAN by establishing a Wi-Fi hotspot that was directed towards the search area into which the rescue workers are moving, by using a 18dBi directed antenna for the mountain 'hotspot' (see Figure 19). This antenna has a vertical radiation pattern of 45° and a horizontal radiation pattern of 75°. From this location, this allows us to cover the entire West side of Grasmoor (a popular mountain) with one antenna. Our backhaul CLEO link consisted of a 5 GHz microwave relay that began at the Rannerdale car park and was relaying traffic to the Low Fell PoP (Figure 20) and from there to the Moota Hill mast which had a direct link to CLEO. The distance of the Rannerdale car park to Low Fell leg of the link is 4.1Km and the Low Fell to Moota Hill leg of the link is 10.1Km.

Figure 19 : IAN provided for local connectivity relaying data to Low Fell PoP

Figure 20 : Low Fell PoP relaying data from Rannerdale to Moota Hill

Results from our tests with our PoP and wireless equipment were very promising given the fact that the region is extremely rural and that we had not set up any permanent equipment which is forbidden by law as the region is designated as a National Park. Therefore, it is very important to emphasize the fact that we were successful in providing IPv6 backhaul connectivity to the global Internet via CLEO to devices roaming around the rescue region. Our PMS client devices were able to transmit their GPS location updates over this multi-hop wireless network back to the CaC software without any perceptible delay. The bandwidth and delay over this link were more than sufficient for the requirements of the PMS. As we wanted to stress out the networking capabilities of our link we connected two laptops and two cameras as well on the IPv6 provided IPv6 Wi-Fi hotspot transmitting data to the Internet and our server backend. The quality of the video feed being received at the CaC was very good and there was no significant degradation experienced.

Results from these tests also show that the purchased equipment was very appropriate for the kind of setup we were looking for. The antennas and the tripods were detachable and could fit in a typical rescue worker's backpack. In addition we managed to keep the total weight below 5Kg. It should also be noted that for a two person trained team it needed approximately five minutes, after arrival at the location, to set up the PoP and align it with the next relay point. The propagation width of the chosen antennas was deliberately 60° wide so that the rescuers would not loose a lot of time to align them and get the link up very quickly and easily. This combination of relatively low backpack weight and quick link establishment allows the rapid-response PoP to become a realistic solution.

5.3. Feedback From Rescuers

Off all of the components we have developed and tested for use in our mountain rescue solution, the method of providing wide area connectivity to the Internet remains the most problematic. In general the

rescue workers were sceptical about both of the connectivity options we experimented with for a number of recurring reasons. They were unsure about the feasibility of any component that required very specific attention and that would ultimately detract from the time they could be out in the field actually performing their search and rescue missions. The time it took to setup the equipment associated with both of the connectivity technologies we tested was not deemed to be completely infeasible in a single instance, however the inherent mobility of the team (including their rescue vehicles) during a mission would see the team members having to unpack, assemble and align the equipment and then re-pack and transport it elsewhere repeatedly throughout many missions. This apparent repetition would quickly anger the rescue workers and most probably incite resentment towards the equipment and by association, the overall communications solution.

In addition to the time involved, equipment that was considered to be bulky, over-technical and the requirement for some form of specialised setup procedure was also viewed negatively. Space inside the rescue vehicles is very restricted as the rescue workers pack as much rescue equipment into the available area as possible. The suggestion to introduce small devices such as PDAs, video cameras and backpack routers is not expecting too much, however the requirement to carry a large and cumbersome satellite dish is certainly a much bigger "ask". In the case of the backhaul connectivity provided by a long range link into the CLEO network, the network equipment size was considered to be more favourable but the necessity to deploy a dedicated relay in a predefined location was treated with scepticism. Specifically, if a victim was suspected to be in an entirely different region to where the relay must be setup, the rescue team must use up some of their resources (rescue team members) by having them travel to an area entirely unrelated to the mission area in order to setup, wait with and ultimately pack up communications equipment (to provide IANs).

Finally, the overall suitability of the equipment for use in the extreme weather conditions in which the mountain rescue team must operate in was also called into question. High winds are not an infrequent occurrence in these areas and both of the solutions we experimented with were ultimately susceptible to disruption from connection misalignment caused by large wind gusts. As well as the possibility of lost or sporadic service outages due to wind, the overall robustness of this equipment is also questionable. Whilst we were able to operate the satellite dish in rain, the ability of the other satellite components or the long range wireless equipment to withstand constant exposure to wind, rain and snow is unlikely.

6. Results of the Command and Control Software Tests

The command and control (CaC) software is located at the Team HQ and provides a central point for various services including the Alarm Service, the PMS, Video and Picture Service and GIS. The main objective of our CaC tests was to evaluate thoroughly the presence management service and verify that the mission coordinator looking at the software could see the rescuers' information being drawn on interactive Ordinance Survey maps of the search and rescue region and enable him to efficiently coordinate the rescuers and allocate resources according to the needs of the mission. Our tests also scoped to evaluate how well the Alarm service was provided within the CaC software, by alerting rescuers for a mission, collating their responses and informing them about on-going changes during a mission. In addition, CaC tests evaluated the Video and Picture service, the recording and latter replay of a mission, as well as how user-friendly was the GUI and how easy it is for a non-technical person to operate it.

6.1. AlarmTILT Integration

An Alarm Service for the Mountain Rescue scenario has been implemented using M-Plify's AlarmTilt service, described in D4.1.1 [7] and D3.2.1 [5]. Using version 4 of the AlarmTilt SOAP API we have implemented an Alarm client frontend in CaC that alerts rescue workers for emergency calls and replaces (or is complimentary to) the current paging system. Messages can be sent via email, SMS, voicemail or a bespoke client-server messaging system. This functionality was implemented mainly during the summer of 2008 and has been seamlessly integrated into the server application of the PMS (CaC) and is being used ever since.

The alerting functionality appears at the CaC interface in a separate tab-page (see Figure 21), from where the mission coordinator can launch an emergency, contact the selected rescue workers and monitor their responses. The rescuers that respond to the emergency are automatically displayed on the maps once their location updates begin to be received.

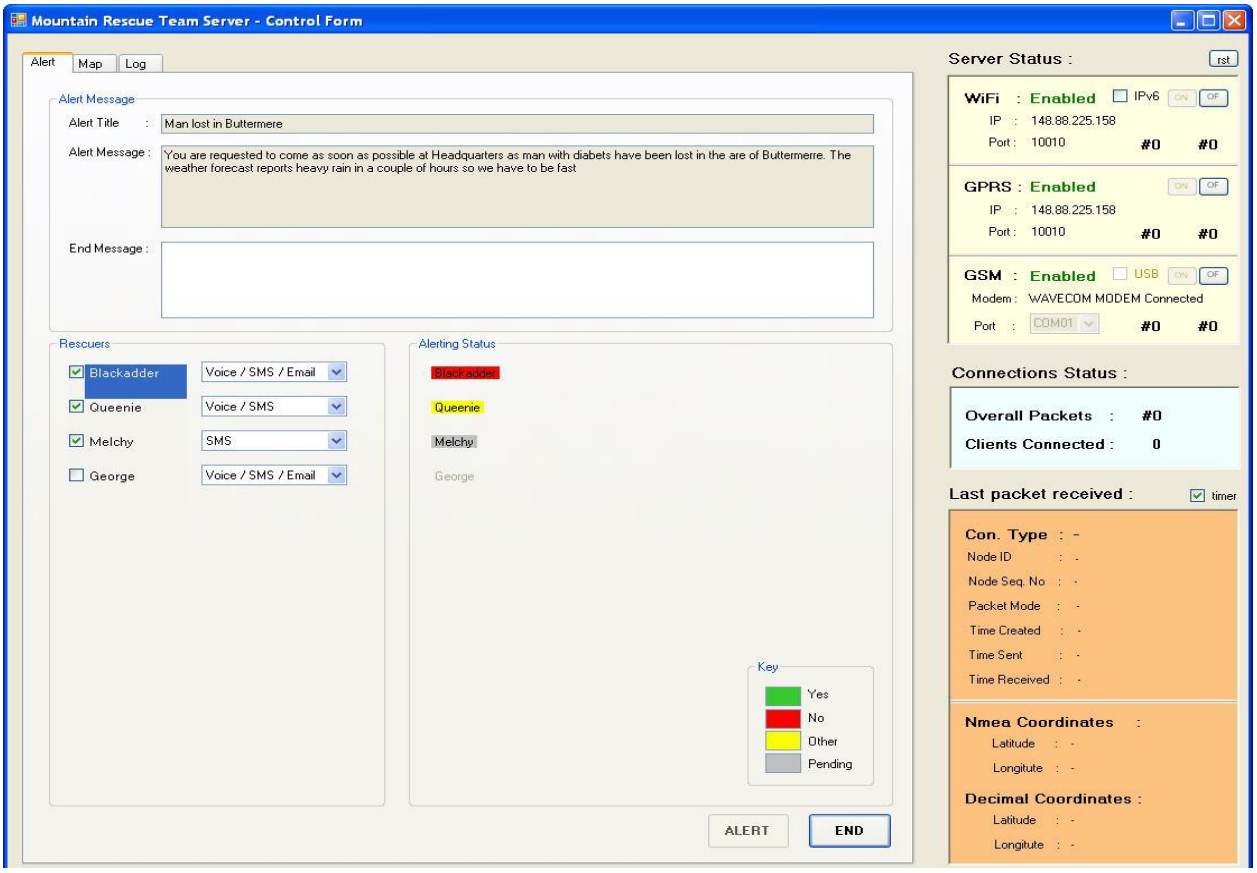

Figure 21 : Screenshot of the Mountain Rescue Alarm Service using AlarmTilt

Various tests were done to evaluate the integration of AlarmTilt with the CaC both in the lab and during our on field trials described in Section 3. M-Plify was developing the SOAP API at the time when we were implementing the integration and therefore our preliminary tests were of great importance for both M-Plify and ourselves. When the implementation of the integration of AlarmTilt within CaC was in a stable version, we started doing thorough tests and providing debugging feedback to Mp-Plify which helped them improve their SOAP API very fast and eventually providing a very reliable and excellent service over it. After this period, almost all of our field and on-mountain tests were starting by simulating an emergency call out using the AlarmTilt from within CaC, which very successfully was launching emergency operations by informing all the available members for an incident. The CaC interface was monitoring the responses of the rescuers and colour coding them based on the XML reply that the interface was getting by polling M-Plify's servers. Results from our tests showed that the rescuers could change their initial reply if they needed to and this was very nicely represented within the CaC interface. The CaC was also able to inform all or a subset of the rescuers for additional information based on the progress of a mission and also notified the rescuers upon closure via a combination of Voice call, SMS and email.

Results from our initial tests showed that it was more convenient to populate the list of rescuers with only members of the team that were available to help (e.g. have not declared being on holidays) and thus we implemented this feature early on.

Generally this integration reached a stable state around October 2008 and very minor changes had to be done to resolve small bugs from that time on. This integration was also demonstrated during an on campus demo at the GA at Lancaster University on the 9th October 2008 (Figure 22)

Figure 22 : "Rescuers" are being informed using AlarmTilt within CaC during Lancaster's GA demo

6.2. Geographic Information System

The GIS part of the CaC interface was a very critical part of the implementation as it was of high importance for the mission coordinator to be able to monitor the rescuers during missions and thus, although we were aware that we were developing a prototype, we wanted that to work very reliably.

There were many different implementation of the GIS part of the system, each of these had a different mapping implementation with their own advantages and disadvantages. At the very beginning of the projected we started by simply using Google Maps which we realized very early that they could not provide the geographical details that the rescuers wanted. Rescuers had strongly requested for a very professional detailed map of the region as it was very important for them to have every morphological detail of the region that they could get. An intermediate implementation, following Google maps, was also done by reading GIS data from xml files, which however was very slow and extremely resource intensive. After many informal meetings with the rescuers and a lot of trial and error implementations, we reached the optimum setup that we could get in the timeline of the project. This setup of the CaC included two different forms, each being shown on its own monitor, one of them being the Control Form and the other being the Map Form (Figure 23). Our Map Form was using a very carefully designed mapping engine that displayed very detailed Ordinance Survey maps of the region that the CMRT operates in. We also felt appropriate to leave the Google maps implementation as a tab page in the Control form so that the mission coordinator could take a "feels-like" view of the terrain that a mission was being undertaken.

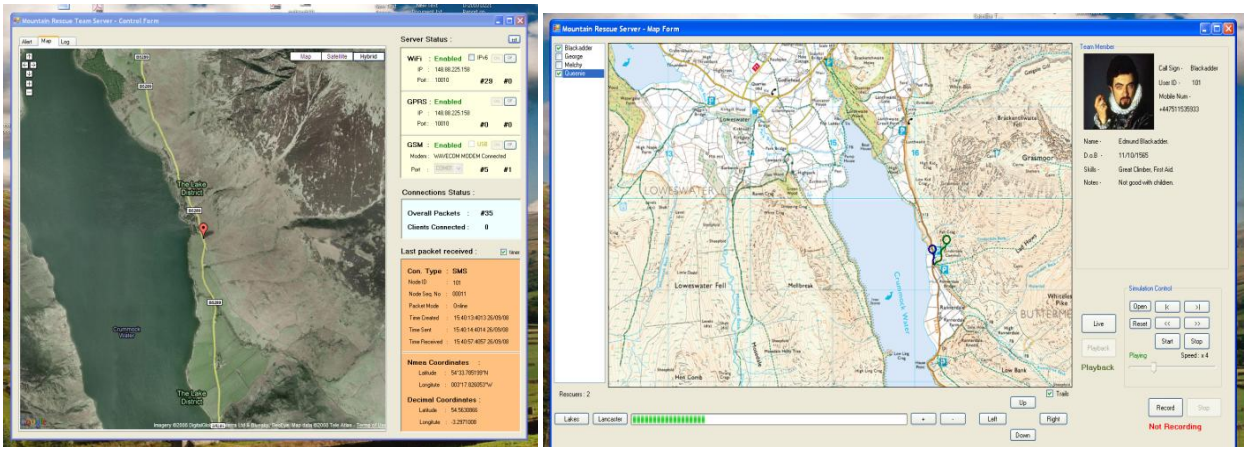

Figure 23 : Control Form (left) and OS Map Form (right) of CaC

The maps implementation was using high resolution and very detailed Ordinance Survey map tiles, which our software was "tailoring" dynamically when being initialized and displaying them as an one piece seamless interactive map. The mission coordinator was able to navigate on the map by either "dragging" it or using the buttons "up", "down", "left", "right" (right part of Figure 23). Furthermore, he could zoom in and out of the map using either the "+" and "-" buttons or the scroll wheel of the pc's mouse. Results from our initial tests of the map though, showed that the application was resource intensive and during several hour of testing even our powerful Windows XP box was running out of resources and CaC was becoming unresponsive.

To resolve this problem and make CaC more robust, we created two different resolution tiles of the OS maps used, namely low and high resolution map tiles. The purpose of this was to load the low resolution map tiles when the user was seeing the map "from a distance" and then swap to the high resolution map tiles when the user was zooming in more than a specified zoom level. This enabled the dramatic drop on resources and made the implementation more lightweight. An additional action that was taken to make the implementation less resource intensive was to load dynamically only the map tiles that were required according to the navigation undertaken by the mission coordinator, than loading all of them when CaC was executed. Therefore, certain tiles where loaded when CaC was initialized and then, when the mission coordinator was moving the map the appropriate tiles were loaded in addition to preloading their nearby tiles (which had the highest probability to be needed next) so that they would be available quickly. This proved to be the implementation that was meeting most of the rescuers' requirements and was stable enough for our further on-mountain tests.

During our on-mountain tests, described in Section 3, the CaC interface was receiving GPS location updates from the PMS client application run on PDAs of users roaming around the search and rescue region. The Control Form of the CaC was responsible to correctly receive the GPS coordinates, authenticate clients, check for the payload's integrity, split merged packets, reorder them if they arrived out of sequence and then pass the appropriate data to the GIS mapping engine of the Map Form. The Map Form was able to draw the "snail-trails" of each rescuer with a different colour and also present additional information about him, when he was "chosen" (clicked) on the map. During these trials we also identified that we had to be able to remove details drawn on the map if needed, in a way that the mission coordinator could be able to dynamically "add" or "remove" information about specific rescuers when required, a feature that was also implemented.

Results from our tests, identified many bugs on the GIS functionality such as the correct identification and display of offline GPS coordinates. Furthermore, we identified additional details that the CaC had to

log for a mission when we were replaying a mission and started studying it (see following Section). In addition, we run into many other bugs during tests with CaC, such as SMS messages being stuck in the GSM modem (and not being forwarded to the application), thread competing for resources and leading to "out of memory" exceptions, counters not being properly updated and others, all of which were resolved and lead to a stable release of the software.

GIS functionality was tested not only for its performance, but also in terms of its accuracy concerning the GPS location updates received and depiction of the appropriate information on the Ordinance Survey maps. Results from each of our tests were analyzed and studied to check whether CaC was drawing correctly the tracks that have been done from each user during our tests. CaC performance on this front was found to be excellent and the tracks being drawn very accurate. To verify this, in one of our tests we setup CaC to run on a laptop that we got with us on the mountain field where we were able to physically see both the users roaming on the field and what exactly CaC was showing on its maps.

One of the problems that we experienced on the GIS part of CaC was when irrational GPS coordinates were received from the PDAs. There were two different but complimentary actions that we took to resolve this problem, one was to try and make sure that the PMS client is obtaining as much accurate coordinates as possible, and the other was to enable some filters on the CaC backend to mark and in some cases discard coordinates that were found to be very far away from the usual region that the CMRT operates in. Obviously, if the software is to be used by another rescue team or organization, these filters would need to be carefully tweaked for the search and rescue mission of the new team.

6.3. Mission Logging and Reconstruction

One of the initial requirements that the CMRT had from the presence management software in general, was to be able to log the missions and if possible to reconstruct/replay them later for offline study. This was of high importance as it would help them to avoid mistakes and train the rescuers carefully on how to react in certain scenarios that in previous mission something went wrong. As we knew that this feature would help the team leader to train his rescuers better and increase the team's overall efficiency, we realized that we had to implement this feature.

Therefore, extensive mission logging was implemented on the CaC, in addition to the mission logging on the PMS client application. Every little detail concerning the network status, packets received, analysis and processing of data, depicting the information in addition to user's interaction with the interface were being logged. Therefore, not only were we able to analyze and debug our software, but also analyze how the users were using it and improve its GUI friendliness and ease of operation.

In addition to the above logging, we implemented a specific feature that was recording a mission and storing all the details on a file from where you could replay a mission later and study how the rescuers were deployed and moved in the rescue field. This feature was proved to be very helpful for the team and this is why we improved it even more by implementing the replay of a mission at different speeds. With this feature, the team did not have to replay a mission in real-time and "lose" for example 5 hours by monitoring the progress of it, but replay it at e.g. 8x speed and also pausing it or slowing it down when they needed to check carefully some specific actions that the did. Implementing the different speeds that a mission could be replayed was a difficult achievement that required many hours of analysis and debugging. However, the feature was fully implemented and in fact was used to replay a mission both in the final project review, as well as the public exhibition after the Tunnel demonstration at Grouft, Luxembourg.

6.4. Instant messaging

It was our intention during the lifetime of the project to integrate a full instant messaging service for both our PMS client and CaC server interface. However, ultimately we decided not to implement an instant message service into our Mountain Rescue solution because the rescuers found it to be unnecessary, because of the interaction required during operation. Instant messaging was an early design decision to be included into the software but as our understanding of the scenario improved we realized it would be surplus to requirement. Consider the role of a mountain rescue worker, whilst in the field during an operation, the likelihood of them being willing to write an entire message on a device keypad is extremely low. In fact, non intrusive communication methods such as hands-free voice services and streaming video are the only viable options in this difficult scenario.

CaC is able to send SMS messages to all or a subset of rescuers using the AlarmTilt service and is also able to receive SMS that are sent from the rescuers during a mission to the SIM card that is on the GSM modem of CaC.

6.5. Video and Picture service

The Video and Picture service, although not a critical requirement of the Mountain Rescue scenario, was seen as highly desirable. The ability to be able to transmit at real-time video and pictures from the search and rescue region back to the HQs was a very useful feature that could help the mission coordinator to collaborate better with the team and improve the efficiency of a mission.

Therefore, we decided to implement it by introducing Video feed and Pictures in a separate tab-page in the Control Form of CaC. This allowed the mission coordinator to be able to see at the same time localization information on the Map Form and also see Video feed on the Control Form very easily as they run on different monitors. The Video and Picture service could also be provided as individual windows (detached from CaC, Figure 24) which offered the ability to make these video windows very small and move them on the side of the screen, so that the person looking at CaC could concentrate on the operation and also glance at points of what the rescuers see without being distracted constantly with the video.

From the rescuers point of view, video and pictures were taken by the IPv6 Panasonic cameras that we chose to use, by attaching them to the central strap of the backpack routers that the rescuers wear. This made sure that the cameras are looking forward the rescuer's chest level and that a rescuer, if required, could detach a wireless camera from the strap and place it in a strategic location transmitting video and pictures, provided that there is wireless access from the IAN.

The two IPv6 Panasonic network cameras that we chose, namely BL-C101 (wired) and BL-C121 (wireless) were able to meet most of the requirements, were highly configurable, easy to work and relatively inexpensive. They offered a variety of video and picture encoding schemes which were essential as the network provided by IAN can suffer from low bandwidth and high delay at times. A very important consideration that we took into account when implementing the video and picture service was that we did not want the rescuers to be troubled to configure the cameras during a mission. In fact, we are providing them in a "turn them on and forget about them" fashion. Therefore, our implementation provides to the mission coordinator the ability to fully configure compression and encoding schemes remotely from within CaC for all the cameras that the rescuers' "wear", or even set this to "auto" so that the cameras could try and adjust to the networking conditions as much as possible. The mission coordinator has even the ability to turn off completely the video service and "swap" to motion pictures (capturing a photo every e.g. 5 seconds) to avoid consuming bandwidth. Another key point that can be

mentioned is that the cameras can also provide a one way audio, and thus the mission coordinator can, if he wants to, listen to what is happening at the rescue field and what the rescuers are saying, in addition to our VoIP service. The high configuration options that are provided from the aforementioned Panasonic cameras were an important factor for their use.

Figure 24 : CaC being demonstrated at public Exhibition at Groumf, Luxembourg. Video feed windows are being used in detached mode

6.6. Feedback From Rescuers

Following the PMS client application's example CaC does not need any configuration or settings to be set from the mission coordinator as all these are read from different configuration files. The only action that the mission coordinator should do is to execute CaC, enable its listening functionality by clicking on the "on" button, and launch an emergency call out when he is being informed about an incident. When we explained this to the mission coordinator he found it very easy to understand and use.

Positive feedback was also received from the use of AlarmTilt within CaC. After an hour training session the mission coordinator operating CaC was able to use the alerting service and launch missions successfully, end them and also communicate with rescuers over it.

The use of Video and Picture service although very easy to use when the cameras were set to "auto", did need a bit more technical information especially on explaining the different video compression schemes and how to react when receiving "broken" video feed. During our training session we did not try to explain technical details to the mission coordinator but just introduced 2 or 3 simple steps that he could do to improve the quality of the video feed and picture service. Detaching the video feed from CaC was also found to be very easy.

On the GIS side of things, the mission coordinator noted that the system responded nicely on their needs, however the scrolling and zooming functions were a bit slow and "jumpy" at times. Other than that, he thought that the map implementation was very easy to navigate and dynamically add/remove rescuers by ticking the appropriate boxes. Positive feedback was also taken for the recording and reconstruction of a mission and the rescuers especially appreciated the replay at high speed functionality that we implemented.

7. Results of the MANEMO Tests

In this section we present and analyse the results of our experimental evaluation of our UMA (Unified MANEMO Architecture) implementation which we developed on the 2.6.22 version of the Linux kernel. For an explanation of MANEMO and how it is used in u-2010, please refer to deliverable D2.2.2 Report on the u-2010 Mobility Solution [4].

In order to perform the tests, we configured two distinct testbeds. Testbed 1 (illustrated in Figure 25) refers to the local setup we devised where all of the associated entities of the testbed are located in the Computing Department at Lancaster University. This setup consisted of five UMA-enabled laptop PCs (configured to operate as MANET nodes), each consisted of a 2Ghz AMD Athlon Processor, 512MB RAM, an onboard Atheros Chipset 802.11b/g wireless interface and a Cardbus Atheros Chipset 802.11a/b/g wireless interface.

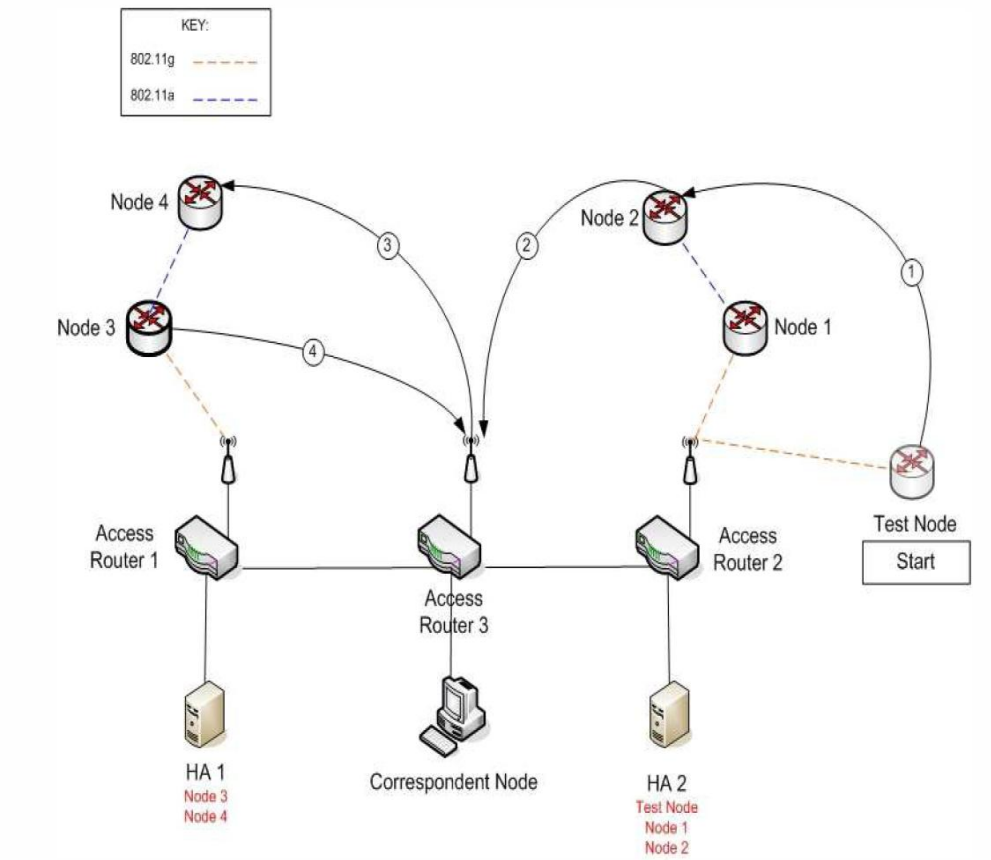

Figure 25 : Local Testbed Setup

This testbed also included two UMA-enabled Linux desktop PCs with 2Ghz CPU, 512MB RAM and 80GB hard drives (configured to operate as HAs), three static IPv6 enabled Cisco routers (labelled Access Router 1 - 3) and three IPv6 enabled Cisco Aironet Wi-Fi Access Points. In all of the experiments, separate interfaces on two of the static routers were used to provide one Home Network and one Access Network per router. In addition, all three of the static routers were also interconnected together using a further interface to provide an Ethernet backbone between all of the networks. Connected to each Home

Network interface via Ethernet was an individual PC configured to operate as a HA (labelled HA1 and HA2 in the diagram). Connected to the Access Network interface of each static router was an IPv6 enabled Aironet Access Point configured to operate in 802.11g mode. Finally, the five UMA-enabled Linux laptop PCs were configured to operate as MANET nodes and therefore form one or more MANETs during testing. 2 (illustrated in Figure 26) on the other hand was designed to illustrate UMA's potential to be deployed for use over the Internet at present, so we therefore incorporated the use of geographically dispersed UMA-enabled HAs (which we located at the University College London's computing department and Lancaster University's main campus network) and Wide Area Internet access technologies (such as a HSDPA link via Vodafone's cellular data network and a satellite communication link via SES Astra's satellite network).

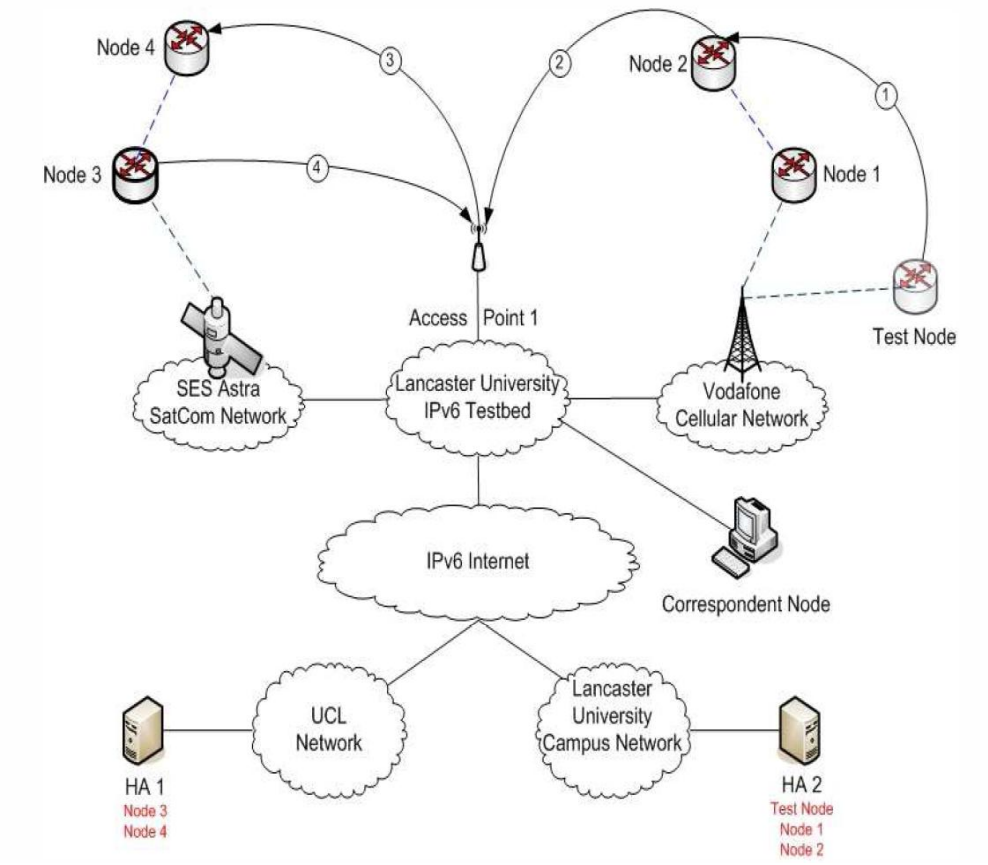

Figure 26 : Global Setup

Over each we performed a 4 stage roaming procedure that tested each of the different potential UMA Binding Update processes that can take place. For each stage, we recorded the handover times experienced, the overall throughput achievable once the handover had taken place and the effect that the UMA approach had on the overall end-to-end latency between a host on the Test Node network and the Correspondent Node. The testing for each stage of UMA mobility we configured was based on the following three step procedure:

1. For each stage we first determined the handover time experienced by using the Ping6 utility in collaboration with the network packet analyser Wireshark. This involved setting the ping request interval to a high value (1 request every 0.01 seconds) and then recording the time difference

between the time the last ping reply was received (i.e. the beginning of the roaming procedure) and the time the next reply was successfully received (i.e. the point at which the connection was re-established).

2. Once the connection was established, the Ping6 utility was then used to collect 1000 Round Trip Time (RTT) measurements to obtain an average latency value.
3. Finally, once the latency test was completed, TCP throughput was determined using the NetPerf bandwidth measurement tool.

For each step in the testing procedure, this regime was repeated 20 times to ensure the results were consistent. We present all of the results from our experimental evaluation over both in their respective sections below and provide a summary of the results for 1 in Table 4 and 2 in Table 5.

7.1. Stage 1: UMA Aggregated Roam

Stage 1 of our testing process was designed to emulate an Aggregated Roaming scenario. In this stage, Node 1 is connected to its respective HA (HA 2) and therefore acts as a Gateway providing an indirect connection to the Internet to Node 2 over its ad-hoc interface. The Test Node then loses its own direct connection to the Internet but is presented with the opportunity to re-establish its connection via Node 2.

Therefore because the Test Node was configured to originate from the same HA as the Gateway (Node 1) in this situation, no HA-HA communication would be required to take place as binding requests from the Test Node would immediately reach its own HA (HA 2) after being tunnelled out of the MANET.

In each of the roaming procedures where the Test Node establishes a connection via an existing MANET, the node is able to perform a 'Make-before-break' handover, whereby it first establishes a layer-2 connection with a MANET node, which it can use as soon as it loses its direct connection to the Internet. In addition, the node is able to register its own ad-hoc interface address as its Care-of-Address (CoA) with its HA as this address is already distributed within the MANET, which means the node is able to avoid the costly process of configuring a topologically correct address as it must do if it establishes a direct connection to the Internet. This therefore results in a relatively quick handover time of under 1 second in 1. In 2 this figure unavoidably increases not only because the round trip time between the Test Node and its HA is much greater, but also because of the lossy nature of the link. In many cases we observed the loss of the initial BU message that the Test Node transmitted, which ultimately causes a longer handover as the node waits to retransmit a second (and in some cases third) BU message. In 1 in this scenario we also saw slight increases in the overall latency experienced and a slight decrease in the throughput measured. This is expected since additional hops via the ad-hoc wireless connections between Node 1, Node 2 and the Test Node are introduced into the end-to-end path. However, in the case of 2 these increases were undetectable because the fluctuation in latency and throughput caused by the HSDPA network link were so large that they effectively masked any performance degradation experienced within the MANET itself.

7.2. Stage 2: NEMO

The movement in Stage 2 represents the Test Node roaming away from the MANET it joined in Stage 1 and establishing its own direct connection to the Internet via Access Router 3. In this situation the Test Node detects that it should act as a Gateway because it becomes involved in the IPv6 Neighbour Discovery process over the interface connected to Access Router 3 and therefore configures a topologically correct address that is valid for use in that network. Again, in this situation it is possible for the MANET node to simultaneously establish an alternative (direct) connection to the Internet at layer 2

whilst it continues to communicate with Internet nodes via its existing connection. This therefore results in the MANET node again being able to quickly perform a handover once it chooses to switch interfaces, as it will already have configured a topologically correct address with which to communicate over the Access Network as well. The resulting configuration that remains in place once the handover in this Stage has been performed offers the best overall latency and throughput performance capabilities because the Test Node is directly connected to the Internet and therefore does not transmit its packets over any additional wireless hops. When carried out over 2, the resulting network configuration from this roaming stage again provided the best performance results as the Test Node was ultimately connected to the highest quality link and registered with the closest HA (HA2).

7.3. Stage 3: UMA Non-Aggregated Roam

In addition to testing the Aggregated Roaming Scenario, it is then important to understand the implications that the additional overhead imposed by the Non-Aggregated Roaming Scenario has on the performance of UMA. Therefore, in this stage of the testing we caused the Test Node to perform a similar handover to an existing MANET by roaming it from Access Router 3 to Node 4. This movement subsequently causes the Test Node to initiate a Non-Aggregated Binding Update because Node 3 (the Gateway) is registered with HA 1 whilst the Test Node is registered with HA 2. This situation therefore highlights the performance implications of the proxy bind request and of the HA-HA communication that is associated with it. What we witnessed in this testing stage was a slight but acceptable increase in the overall handover time required in comparison to the Aggregated Roaming Scenario and similarly acceptable degradation in the latency and throughput performance. This overall performance hit could obviously be expected since the binding process in this scenario involves an additional party and an increase in the overall amount of processing that must be performed. In addition, the introduction of the Proxy-HA into the network configuration also impacted on the overall latency and the achievable throughput. Packets in this scenario were transmitted via an additional hop via the Proxy-HA before reaching the Test Node's own HA, but also incurred the processing overhead of a further IPv6-in-IPv6 encapsulation stage between the two HAs. It is important to note that this procedure represents the most complicated roaming event that can occur with UMA, and therefore no other UMA roaming scenario results in an operation with any greater level of processing overhead. Whilst variable factors such as the density of the MANET (and thus the number of ad-hoc wireless hops packets must travel before they are delivered) or the distance between two inter-communicating HAs will affect the overall service received by a node in a MANET using UMA, the amount of processing in the HAs imposed by the protocol never increases. Regardless of the number of nodes in a MANET or the possible configuration of the HAs, for any individual MANET node the UMA protocol will only result in one level of IPv6-in-IPv6 tunnel (no nested tunnelling is performed) and a potential connection with one Proxy-HA.

With this in mind, these preliminary evaluation results are an encouraging display of the overall capabilities of the UMA protocol. In addition to the performance observations we made when this stage was carried out over 1, this roaming stage also involved communication over the satellite link when we performed it over 2. Utilising this link therefore imposed harsh limits on the level of throughput achievable and increased the Latency experienced significantly. We also observed the highest level of loss over this link and this contributed negatively to the average handover time we recorded.

7.4. Stage 4: UMA Gateway Roam

Finally, we wanted to test the implications of roaming the Gateway node when it has a Non-Aggregated connection in place via a Proxy-HA. To achieve this, we used the resulting MANET network configuration that remained in place after the testing performed in Stage 3 and instead of causing the Test

Node to perform a roaming procedure, we roamed the Gateway (Node 3) from Access Router 1 to Access Router 3.

Since all of the packets that are transmitted between the Test Node and its HA in the Non-Aggregated Roaming Scenario are routed based on the appropriate tunnel ID numbers (i.e. both the HA-HA tunnel ID and the Gateway's tunnel ID), the extent of the packet loss experienced by the Test Node is only determined by the loss of availability of the Gateway connection. For this reason, the roaming of a Gateway from one Access Router to another is the same procedure that a NEMO mobile network performs when it changes its point of attachment to the Internet. This is because the Gateway node must first break its connection to an Access Network in order to subsequently re-establish it with another, different Access Network. As with the NEMO BS protocol, this layer 2 handover time imposes significant performance implications on the overall network layer handover time experienced in these scenarios. This stage in our testing highlights that the performance experienced when using the UMA protocol is only ever at worst equal to the performance that is supported by the NEMO BS protocol.

In this stage of the testing, the resulting network configuration that is in place after the Gateway's roaming procedure has completed is exactly the same as in testing Stage 4 (i.e. Node 3 performing the role of Gateway node with Node 4 and the Test Node attached behind it). Therefore, the latency and throughput results from the testing performed over 1 were observed to be very similar. In contrast however, the resulting configuration in this stage when we performed this testing over 2 culminates with the Gateway node (Node 3) accessing the Internet via an 802.11g Access Point connected to the IPv6 at Lancaster University. This therefore resulted in much improved performance to that experienced in Stage 3 where the Gateway was attached to the satellite network of SES Astra. However, in comparison to the results attained over 2 during Stage 2 of our evaluation (when again the Gateway node has a connection to the Internet via an 802.11g Access Point connected to a relatively high speed network) the throughput performance was considerably lower.

In this situation we ascertained through additional analysis that the bottleneck was in fact imposed by the path available to the HA located at UCL, we were able to determine that even a direct transfer between these two sites was constrained to similar levels of throughput as those we recorded with UMA.

Table 4 Test Results of Local

Stage	Handover (seconds)	Latency (milliseconds)	Throughput
Stage 1	0.89	3.79	9.06 Mbps
Stage 2	0.84	3.48	11.27 Mbps
Stage 3	1.28	6.37	8.84 Mbps
Stage 4	1.47	6.41	8.81 Mbps

Table 5 Test Results of Global

Stage	Handover (seconds)	Latency (milliseconds)	Throughput
Stage 1	3.16	461	140.23 Kbps
Stage 2	1.04	11.32	11.04 Mbps
Stage 3	5.48	637	100.9 Kbps
Stage 4	1.89	22.43	1.76 Mbps

8. Results of the Voice Service Tests

The test results for the Voice Service are displayed in the following fashion, as shown in Table 6, in tables with the result set for each test clearly laid out. Each test has a reference number used when referring about that particular test. The voice quality for each part of individual tests is rated on a scale of 1 to 5 with 1 being bad and 5 being perfect. A value of 0 in any box indicates no voice data getting through at all. The tables are divided up into tests that used Linphone for the Voice Service and ones that used the bespoke VoIP service.

Table 6 Voice Service Results Format

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
0.101	Test 101- New Test	~100ms	X (1-5)	X (1-5)	X (1-5)	X (1-5)	X (1-5)	X (1-5)	No Errors Noticed
0.102	Test 102- New Test	~110ms	X (1-5)	X (1-5)	X (1-5)	X (1-5)	X (1-5)	X (1-5)	No Errors Noticed

8.1. Testing Stage One

Stage one consisted of initial testing just using a single Linux box as a wireless IPv6 access point. Due to these tests being very simple and only as a baseline for all subsequent tests, not all tests were run for Test Set 1. The set was restricted to five tests for each application. Even though the initial test set is limited, we are still be able to see a certain amount of useful information from these tests such as bandwidth usage for each application in an ideal scenario.

Table 7 Custom VoIP Application - Stage 1

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
1.01	Basic Connectivity PDA1 to PDA 2	~1ms	5	5	5	5	5	5	Voice Perfect with no errors heard and no apparent lost packets.
1.02	Basic Connectivity PDA2 to PDA 1	~1ms	5	5	5	5	5	5	Voice Perfect with no errors heard and no apparent lost packets.
1.03	Basic Connectivity Both directions	~1ms/~1ms	5	5	5	5	5	5	Voice Perfect with no errors heard and no apparent lost packets.

1.04	Basic Connectivity Going out of range then back in range - 1 device.	~1ms/~1ms	5	5	3	5	5	4	Voice Perfect while in range but lost while out of range. Some loss near edge of Wi-Fi range.
1.05	Basic Connectivity Going out of range then back in range – both devices	~1ms/~1ms	5	5	3	5	5	4	Voice Perfect while in range but lost while out of range. Some loss near edge of Wi-Fi range.

Table 8 Linphone - Stage 1

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
1.01	Basic Connectivity Laptop 1 to Laptop 2	~1ms	5	5	5	5	5	5	Voice Perfect with no errors heard and no apparent lost packets.
1.02	Basic Connectivity Laptop 2 to Laptop 1	~1ms	5	5	5	5	5	5	Voice Perfect with no errors heard and no apparent lost packets.
1.03	Basic Connectivity Both directions	~1ms	5	5	5	5	5	5	Voice Perfect with no errors heard and no apparent lost packets.
1.04	Basic Connectivity Going out of range then back in range - 1 device.	~1ms	5	5	3	0	0	3	Voice Perfect while in range but lost while out of range and no connection once back in range. Some loss near edge of Wi-Fi range.

1.05	Basic Connectivity Going out of range then back in range – both devices	~1ms	5	5	3	0	0	3	Voice Perfect while in range but lost while out of range and no connection once back in range. Some loss near edge of Wi-Fi range.

After the initial set of tests we can see that both programs performed perfectly in tests 1.01, 1.02 and 1.03. The perceived connection on both applications was perfect and no major packet loss was seen from either device. The only packet loss was negligible, in the order of single figure. From these tests we can conclude that both applications are working to at least a basic standard. The graphed results from these can be seen in Figure 27 and Figure 28. From these we can see the data rates that each application was achieving. The custom VoIP application used a constant bit rate Codec and so sat at around 25kbps for the entire test with only the odd spike. Linphone on the other handed seemed to be using a variable bit rate which sat at around 80Kbps while actual talking was present but dropped to around 20Kbps when no talking was present. Overall, Linphone had a higher data usage rate. Upon examination of Linphone's settings it was determined that it was using the Speex 1600Hz Codec. This means that it would be using a minimum of 28Kbps, which is exactly what we saw when not much audio was present. This has proved that both applications are running normally over an IPv6 connection and that they were both working to the expected standard. In Figure 27 we can also see a spike after the PDA has stopped transmitting. On first glance this just looks like more audio is being sent randomly but in fact upon observation it is simply just a short burst of control messages from the application so can be considered normal behaviour.

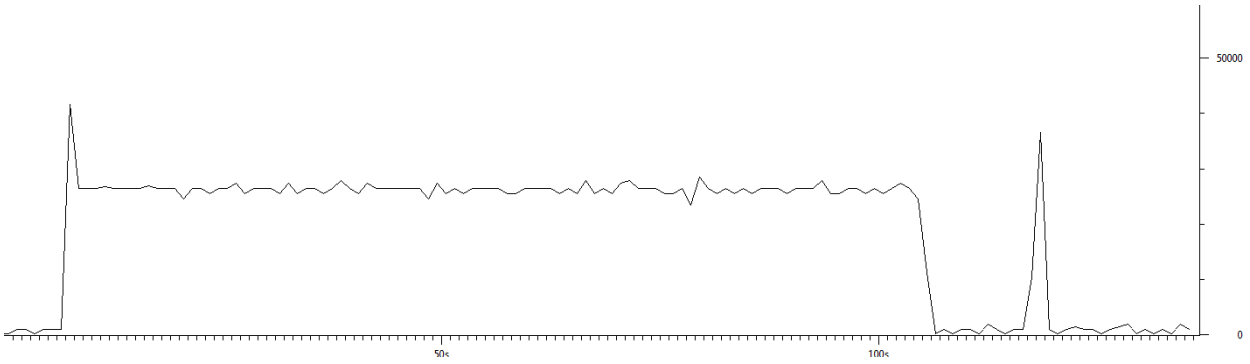

Figure 27 : Custom VoIP application Tests 1.01 - 1.03

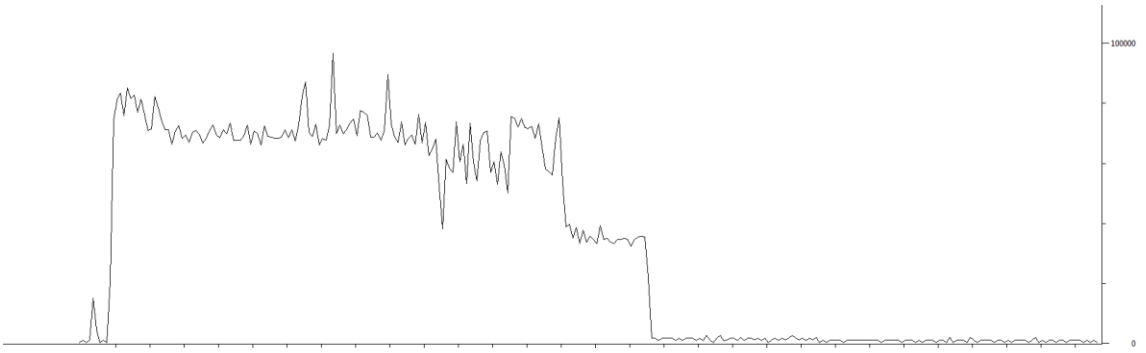

Figure 28 : Linphone 1.01 - 1.03

In tests 1.04 and 1.05, both devices were taken out of range of the wireless network and then back in range. In these tests audio files 1 and 2 were transmitted well within signal range, file 3 was transmitted while walking out of signal range and files 4 and 5 were transmitted while walking back in signal range again. Using the custom VoIP application the connection was maintained during the transmission of file 3, albeit with a few errors which degraded the received quality, as the device moved out of signal range. Once signal had been lost, the connection was dropped. So, only part of the file was heard successfully. Upon returning to within signal range, the application immediately resumed meaning that files 4 and 5 were unaffected. In contrast, Linphone was not able to resume the transmission of files 4 and 5 under identical circumstances. This was entirely expected though as most VoIP software will drop a call if connection is lost. To resume connectivity we would have had to dial a new call.

8.2. Testing Stage Two

Stage two used two access points with a wired Ethernet between them running IPv6. Table 9 and Table 10 show the results for all tests run within stage two of the testing. From this point forward tests where traffic is generated on the network used the IPERF utility to generate the traffic. The traffic was generated from the Linux Access point router and traversed the backhaul medium only.

Table 9 Custom VoIP Application - Stage 2

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
2.01	Basic Connectivity Device 1 to Device 2	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
2.02	Basic Connectivity Device 2 to Device 1	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.

2.03	Basic Connectivity Both Directions Simultaneously	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
2.04	Basic Connectivity Going out of range then back in range – both devices	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
2.05	Mobility Test 1 – Device 1 Moving in Mobility Pattern	~2ms	5	5	2	5	5	4	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared during disconnection time.
2.06	Mobility Test 2 – Device 2 Moving in Mobility Pattern	~2ms	5	5	2	5	5	4	Voice Perfect through all of test and very little packet loss.
2.07	Mobility Test 3 – Both Devices Moving in Mobility Pattern	~2ms	5	4	2	3	4	3	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost and
2.08	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP.	~2ms	3	3	3	3	3	3	Voice very broken but still usable but some packet loss.

2.09	Mobility Test 5 – Both Devices on Edge of Signal Range	~2ms	2	2	2	2	2	2	Voice very broken but still usable but a high amount of packet loss which seemed to come in groups. i.e. Lots of packet loss one second but fine for the next 20 seconds.
2.10	Traffic Test 1- Both Devices in good signal Range with IPERF running at 10kbps UDP over backhaul medium	~2ms	5	5	5	5	5	5	Perfect voice with very little packet loss. No apparent effect from extra traffic.
2.11	Traffic Test 2- Both Devices in good signal Range with IPERF running at 100kbps UDP over backhaul medium	~2ms	5	5	5	5	5	5	Perfect voice with very little packet loss. No apparent effect from extra traffic.
2.12	Traffic Test 3- Both Devices in good signal Range with IPERF running at 1000kbps UDP over backhaul medium	~2ms	4	3	4	4	4	4	Voice a bit broken and a small amount of packet loss but still quite usable. Packet loss in bursts.
2.13	Traffic Test 4- Both Devices in good signal Range with IPERF running at 10000kbps UDP over backhaul medium	~2ms	2	1	2	1	1	1	Some voice traffic getting through on some parts but overall unusable. Large amounts of packet loss in bursts.

2.14	Traffic Test 5- Both Device in good signal Range with IPERF running at maximum output (network card limited) on UDP over backhaul medium	~2ms	0	0	0	0	0	0	With IPERF generating as much traffic on the backhaul network as possible the number of VoIP packets lost was very high and so no actual VoIP traffic was able to get through.
2.15	Traffic Test 6- Both Device on edge of signal range with IPERF running at rate for whichever test provided light interference	~2ms	3	3	3	3	3	3	Voice mostly ok but broken in parts. Packet loss between 10% and 20% on average.

Table 10 Linphone - Stage 2

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
2.01	Basic Connectivity Device 1 to Device 2	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
2.02	Basic Connectivity Device 2 to Device 1	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
2.03	Basic Connectivity Both Directions Simultaneously	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.

2.04	Basic Connectivity Going out of range then back in range – both devices	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
2.05	Mobility Test 1 – Device 1 Moving in Mobility Pattern	~2ms	5	5	2	0	0	3	Voice Perfect through all of tests until disconnect and then no traffic at all.
2.06	Mobility Test 2 – Device 2 Moving in Mobility Pattern	~2ms	5	5	2	0	0	3	Voice Perfect through all of test and very little packet loss.
2.07	Mobility Test 3 – Both Devices Moving in Mobility Pattern	~2ms	5	4	2	0	0	2	Voice Stable through a lot of the test but lost when out of range for rest of the test. When devices were far away voice was intermittent.
2.08	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP.	~2ms	3	3	1	0	0	2	Voice very broken but still usable until disconnect, not afterwards.
2.09	Mobility Test 5 – Both Devices on Edge of Signal Range	~2ms	2	2	1	0	0	1	Voice very broken but still usable until disconnect.
2.10	Traffic Test 1- Both Devices in good signal Range with IPERF running at 10kbps UDP over backhaul medium	~2ms	5	5	5	5	5	5	Perfect voice with very little packet loss. No apparent effect from extra traffic.

2.11	Traffic Test 2- Both Devices in good signal Range with IPERF running at 100kbps UDP over backhaul medium	~2ms	5	5	5	5	5	5	Perfect voice with very little packet loss. No apparent effect from extra traffic.
2.12	Traffic Test 3- Both Devices in good signal Range with IPERF running at 1000kbps UDP over backhaul medium	~2ms	4	3	4	4	4	4	Voice a bit broken and a small amount of packet loss but still quite usable. Packet loss in bursts.
2.13	Traffic Test 4- Both Devices in good signal Range with IPERF running at 10000kbps UDP over backhaul medium	~2ms	2	1	2	1	1	1	Some voice traffic getting through on some parts but overall unusable. Large amounts of packet loss in bursts.
2.14	Traffic Test 5- Both Device in good signal Range with IPERF running at maximum output (network card limited) on UDP over backhaul medium	~2ms	0	0	0	0	0	0	With IPERF generating as much traffic on the backhaul network as possible the number of VoIP packets lost was very high and so no actual VoIP traffic was able to get through.
2.15	Traffic Test 6- Both Device on edge of signal range with IPERF running at rate for whichever test provided light interference	~2ms	3	3	3	3	3	3	Voice mostly ok but broken in parts. Packet loss between 15% and 25% on average.

Both applications performed almost identically on both sets of tests apart from Linphone disconnecting during the mobility tests. Both applications performed better than expected and maintained audio streams over difficult connections.

In terms of mobility, the custom VoIP application was far superior and maintained a better connection throughout most tests. It also successfully re-established its connection upon entering the wireless signal range after leaving it. The custom VoIP application was also able to go slightly closer to edge of the wireless network with less degradation. This is most likely due to it not requiring as much bandwidth as Linphone and therefore being able to cope better when less bandwidth is available during low signal quality.

With the tests involving IPERF both applications were affected by the extra traffic in that some of their packets started to get dropped. It was observed that with moderate traffic Linphone was affected slightly more. Again, this was most probably due to its higher bandwidth requirement than the custom application.

There is an interesting result that in test 2.14 neither application actually managed to get any voice through even though they were both transmitting. This was due to the network being too heavily loaded and most of the voice packets were dropped. This is a scenario where QoS mechanisms should be implemented in the network.

8.3. Testing Stage Three

In stage three of the tests we introduced a large number of hops into the backhaul medium. This was achieved through the use of an IPv6-in-IPv4 tunnel into the Internet to which all packets will be sent and received. Two different tunnels were used during testing to provide varying lengths of delay and hop count. Firstly a relatively close endpoint in London was used followed by a long distance end point located in Hong Kong.

Table 11 Custom VoIP Application - Stage 3 tests

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
3.01	Basic Connectivity Device 1 to Device 2 - London Tunnel	~30ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
3.02	Basic Connectivity Device 2 to Device 1 London Tunnel	~30ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.

3.03	Basic Connectivity Both Directions Simultaneously London Tunnel	~30ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
3.04	Basic Connectivity Going out of range then back in range – both devices London Tunnel	~30ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
3.05	Mobility Test 1 – Device 1 Moving in Mobility Pattern London Tunnel	~30ms	5	5	2	5	5	4	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared during disconnection time.
3.06	Mobility Test 2 – Device 2 Moving in Mobility Pattern London Tunnel	~30ms	5	5	2	5	5	4	Voice Perfect through all of test and very little packet loss.
3.07	Mobility Test 3 – Both Devices Moving in Mobility Pattern London Tunnel	~30ms	5	4	2	3	4	3	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost and
3.08	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP. London Tunnel	~30ms	3	3	3	3	3	3	Voice very broken but still usable but some packet loss.

3.09	Mobility Test 5 – Both Devices on Edge of Signal Range London Tunnel	~30ms	2	2	2	2	2	2	Voice very broken but still usable but a high amount of packet loss which seemed to come in groups
3.10	Basic Connectivity Device 1 to Device 2 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
3.11	Basic Connectivity Device 2 to Device 1 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
3.12	Basic Connectivity Both Directions Simultaneously Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
3.13	Basic Connectivity Going out of range then back in range – both devices Hong Kong Tunnel	~800ms	5	5	4	5	5	4	Voice perfect through all of test with only outage during file 3 while disconnected for a short while. Noticeable delay but not long enough to cause issues with conversation.

3.14	Mobility Test 1 – Device 1 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	5	2	5	5	4	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared during disconnection time.
3.15	Mobility Test 2 – Device 2 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	5	2	5	5	4	Voice Perfect through all of test and very little packet loss.
3.16	Mobility Test 3 – Both Devices Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	4	2	3	4	3	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost and
3.17	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP. Hong Kong Tunnel	~800ms	3	3	3	3	3	3	Voice very broken but still usable but some packet loss.
3.18	Mobility Test 5 – Both Devices on Edge of Signal Range Hong Kong Tunnel	~800ms	2	2	2	2	2	2	Voice very broken but still usable but a high amount of packet loss which seemed to come in groups

Table 12 Linphone - Stage 3 tests

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
3.01	Basic Connectivity Device 1 to Device 2 - London Tunnel	~30ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
3.02	Basic Connectivity Device 2 to Device 1 London Tunnel	~30ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
3.03	Basic Connectivity Both Directions Simultaneously London Tunnel	~30ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
3.04	Basic Connectivity Going out of range then back in range – both devices London Tunnel	~30ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
3.05	Mobility Test 1 – Device 1 Moving in Mobility Pattern London Tunnel	~30ms	5	5	2	0	0	3	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared completely after disconnect.
3.06	Mobility Test 2 – Device 2 Moving in Mobility Pattern London Tunnel	~30ms	5	5	2	0	0	3	Voice Perfect through all of test and very little packet loss. Disappeared completely after disconnect.

3.07	Mobility Test 3 – Both Devices Moving in Mobility Pattern London Tunnel	~30ms	5	4	2	0	0	3	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost. Disappeared completely after disconnect.
3.08	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP. London Tunnel	~30ms	3	3	3	0	0	2	Voice very broken but still usable but some packet loss. Disappeared completely after disconnect.
3.09	Mobility Test 5 – Both Devices on Edge of Signal Range London Tunnel	~30ms	2	2	2	0	0	2	Voice very broken but still usable but a high amount of packet loss which seemed to come in group. Disappeared completely after disconnect.
3.10	Basic Connectivity Device 1 to Device 2 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
3.11	Basic Connectivity Device 2 to Device 1 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.

3.12	Basic Connectivity Both Directions Simultaneously Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
3.13	Basic Connectivity Going out of range then back in range – both devices Hong Kong Tunnel	~800ms	5	5	3	0	0	3	Voice perfect through all of test with only outage during file 3 while disconnected for a short while. Noticeable delay but not long enough to cause issues with conversation.
3.14	Mobility Test 1 – Device 1 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	5	2	0	0	3	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared during disconnection time. Disappeared completely after disconnect.
3.15	Mobility Test 2 – Device 2 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	3	4	2	0	0	2	Voice broken through all of test. Still usable in places though. Disappeared completely after disconnect.

3.16	Mobility Test 3 – Both Devices Moving in Mobility Pattern Hong Kong Tunnel	~800ms	4	4	2	0	0	2	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost. Disappeared completely after disconnect.
3.17	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP. Hong Kong Tunnel	~800ms	3	3	3	0	0	2	Voice very broken but still usable but some packet loss. Disappeared completely after disconnect.
3.18	Mobility Test 5 – Both Devices on Edge of Signal Range Hong Kong Tunnel	~800ms	2	1	1	0	0	1	Voice very broken but still usable for part of the time but a high amount of packet loss Disappeared completely after disconnect.

The results from the tests using the London tunnel were almost identical to those just using the Ethernet backhaul medium. Using the Hong Kong tunnel the tests yield surprisingly similar results even due to the massive difference in RTT. There was a noticeable delay on both applications but neither had an issue at maintaining perfect audio streams. Upon examination of the logs it was seen that the custom VoIP application managed to maintain the same data rate of around 25kbps all through the test but with a few small spikes for traffic. Linphone on the other hand seemed to use much higher data rates than previously. Figure 29 shows a graph for Linphone's data usage over one Hong Kong test. It shows how, for short periods, data rates peaked at twice the average rate. The increased latency seemed to cause the data usage rates to spike in order to maintain a good quality connection. The custom VoIP application did not need demonstrate this behaviour.

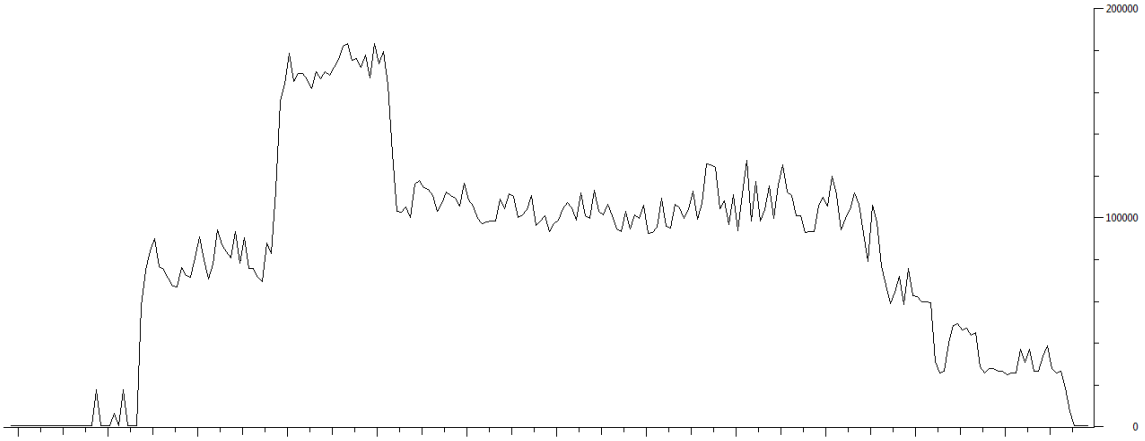

Figure 29 : Linphone conversation over Hong Kong Link

8.4. Testing Stage Four

In stage 4 of the tests we introduced a MANEMO infrastructure consisting of two mobile routers. Linphone was not used for test set 4.1, but was tested in the tunnel environment this was the scenario most likely to produce differences.

Table 13 Custom VoIP application Test Set 4.1

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
4.1.01	Basic Connectivity Device 1 to Device 2	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
4.1.02	Basic Connectivity Device 2 to Device 1	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
4.1.03	Basic Connectivity Both Directions Simultaneously	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
4.1.04	Basic Connectivity Going out of range then back in range – both devices	~2ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.

4.1.05	Mobility Test 1 – Device 1 Moving in Mobility Pattern	~2ms	5	5	2	5	5	4	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared during disconnection time.
4.1.06	Mobility Test 2 – Device 2 Moving in Mobility Pattern	~2ms	5	5	2	5	5	4	Voice Perfect through all of test and very little packet loss.
4.1.07	Mobility Test 3 – Both Devices Moving in Mobility Pattern	~2ms	5	4	2	3	4	3	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost and
4.1.08	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP.	~2ms	3	3	3	3	3	3	Voice very broken but still usable but some packet loss.
4.1.09	Mobility Test 5 – Both Devices on Edge of Signal Range	~2ms	2	2	2	2	2	2	Voice very broken but still usable but a high amount of packet loss which seemed to come in groups. i.e. Lots of packet loss one second but fine for the next 20 seconds.
4.1.10	Traffic Test 1- Both Devices in good signal Range with IPERF running at 10kbps UDP over backhaul medium	~2ms	5	5	5	5	5	5	Perfect voice with very little packet loss. No apparent effect from extra traffic.

4.1.11	Traffic Test 2- Both Devices in good signal Range with IPERF running at 100kbps UDP over backhaul medium	~2ms	5	5	5	5	5	5	Perfect voice with very little packet loss. No apparent effect from extra traffic.
4.1.12	Traffic Test 3- Both Devices in good signal Range with IPERF running at 1000kbps UDP over backhaul medium	~2ms	4	3	4	4	4	4	Voice a bit broken and a small amount of packet loss but still quite usable. Packet loss in bursts.
4.1.13	Traffic Test 4- Both Devices in good signal Range with IPERF running at 10000kbps UDP over backhaul medium	~2ms	2	1	2	1	1	1	Some voice traffic getting through on some parts but overall unusable. Large amounts of packet loss in bursts.
4.1.14	Traffic Test 5- Both Device in good signal Range with IPERF running at maximum output (network card limited) on UDP over backhaul medium	~2ms	0	0	0	0	0	0	With IPERF generating as much traffic on the backhaul network as possible the number of VoIP packets lost was very high and so no actual VoIP traffic was able to get through.
4.1.15	Traffic Test 6- Both Device on edge of signal range with IPERF running at rate for whichever test provided light interference	~2ms	3	3	3	3	3	3	Voice mostly ok but broken in parts. Packet loss between 10% and 20% on average.

4.1.16	Mobile Router Handover Test 1 – MR1 Swaps from Linux AP to access point.	~2ms	5	5	4	5	5	4	All perfect and only a slight delay in middle due to handover but barely noticeable.
4.1.17	Mobile Router Handover Test 2 – MR2 Swaps from Cisco AP to Linux AP	~2ms	5	5	4	5	5	4	All perfect and only a slight delay in middle due to handover but barely noticeable.
4.1.18	Mobile Router Mobility Test 3 – 1 Mobile Router Goes to Edge of signal range and devices stay close to their AP	~2ms	5	5	4	5	5	4	Mostly ok but a few bit of break-up but not as much as expected.
4.1.19	Mobile Router Mobility Test 4 – 1 Mobile Router Goes to Edge of signal range and device attached goes to edge of its MR range.	~2ms	4	3	3	3	3	3	Quite a few break-ups but mostly ok as long as don't get too close to the edge of signal range.

Due to the London tunnel having no significant effect during the tests in stage three, it was decided that the results would be very similar to stage 4.1. Hence, we decided to only use the Hong Tunnel for subsequent tests.

Table 14 Custom VoIP application Test Set 4.2

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
4.2.01	Basic Connectivity Device 1 to Device 2 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Perfect but with delay but not enough to make it unusable.

4.2.02	Basic Connectivity Device 2 to Device 1 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
4.2.03	Basic Connectivity Both Directions Simultaneously Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
4.2.04	Basic Connectivity Going out of range then back in range – both devices Hong Kong Tunnel	~800ms	5	5	4	5	5	4	Voice perfect through all of test with only outage during file 3 while disconnected for a short while. Noticeable delay but not long enough to cause issues with conversation.
4.2.05	Mobility Test 1 – Device 1 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	5	2	5	5	4	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared during disconnection time.
4.2.06	Mobility Test 2 – Device 2 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	5	2	5	5	4	Voice Perfect through all of test and very little packet loss.

4.2.07	Mobility Test 3 – Both Devices Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	4	2	3	4	3	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost and
4.2.08	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP. Hong Kong Tunnel	~800ms	3	3	3	3	3	3	Voice very broken but still usable but some packet loss.
4.2.09	Mobility Test 5 – Both Devices on Edge of Signal Range Hong Kong Tunnel	~800ms	2	2	2	2	2	2	Voice very broken but still usable but a high amount of packet loss which seemed to come in groups
4.2.10	Mobile Router Handover Test 1 – MR1 Swaps from Linux AP to access point. Hong Kong Tunnel	~800ms	5	5	4	5	5	5	Perfect apart from delay in conversation and slight interruption during handover. Delay does not make it unusable.
4.2.11	Mobile Router Handover Test 2 – MR2 Swaps from Cisco AP to Linux AP Hong Kong Tunnel	~800ms	5	5	4	5	5	5	Perfect apart from delay in conversation and slight interruption during handover. Delay does not make it unusable.

4.2.12	Mobile Router Mobility Test 3 – 1 Mobile Router Goes to Edge of signal range and devices stay close to their AP Hong Kong Tunnel	~800ms	4	4	4	4	4	4	Perfect apart from delay in conversation and intermittent interruptions Delay does not make it unusable.
4.2.13	Mobile Router Mobility Test 4 – 1 Mobile Router Goes to Edge of signal range and device attached goes to edge of its MR range. Hong Kong Tunnel	~800ms	3	3	2	3	3	3	Persistent interruptions in audio but still useable but have to repeat some things during conversation.

Table 15 Linphone Test Set 4.2

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
4.2.01	Basic Connectivity Device 1 to Device 2 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Perfect but with delay but not enough to make it unusable.
4.2.02	Basic Connectivity Device 2 to Device 1 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
4.2.03	Basic Connectivity Both Directions Simultaneously Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.

4.2.04	Basic Connectivity Going out of range then back in range – both devices Hong Kong Tunnel	~800ms	5	5	4	5	5	4	Voice perfect through all of test with only outage during file 3 while disconnected for a short while. Noticeable delay but not long enough to cause issues with conversation.
4.2.05	Mobility Test 1 – Device 1 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	5	2	0	0	3	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared during disconnection time. Disappeared completely after disconnect.
4.2.06	Mobility Test 2 – Device 2 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	3	4	2	0	0	2	Voice broken through all of test. Still usable in places though. Disappeared completely after disconnect.

4.2.07	Mobility Test 3 – Both Devices Moving in Mobility Pattern Hong Kong Tunnel	~800ms	4	4	2	0	0	2	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost. Disappeared completely after disconnect.
4.2.08	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP. Hong Kong Tunnel	~800ms	3	3	3	0	0	2	Voice very broken but still usable but some packet loss. Disappeared completely after disconnect.
4.2.09	Mobility Test 5 – Both Devices on Edge of Signal Range Hong Kong Tunnel	~800ms	2	1	1	0	0	1	Voice very broken but still usable for part of the time but a high amount of packet loss Disappeared completely after disconnect.
4.2.10	Mobile Router Handover Test 1 – MR1 Swaps from Linux AP to access point. Hong Kong Tunnel	~800ms	5	5	4	5	5	5	Perfect apart from delay in conversation and slight interruption during handover. Delay does not make it unusable.
4.2.11	Mobile Router Handover Test 2 – MR2 Swaps from Cisco AP to Linux AP Hong Kong Tunnel	~800ms	5	5	4	5	5	5	Perfect apart from delay in conversation and slight interruption during handover. Delay does not make it unusable.

4.2.12	Mobile Router Mobility Test 3 – 1 Mobile Router Goes to Edge of signal range and devices stay close to their AP Hong Kong Tunnel	~800ms	4	4	4	4	4	4	Perfect apart from delay in conversation and intermittent interruptions Delay does not make it unusable.
4.2.13	Mobile Router Mobility Test 4 – 1 Mobile Router Goes to Edge of signal range and device attached goes to edge of its MR range. Hong Kong Tunnel	~800ms	3	3	0	0	0	2	Persistent interruptions in audio until complete loss of connection. Possible timeout and call dropped.

If we analyse these results we can see that the initial set of tests in each of 4.1 and 4.2 for the custom VoIP application are very similar and the only main difference is the delay experienced because of routing through Hong Kong. In this set of tests there was also a more noticeable set of fluctuations in data rates. The bi-directional traffic can be seen in Figure 30. It seems when the application starts up the data rate increases by a small amount for each device, around 10Kbps. This variation though is still not enough to stop the application from working adequately and it is currently outperforming expectations.

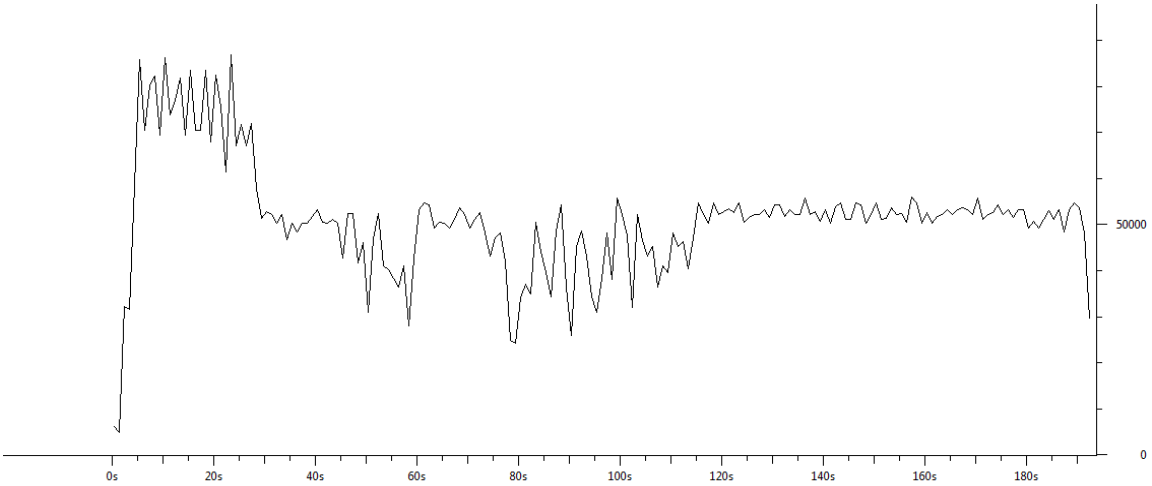

Figure 30 : Hong Kong MANEMO Custom VoIP – Both ways

If we also look at the results from test set 4.1 we can see that during QoS testing the same detrimental effect occurred on the traffic over a MANEMO infrastructure. This is not surprising as MANEMO does

not yet work with any QoS mechanisms. If we look at Figure 31 we can see the traffic that got through the backhaul to the local router and almost none of this was voice data. The large spikes on the graph are the generated UDP traffic that got through and the low red line is the voice data. Unfortunately, this was not a sufficient data rate with which to hold a conversation.

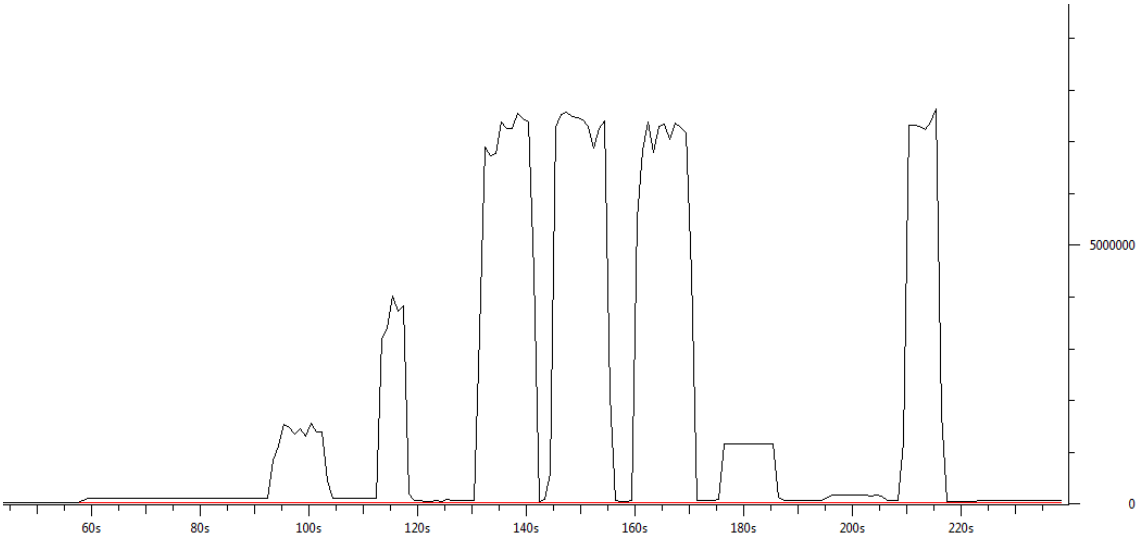

Figure 31 : QoS MANEMO Custom VoIP

Overall, MANEMO did help on performance by helping in the tunnel scenarios to route local traffic locally rather than both ways via the tunnel. This significantly reduced packet latency and made the conversation easier. This effect is extremely beneficial when a device has connections with local peers but the connection to its HA over the global link incurs large latencies. One example of this could be when a rescuer needs to connect to others by satellite but still needs to be connected to other rescuers in the same local network. In this case, the traffic will be routed by optimal means such that local traffic will not go over the satellite connection.

From looking into the mobile router handovers it can be seen that the results with and without the tunnel link are exactly the same. This is a very promising sign as it shows MANEMO is able to handle the voice traffic perfectly and that the voice applications can cope with the movement events that occur at the network layer.

8.5. Testing Stage Five

In test stage 5 a series of tests were run using a chain of mobile routers connected in MANEMO infrastructure. There were a total of 5 wireless hops in the chain.

Table 16 Test Set 5.1 Custom VoIP Application Results Table

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
5.1.01	Basic Connectivity Device 1 to Device 2	~10ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
5.1.02	Basic Connectivity Device 2 to Device 1	~10ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
5.1.03	Basic Connectivity Both Directions Simultaneously	~10ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
5.1.04	Basic Connectivity Going out of range then back in range – both devices	~10ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss.
5.1.05	Mobility Test 1 – Device 1 Moving in Mobility Pattern	~10ms	5	5	2	5	5	4	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared during disconnection time.
5.1.06	Mobility Test 2 – Device 2 Moving in Mobility Pattern	~10ms	5	5	2	5	5	4	Voice Perfect through all of test and very little packet loss.

5.1.07	Mobility Test 3 – Both Devices Moving in Mobility Pattern	~10ms	5	4	2	3	4	3	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost and
5.1.08	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP.	~10ms	3	3	3	3	3	3	Voice very broken but still usable but some packet loss.
5.1.09	Mobility Test 5 – Both Devices on Edge of Signal Range	~10ms	2	2	2	2	2	2	Voice very broken but still usable but a high amount of packet loss which seemed to come in groups. i.e. Lots of packet loss one second but fine for the next 20 seconds.
5.1.10	Traffic Test 1- Both Devices in good signal Range with IPERF running at 10kbps UDP over backhaul medium	~10ms	5	5	5	5	5	5	Perfect voice with very little packet loss. No apparent effect from extra traffic.
5.1.11	Traffic Test 2- Both Devices in good signal Range with IPERF running at 100kbps UDP over backhaul medium	~10ms	5	5	5	5	5	5	Perfect voice with very little packet loss. No apparent effect from extra traffic.

5.1.12	Traffic Test 3- Both Devices in good signal Range with IPERF running at 1000kbps UDP over backhaul medium	~10ms	4	3	4	4	4	4	Voice a bit broken and a small amount of packet loss but still quite usable. Packet loss in bursts.
5.1.13	Traffic Test 4- Both Devices in good signal Range with IPERF running at 10000kbps UDP over backhaul medium	~10ms	2	1	2	1	1	1	Some voice traffic getting through on some parts but overall unusable. Large amounts of packet loss in bursts.
5.1.14	Traffic Test 5- Both Device in good signal Range with IPERF running at maximum output (network card limited) on UDP over backhaul medium	~10ms	0	0	0	0	0	0	With IPERF generating as much traffic on the backhaul network as possible the number of VoIP packets lost was very high and so no actual VoIP traffic was able to get through.
5.1.15	Traffic Test 6- Both Device on edge of signal range with IPERF running at rate for whichever test provided light interference	~10ms	3	3	3	3	3	3	Voice mostly ok but broken in parts. Packet loss between 10% and 20% on average.

5.1.16	Mobile Router Handover Test 1 – Wireless Chain Performs Handover to other AP	~10ms	5	5	4	5	5	5	Voice Remained perfect throughout with only a slight interrupt when the actual handover occurred. This was about a 1 second gap
5.1.17	Mobile Router Handover Test 2 – Distance Increased between mobile routers and the chain performs handover to other AP	~10ms	5	5	4	5	5	5	Voice Remained perfect throughout with only a slight interrupt when the actual handover occurred. This was about a 1 second gap.

Table 17 Test Set 5.2 Custom VoIP Application Results Table

Test Num	Test Description	Approx RTT	File 1	File 2	File 3	File 4	File 5	User Voice	Comments
5.2.01	Basic Connectivity Device 1 to Device 2 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
5.2.02	Basic Connectivity Device 2 to Device 1 Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.

5.2.03	Basic Connectivity Both Directions Simultaneously Hong Kong Tunnel	~800ms	5	5	5	5	5	5	Voice Perfect through all of test and very little packet loss despite the tunnel length.
5.2.04	Basic Connectivity Going out of range then back in range – both devices Hong Kong Tunnel	~800ms	5	5	4	5	5	4	Voice perfect through all of test with only outage during file 3 while disconnected for a short while. Noticeable delay but not long enough to cause issues with conversation.
5.2.05	Mobility Test 1 – Device 1 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	5	2	5	5	4	Voice Perfect through all of tests apart from one where it broke up just on edge of signal range and disappeared during disconnection time.
5.2.06	Mobility Test 2 – Device 2 Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	5	2	5	5	4	Voice Perfect through all of test and very little packet loss.
5.2.07	Mobility Test 3 – Both Devices Moving in Mobility Pattern Hong Kong Tunnel	~800ms	5	4	2	3	4	3	Voice Stable through a lot of the test but with both devices going at varying signal ranges more packets were lost and

5.2.07	Mobility Test 4 – One Device on edge of wireless signal range the other near its AP. Hong Kong Tunnel	~800ms	3	3	3	3	3	3	Voice very broken but still usable but some packet loss.
5.2.09	Mobility Test 5 – Both Devices on Edge of Signal Range Hong Kong Tunnel	~800ms	2	2	2	2	2	2	Voice very broken but still usable but a high amount of packet loss which seemed to come in groups
5.2.10	Mobile Router Handover Test 1 – Wireless Chain Performs Handover to other AP Hong Kong Tunnel	~800ms	5	5	4	5	5	5	Voice Remained perfect throughout with only a slight interrupt when the actual handover occurred. This was about a 1 second gap delayed because of large RTT though.
5.2.11	Mobile Router Handover Test 2 – Distance Increased between mobile routers and the chain performs handover to other AP Hong Kong Tunnel	~800ms	5	5	4	5	5	5	Voice Remained perfect throughout with only a slight interrupt when the actual handover occurred. This was about a 1 second gap delayed because of large RTT though.

From the results we can see that in the MANEMO environment things function in a very similar way on the surface for everything in tests 5.2.02-5.2.09. There are only a few minor differences but these could be due to just random interference.

In general this set of tests shows very similar results to test stage 4 showing that the wireless hops had little difference in packet loss when they were not too far away from each other. The major results that are interesting from this test set are for tests 5.2.10, 5.2.11, 5.1.16 and 5.1.17 where the entire wireless chain handover took place. If we just look at the handover graphed for the bi-directional traffic without the tunnel we can see the effect on the VoIP traffic. This is shown in Figure 32. The start of the graph is when the handover occurs. We can see the data rate drops off substantially but briefly, before spiking for a short period of time and then returning to normal data rates.

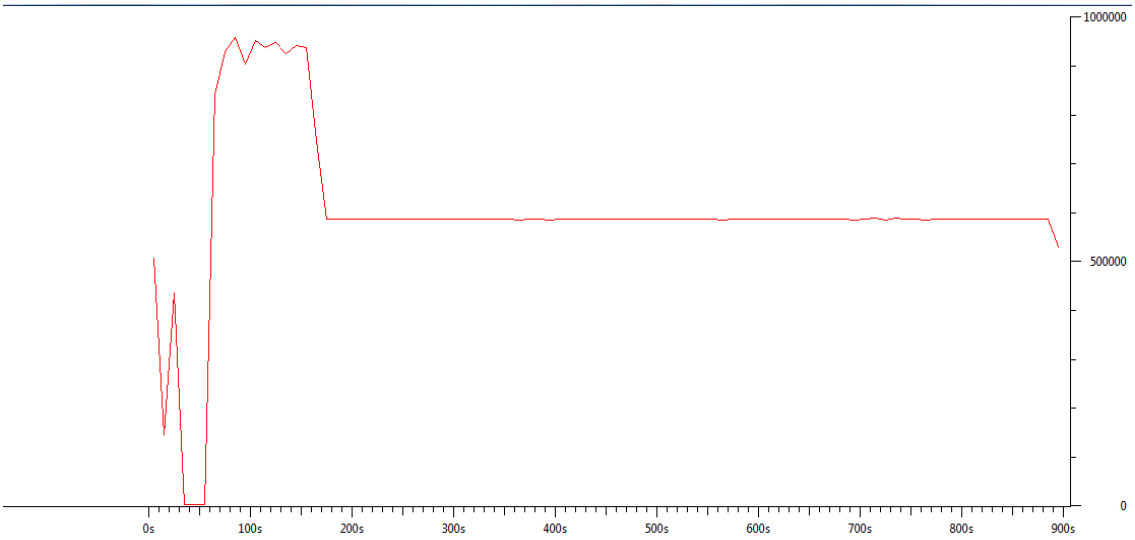

Figure 32 : Custom VoIP application both traffic sets on handover

If we look at the results closely we can see that the tunnel tests have slightly better results when using MANEMO. At first glance it is not apparent why this is so until the logs are examined. The data rate fluctuates around 50Kbps for both devices (25Kbps * 2) but occasionally drops to 25Kbps for period of time. This looks like one device is not transmitting but in actuality, the traffic of that device is routed locally and avoids the tunnel (and also the logs being made at the home agent). This is exactly the effect that is desired from using MANEMO. The other fluctuations are to do with the distance from the access point as described in previous test sets.

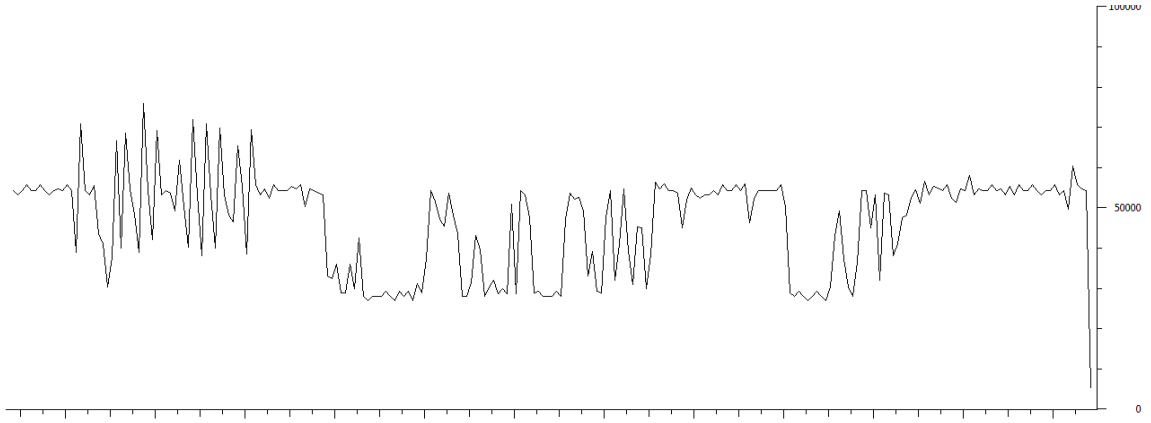

Figure 33 : Custom VoIP application over Hong Kong Tunnel

Overall in this section MANEMO did help on performance by routing appropriate traffic locally rather than both ways via the tunnel. This significantly reduced the end-to-end packet latency

8.6. Overview

Now all the tests have been carried out and analysed it is easy to see how the applications faired against each other in terms of performance, reliability and ability to cope with the test scenarios. Both applications performed extremely well under most tests. The tests exposed interesting facts about VoIP over wireless and informative information about how MANEMO affects streamed UDP traffic. In particular it was interesting to see the effect of the traffic over the tunnel to Hong Kong and despite the enormous number of network hops involved, voice traffic was still mostly intact and packet loss was kept to a minimum.

Overall, we can see that tests where one or both devices were far away from the access points produced the worst results. This shows that wireless networks lose significantly more at long range than when devices are close to the access point. This seems to play a much larger part in packet loss than having multiple wireless hops, which barely affected the VoIP traffic, provided range (and thus signal quality), was kept acceptable.

The MANEMO (specifically, UMA) environment has definitely shown advantages in certain areas where there would normally be a long packet delay for one connection. One set of packets are able to be routed locally, optimising packet latency and making the user experience much more improved than without having MANEMO running.

8.7. Feedback From Rescuers

Feedback from the rescuers after using the applications was taken after each set of tests. This is what was used to give the results for the perceived effect of the VoIP audio streams and the assessment of how useable the software was.

For the custom VoIP application the overall user perception was that it was good, reliable and preferred over the use of Linphone due to the connection stay alive properties. Not actually having to answer the phone when voice is being received was also a benefit. This meant for someone just listening the device

could be used with a headset and placed in a pocket leaving their hands free to do other things. The only negative comments were that if packets arrived out of order they would get mixed into the current audio stream which meant the audio could become slightly garbled at some stages. However, this seldom happened in testing even over the links with extremely long RTT's. This is due to the fact that packets are more likely to be lost than arrive out of order, so it is not a major area of concern.

For Linphone, the users felt it was a useful program with nice audio quality but the lack of features to help it cope with the tests made it a minor hassle to use in some scenarios. It was commented that the audio streams would break up a lot more easily that with the custom VoIP application, which could be due to the use of SIP in it attempting to provide call management features.

9. Results of the Video Service Tests

After support for voice communication and localising services, video is seen as the next most important service that could be provided to the mountain rescue team. The relatively high bandwidth capabilities of our networking approach mean that we can potentially provide this service in this challenging domain where live streaming video has never been a possibility before. It is possible that with further exposure, the mountain rescue team could find video services to be as important, if not more important, than voice services in certain circumstances in the future, once they become accustomed to its use. In this section we detail the results from testing both the basic requirements of video camera hardware (detailed in Section 2.6: Video Service Tests) and the streaming video services we were able to achieve across typical topologies generated by our MANEMO networking solution (detailed in Section 2.9: MANEMO and Video Service).

9.1. Basic Video Service Test Results

Before testing the capabilities of the streaming video service over complex mobile networking topologies we first performed some basic tests to ascertain that our Video Service would suitably support some of the most basic requirements of our scenario.

Firstly, of key importance we ensured that our Video Service could stream using each codec and in all available frame rates over IPv6. To do this we statically connected the cameras to our University IPv6 and accessed them from numerous different locations around the University campus, as well as accessing them from a U-2010 partner's remote location (UCL). In all instances the picture quality transmitted exceeded our expectations of the relatively in-expensive hardware we had selected. Another important factor was the data rates observed when the Video Service was started. For each camera we identified the amount of bandwidth consumed for video streams using different encoding schemes and resolutions and also multiple simultaneous streams. The results for these tests are presented in Table 18and Table 19 below.

Table 18 Video Service Data Rates – MPEG-4 Encoding

MPEG-4	Wired Cam	Wireless Cam
Idle	0bps	0bps
192x144	~1.5Mbps	~1.5Mbps
320x240	~3.1 Mbps	~3.1 Mbps
640x480	~5.5 Mbps	~5.5 Mbps

Table 19 Video Service Data Rates - MJPEG Encoding

MJPEG	Wired Cam	Wireless Cam
Idle	0bps	0bps
192x144	~1.5 Mbps	~1.5 Mbps
320x240	~3.1 Mbps	~3.1 Mbps
640x480	~5.7 Mbps	~5.7 Mbps

Firstly, what is most notable is the similarity between the wired and wireless camera results we observed as well as the similarity between the different streaming modes. As both cameras are very similar products (except the wireless camera has an additional Wi-Fi interface) it is therefore obvious from these results that the wireless camera's functionality has not been modified in anyway to operate differently to that of the wired camera. In fact, without close inspection, when transmitted across a high quality end-to-end path it is difficult to see any difference in the resulting video stream from the wireless or the wired camera. This was most probably one of the design goals of the camera manufacturer.

At each resolution the throughput generated can be seen to be relatively stable with significant differences between each level of quality. Specifically, an increase in the resolution level requested can be seen to increase the bandwidth used by around 100%. The lowest quality stream generates an average of 1.5 Mbps of traffic, whereas the medium quality stream can be seen to generate over 3 Mbps, and finally the high quality stream generates almost 6 Mbps of traffic. This is logically what would be expected since each resolution increase effectively doubles the size of the video images that are transmitted.

When we inspect the traffic generated at the network level, it is possible to notice some differences however, mainly in the steadiness of the output stream that the cameras produce. Figure 34 illustrates the traffic generated by the wired video camera. The resulting graph is very well defined with very obvious alterations in throughput as the resolution requested was changed, as well as smooth continuous transfers at each different rate. However with the wireless webcams we observed a less smooth flow of traffic, as interference and subsequent packet loss the transmission of packets to jump around more sporadically.

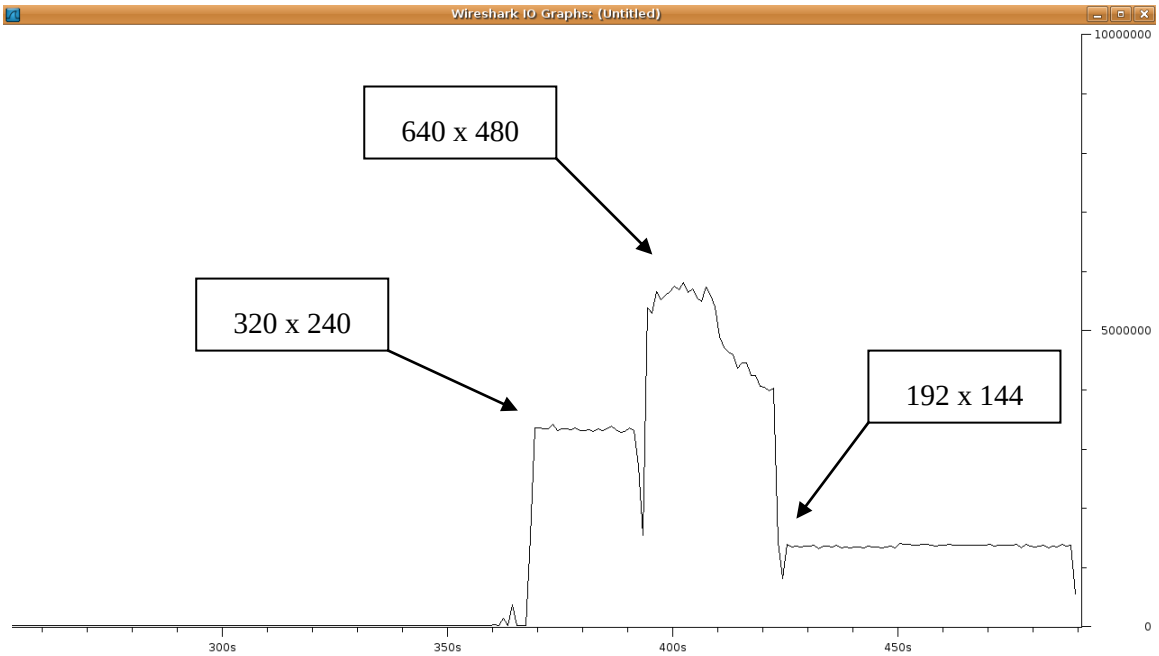

Figure 34 : Video Service Bandwidth Utilisation

In addition to the throughput observed when the video is requested, it is also important to note the significance of the camera transmitting no traffic when it is idle. This is important because if a stream from an individual or a vehicle's camera feed is not required then the camera can remain silent, awaiting a request for it to begin transmitting. This is much more efficient than a broadcast approach where every camera must continually transmit regardless of whether anyone is watching or not. This is possible because of the global reachability afforded the camera by IPv6 and the MANEMO mobility protocol. Finally, the video streams transmitted in this basic setup were observed to be smooth flowing and with

good definition since they were able to transmit at the required bandwidth level continuously throughout use.

9.1.1. Power Consumption of the Video Camera Hardware

The power consumption level of the streaming video cameras is also important in the Mountain Rescue scenario since any remote camera worn by a rescuer will ultimately need to be battery powered independently and therefore the power consumption levels will affect the type of battery solution required. As with the bandwidth consumption tests, both the wired and wireless versions of the cameras used in the Video Service were tested for their power consumption rates when transmitting video at various resolutions.

To carry out these tests we detected each of the cameras' power consumption levels using a clamp meter, whilst simultaneously transmitting video over our MANEMO network topologies. The results from our tests are presented in

Table 20 where we show the power consumption observed for both the wired and wireless versions of the two different cameras we tested, when transmitting at the 3 different resolutions they support. We observed that the power consumption increases only slightly when the video resolution is increased. However a far greater difference in power consumption rates were noticed between the wired and wireless versions of the camera, with the wireless version consuming around twice as much current as the wired version. When not transmitting (idle) the wired camera consumes a rate of 150 milliamps (mA) and the wireless camera consumes around 320 milliamps. The cameras support three different resolution qualities and two different video codecs (MJPEG and MPEG4). Utilising different codecs had no noticeable effect, whilst altering the resolution caused only minor increases in the level of power consumption.

Table 20 Power consumption vs. Transmitted Video Resolution

Video Resolution	Wired Cam	Wireless Cam
Idle	150mA	320mA
192x144	160mA	320mA
320x240	170mA	330mA
640x480	180mA	340mA

For the purpose of an actual deployment, the wireless camera makes a lot of sense, but using a wired camera also has advantages (for instance its transmissions don't contribute to the interference levels experienced by a Backpack router). In a deployment where every individual and vehicle that requires a video camera can be assumed to also have a Backpack router then the wired camera represents the best video approach because its battery will last over twice as long. However, the wireless camera has the distinct advantage that the wearer does not need to also have their own personal Backpack router. Rather, the wireless camera only needs to be within range of any available Backpack router or other wireless network that is connected to the Incident Area Network. In this case then there is a trade off between increased wireless freedom and interference levels and battery life.

9.2. MANEMO and Video Service Tests Results

As with the Voice Service over MANEMO networks testing performed in the previous section, the purpose of these tests was to ensure that streaming video media across typical MANEMO topologies was possible. In this stage of the testing a key observation was the extent of the degradation to any given video stream, rather than just confirmation of whether video service was possible or not. This is because unlike voice where 2-way interaction is often key and therefore data delivery constraints are very strict, video can often be extremely useful when consumed only in one direction (in a mountain rescue scenario this would namely be a stream from mountain side rescuers to a coordinator in the HQ). In this situation, video quality can be tolerated to degrade significantly as the resulting effect will only be a hindrance to the individual viewing the video stream, but they will still be able to receive good information from its availability. Beyond a certain point however a video stream will become so poor quality and such a poor representation of what is actually visible in the remote location that the viewer will lose trust in it altogether. In some situations an extremely poor quality video stream could result in a rescue coordinator viewing significantly outdated images and may ultimately be dangerous.

To test the capability of our video service over MANEMO networks we employed the same network topologies as those configured for testing in the MANEMO and Voice Service tests (described in Section 2.7 and carried out in Section 8). In particular we configured the network topologies illustrated in Figure 4, Figure 5 and Figure 6, but instead of analysing traffic flow and service between two devices on each mobile network, we instead analysed a traffic flow back to the HA, generated by video cameras connected to each mobile network. In addition, for this results section we only discuss the testing performed with the wireless camera as both cameras were seen to generate exactly the same flows of traffic and only the wireless camera was subject to further constraints imposed potentially by interference.

9.2.1. Video Service and MANEMO

In our video tests we wanted to analyse what occurred at the network layer (such as bandwidth utilisation and latency) and then compare these observations with the actual affects we witnessed on the live video stream. To determine the bandwidth used by our video service we initiated video streams from our cameras in each of the different modes that we wanted to test and then monitored the amount of traffic they generated at an intermediary point in the network. To carry out this monitoring we used the IO graphing facility provided by the Wireshark traffic analyser to illustrate the packet flow that specifically the video service was generating.

In our first set of tests we began by running the video service over a straightforward MANEMO topology consisting of only two wireless hops (one from the camera to the backpack router, and one from the backpack to the backhaul Access Point). In these sets of tests we found the throughput generated to be near identical to the bandwidth used over a purely wired, high performance network, this meant that none of the links in this topology acted as a bottleneck and likewise the latency experienced was very low (less than 20 ms).

Figure 35 and Figure 36 illustrate the throughput observed when a MJPEG and MPEG-4 stream were initiated by the end user, on this graph the x-axis represents time and the y-axis represents throughput in bits per second. In both cases the video stream was started in medium resolution and then allowed to run for a prolonged period of time, after which the stream was switched to the highest quality resolution and then subsequently followed by the lowest resolution. The throughput levels we observed during these tests were the same as with the basic tests across a wired network for both the low and medium quality streams. However, with the highest quality stream we clearly see a degradation in the throughput achieved to around about 4 Mbps, which has been imposed by the additional hop generated by the backpack router. This reduction in attainable throughput manifests itself as some slightly noticeable glitches in the resulting video images, with movement appearing a little less fluid. For the other two

resolution video streams, as expected, the resulting video images transmitted over this topology are clear and responsive as the camera is able to access all of the network resources it requires.

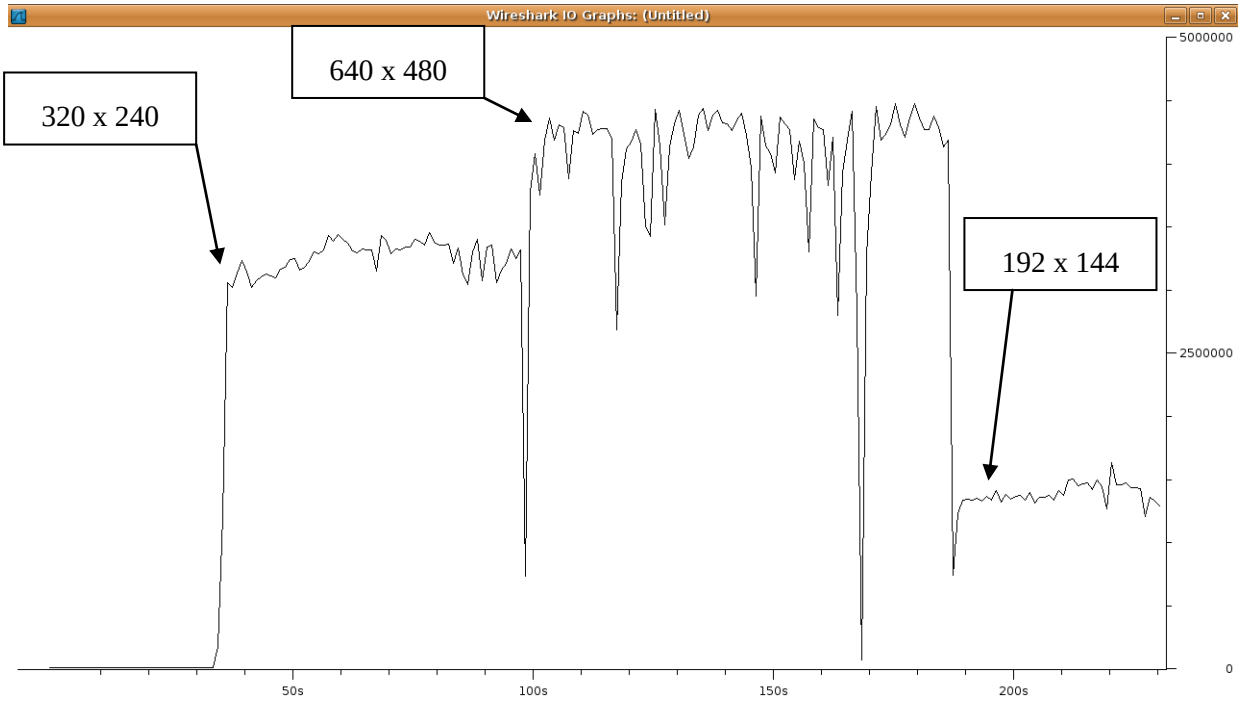

Figure 35 : Bandwidth Utilisation for MJPEG Stream

After testing the outcome of utilising our video service using the MJPEG codec we then performed a similar set of test over the same topology using the MPEG-4 codec that the cameras support.

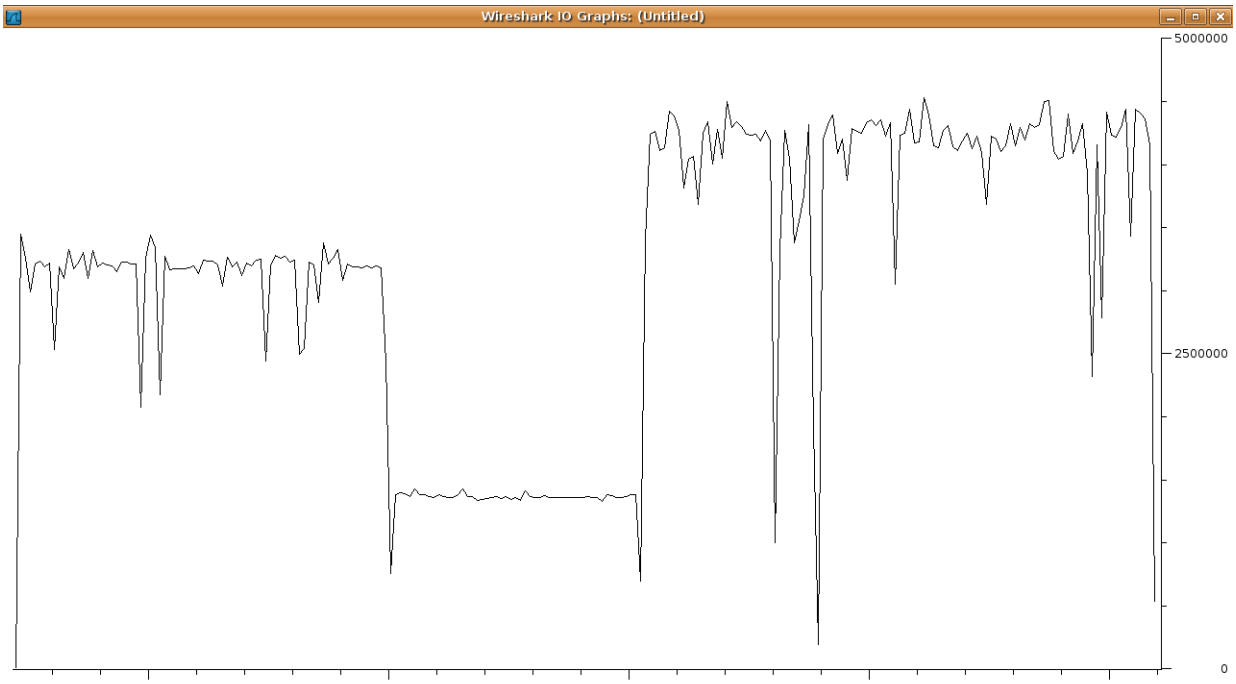

Figure 36 : Bandwidth Utilisation for MPEG-4 Stream

9.2.2. Video Service and MANEMO – Long Distance

After the first set of MANEMO tests were completed we then introduced a long distance route into the way that the video stream's packets were transmitted over the Internet. In order to achieve this we again established an IPv6-in-IPv4 tunnel via a broker in Hong Kong and routed traffic via this link in order to introduce a network latency of over 650 ms into the end-to-end path. The results for this stage of testing are illustrated in Figure 37 and Figure 38. Overall what they show is that the tunnel approach used throttles the total throughput achievable to just under 250 Kbps. In both of these graphs the video stream was first initiated in the medium quality resolution, then switched to high quality and then switched down to low quality. However, rather than there being any discernable increase or drop in the amount of data passed through the network, what we see instead is with each video different resolution stream there is a dip during changeover and then the camera returns to transmitting at the ceiling rate it can achieve.

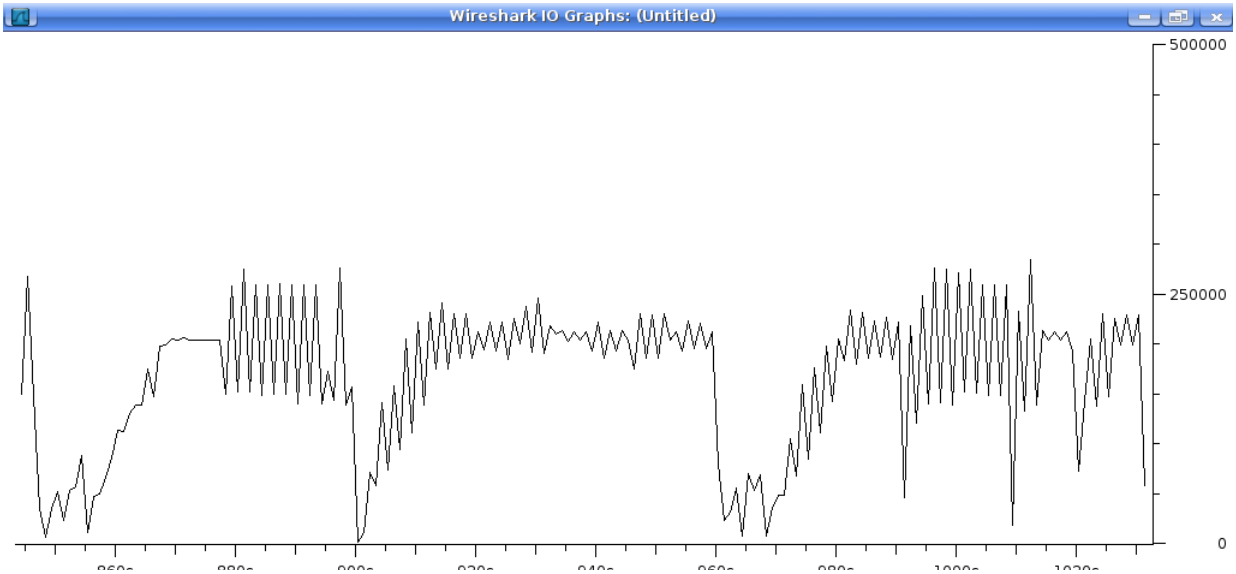

Figure 37 : Bandwidth Utilisation for MJPEG Stream - Long Distance

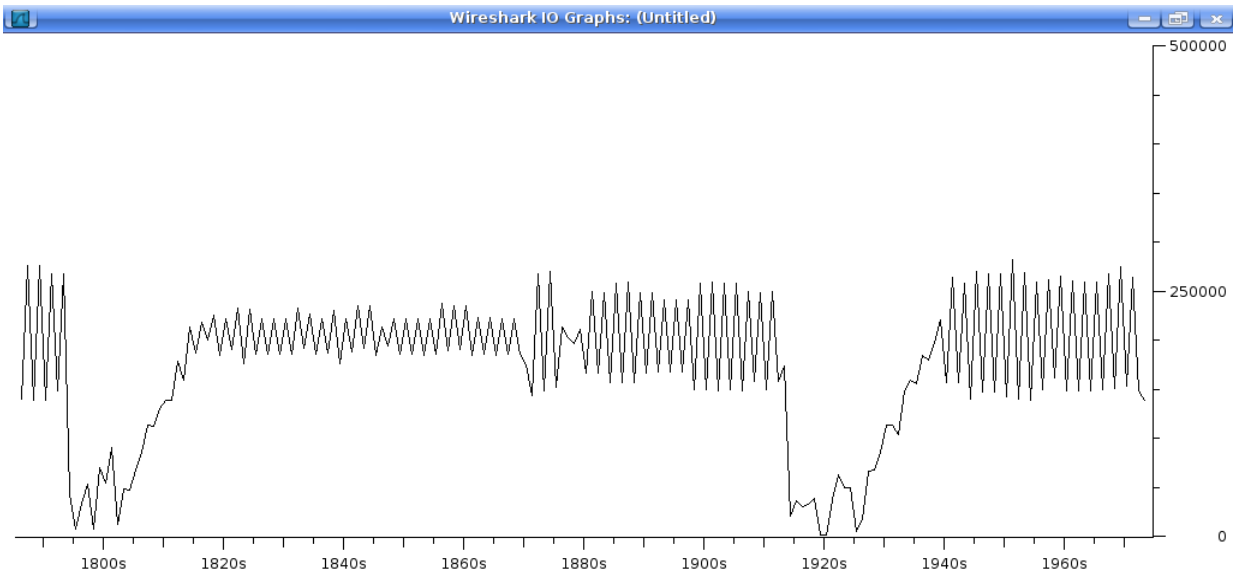

Figure 38 : Bandwidth Utilisation for MPEG-4 Stream - Long Distance

Interestingly however, even at this greatly reduced level of throughput, the video stream generated by the cameras is still useable by the end user. The fluidity of the footage is effected (movements appear jerky and slow) and the responsiveness is also compromised (actions appear to take around 2 seconds before they are shown to the end user), however in a mountain rescue scenario this video could still be extremely useful. In these scenarios the purpose of the streaming video is to provide the rescue coordinator with a better understanding of what is happening across the areas of operation. Even if the images the coordinator receives back to the HQ or mobile command post are disjointed and a few seconds delayed, they still provide them with an excellent understanding of the conditions and how things are unfolding across the search and rescue site.

9.2.3. Video Service and MANEMO – Large Number of Wireless Hops

For our final set of tests we again incremented the topology by introducing a number of extra backpack routers into the end-to-end path in order to increase the number of wireless hops the video data was required to traverse before it was again transmitted over the long distance backhaul connection. This test was designed to determine whether scenarios where video service traffic must first be transmitted via a number of other backpack routers before reaching a gateway that is able to transmit it back to the HQ could be supported. The results from this set of tests are presented in Figure 39 and Figure 40 below. As our results show, the introduction of four wireless hops into the end-to-end path causes a further level of degradation in the average throughput that the video camera can attain. In these sets of tests the camera was first set to transmit in medium quality, but then change to low quality and then high quality transmission after that. However in every case it is again not evident which video resolution is being transmitted because the video service is consuming the total amount of bandwidth available in every case. In particular, the introduction of multiple extra wireless hops before transmission across the long distance tunnel path can be seen to further reduce the available throughput to around 100Kbps and in addition, the transmission pattern experienced can be seen to be even more erratic.

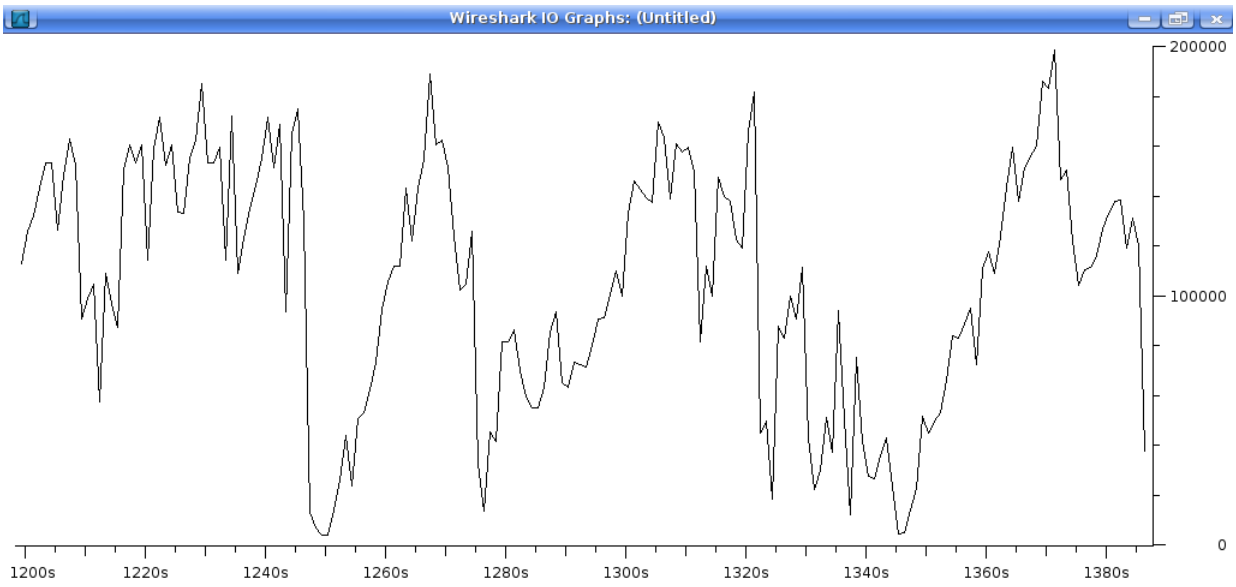

Figure 39 : Bandwidth Utilisation for MJPEG Stream - MANEMO + Long Distance

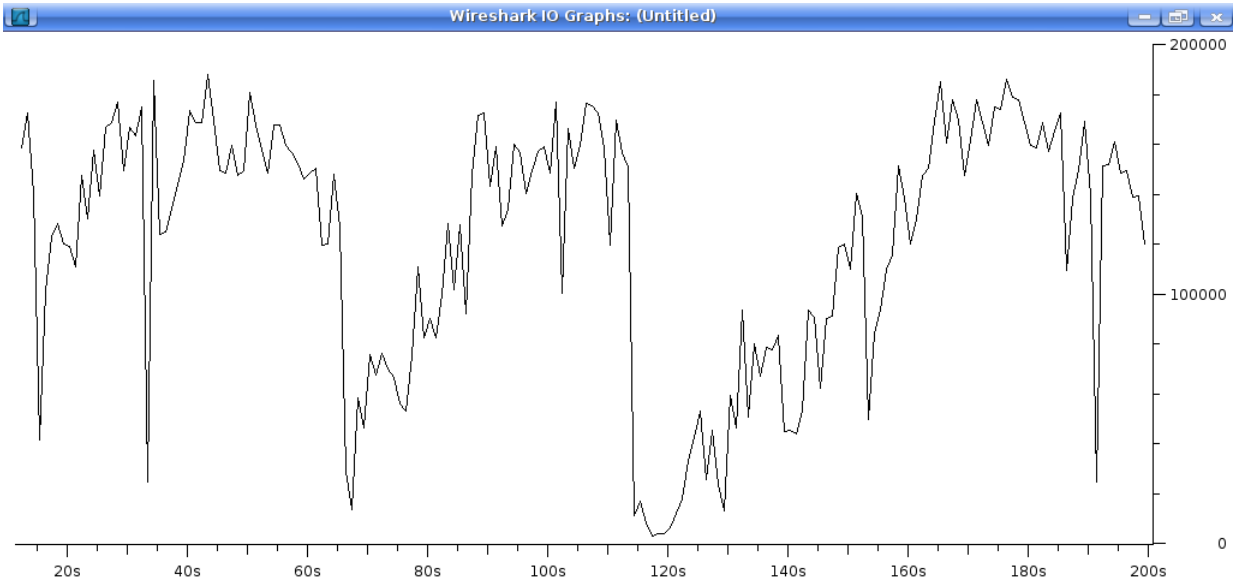

Figure 40 : Bandwidth Utilisation for MPEG-4 Stream - MANEMO + Long Distance

As with the tests performed in the previous subsection, this drastic reduction in throughput and erratic transmission pattern had consequences on the video images displayed. Again the transmission was seen to be even less fluid and responsiveness dropped to around 5 seconds before changes were observed. In this respect the video stream became similar to the periodic picture service (more like periodic updates of an image rather than a constantly moving video). Therefore in a scenario where this type of connectivity could be expected to be the typical connection available, default use of the picture service may be more suitable.

10. Evaluation Against Original Requirements

10.1. Communication Requirements

Reference	Pri.	Description	Achieved	Comments
MR-14-C-00-00	1	In general: anyone involved in a search and rescue operation should be able to communicate with anyone else involved in the same search and rescue operation.	✓	Provided by mobile devices, voice service, CaC software, IAN, Rapid Response PoPs and satellite backhaul.
HQ / 112 Centre				
MR-14-C-01-00	1	Where possible, controllers at the HQ or 112 Centre must have communication links with the Mountain Rescue Team Leaders at all times.	✓	Provided by mobile devices, voice service, CaC software, IAN, Rapid Response PoPs and satellite backhaul.
MR-14-C-01-01	2	The HQ or 112 Centre must have communication links with the Mountain Rescue Team Leaders when they are en-route to the search location(s).	✓	
MR-14-C-01-02	1	The HQ or 112 Centre must have communication links with the Mountain Rescue Team Leaders when stationed at search location bases.	✓	
MR-14-C-01-03	1	The HQ or 112 Centre must have communication links with the Mountain Rescue Team Leaders when conducting search and rescue operations away from location bases (i.e. when they are on the mountainside).	✓	
MR-14-C-01-04	2	The HQ or 112 Centre must have communication links with the Mountain Rescue Team Leaders when en-route between search location bases.	✓	
MR-14-C-01-05	1	(Slovenia): If there are personnel who remain at the MR HQ, they must have communications links with the 112 Centre (likely that future operations will see MR personnel located in their HQ/base to coordinate more complex/multiple missions).	✓	
MR-14-C-01-06	2	The controllers at the HQ or 112 Centre should have the capability to communicate directly with rescue workers involved in the search.	✓	

Reference	Pri.	Description	Achieved	Comments
MR-14-C-01-07	1	The HQ or 112 Centre should have communications with the helicopter agency and pilot (police, army, RAF) and know the helicopter location at any time. (Slovenia): 112 Centre in Kranj cannot communicate with a helicopter when it is in a location served by a different 112 Centre due to lack of correct procedures (i.e. notification of helicopter location).	~	Helicopters were not used in the trials. However, they can be integrated into the system by deploying the appropriate hardware.
MR-14-C-01-08	1	In multiple team searches, the HQ or 112 Centre must have communications links with Team Leaders from all participating Mountain Rescue teams.	✓	Provided by mobile devices, voice service, CaC software, IAN, Rapid Response PoPs and satellite backhaul
Team Leaders				
MR-14-C-02-00	1	Team Leaders must have communication links with all rescue personnel under their jurisdiction during the course of a search and rescue operation.	✓	Provided by mobile devices, voice service, CaC software, IAN, Rapid Response PoPs and satellite backhaul
MR-14-C-02-01	2	In multiple team searches, Team Leaders should have communication links with rescue personnel assigned to other Team Leaders.	✓	
MR-14-C-02-02	2	In multiple team searches, Team Leaders must have communication links with Team Leaders of all participating MR teams.	✓	
MR-14-C-02-03	2	Team Leaders should have direct communications with the helicopter agency and pilot.	~	Helicopters were not used in the trials. However, they can be integrated into the system by deploying the appropriate hardware.
MR-14-C-02-04	1	(Slovenia): If there are personnel who remain at the MR HQ, they must have communications links with the MR Team Leaders.	✓	Provided by mobile devices, voice service, CaC software, IAN, Rapid Response PoPs and satellite backhaul.
Rescue workers				
MR-14-C-03-00	1	(Slovenia): If there are personnel who remain at the MR HQ, they must have	✓	Provided by mobile devices,

Reference	Pri.	Description	Achieved	Comments
		communications links with the rescue workers.		voice service, CaC software, IAN, Rapid Response PoPs and satellite backhaul.
MR-14-C-03-01	2	Individual rescue workers may need the capability to have direct communications with the helicopter agency and pilot.	~	Helicopters were not used in the trials. However, they can be integrated into the system by deploying the appropriate hardware.
MR-14-C-03-02	1	Communications equipment carried by rescue workers must be capable of providing temporary communications to other rescue workers in the vicinity.	✓	Provided by backpack routers and Rapid Response PoPs.
Rescue vehicles				
MR-14-C-04-00	1	Rescue vehicles must be able to provide temporary communications for rescue personnel close to its location.	✓	Provided by vehicle routers and Rapid Response PoPs.
MR-14-C-04-01	2	The temporary communications provided by rescue vehicles should be able to adapt to the movements of the rescue personnel it is serving.	✓	Sector antennae can be steered to appropriate locations.
Technical				
MR-14-C-05-00	1	802.11b/g optimised for outdoor non-omni-directional coverage may be used to provide the temporary communications.	✓	Provided by backpack routers and Rapid Response PoPs.
MR-14-C-05-01	2	802.16 / WiMAX may be used to provide the temporary communications.	~	No 802.16/WiMAX service was used in the trials. However, such a service can be easily integrated into the IAN by deploying the appropriate hardware.
MR-14-C-05-02	1	Rescue workers must be able to take advantage of existing public or private communications infrastructure in addition to the temporary communications provided.	✓	Provided by backpack routers.

Reference	Pri.	Description	Achieved	Comments
MR-14-C-05-03	1	Communications equipment carried by rescue workers must be able to support a PAN (Personal Area Network) to connect together personal devices and any sensor equipment.	✓	
MR-14-C-05-04	1	Communications equipment should be able to use whatever network technology is available on location.	✓	Provided by MANEMO and the PMS.
MR-14-C-05-05	1	Communications equipment must be able to change their point of attachment to the network without breaking existing application sessions.	✓	
MR-14-C-05-06	2	Communications equipment should have the capability to use 2 or more networks simultaneously if available.	✓	
MR-14-C-05-07	1	Communications equipment must be able to switch from one available network technology to another without breaking existing application sessions.	✓	
Ad-hoc requirements				
MR-14-C-06-00	1	End nodes must configure themselves automatically with respect to network connectivity and appropriate authorisation and/or association protocols.	✓	Provided by backpack routers and MANEMO.
MR-14-C-06-01	1	Nodes must automatically discover the appropriate routes and gateways for the ad-hoc network.	✓	
MR-14-C-06-02	1	Nodes must adapt to changes in connectivity, routes and available gateways without breaking existing application sessions.	✓	
Security				
MR-14-C-07-00	1	Appropriate access controls must be provided to prevent misuse of the system.	✓	Provided by backpack routers / IAN and CaC software.
MR-14-C-07-01	1	Appropriate authorisation methods must be in place to identify valid users of the system.	✓	
MR-14-C-07-02	1	Sensitive data must be sufficiently encrypted to prevent eavesdropping by third parties. Sensitive data includes any communications with military entities and any medical telemetry.	✓	
MR-14-C-07-03	2	Users must not have to manually enter security credentials at any time, in order to gain access to the network or to secure their communications.	✓	

Reference	Pri.	Description	Achieved	Comments
MR-14-C-07-04	2	When moving to a different network, any establishment of authentication/access control for the new network must not adversely affect existing application sessions.	~	When roaming to networks outside the control of the Mountain Rescue teams, delays can adversely affect applications.
Other				
MR-14-C-08-00	1	Voice communications must be supported for one-to-one, group and open broadcast.	✓	Provided by voice service.
MR-14-C-08-01	1	Data communications must be supported for 1-to-1, group and open broadcast.	✓	Provided by applications and IPv6.
MR-14-C-08-02	1	Call setup must be automatic once the person/group is selected by the user.	✓	Provided by voice service.
MR-14-C-08-03	1	Rescue workers' devices must contain GPS modules to provide location updates to the monitoring and management middleware.	✓	Provided by the PMS PDAs and CaC software.
MR-14-C-08-04	1	When the casualty is located the appropriate communications device must broadcast a 'casualty located' signal containing location coordinates to all search and rescue operatives.	✓	
MR-14-C-08-05	2	Anyone must be able to call a mobile phone number from within the proposed system.	✗	The voice system does not currently support GSM calls.
MR-14-C-08-06	1	Both the UK and Slovenia cases require logs of all communications between the HQ or 112 Centre and anyone else (MR personnel, police, ambulance, helicopter agency, casualty etc.).	✓	Provided by the CaC software and voice system.
MR-14-C-08-07	3	Communication logs between MR personnel are not required although the functionality may be provided.	✓	
Quality of Service				
MR-14-C-09-00	1	Whenever possible, the network should guarantee the resources required by the user applications in use.	✗	QoS was not implemented due to the complexities of the MANEMO protocols. Further research is required to solve this issue.
MR-14-C-09-01	1	The network must serve application traffic in accordance with the importance of the particular application type and the rank/authority of the end users.	✗	
MR-14-C-09-02	1	In times of limited network resources, less important applications and users must yield	✗	

Reference	Pri.	Description	Achieved	Comments
		their resources to higher ranked applications and users.		
MR-14-C-09-03	2	When utilising public networks, Mountain Rescue personnel should have priority access over the general public for all application types.	✘	

10.2. Application/Middleware Requirements

Reference	Pri.	Description	Achieved	Comments
MR-14-A-00-00	1	The control room at HQ/112 Centre must have application software (middleware) to help controllers monitor, manage and control search and rescue operations.	✓	Provided by the CaC software, voice system and the PMS.
MR-14-A-00-01	1	The application/middleware will maintain communication links with all MR personnel and vehicles for the entire duration of a search and rescue operation.	✓	
MR-14-A-00-02	1	The application/middleware must maintain the locations of all MR personnel and vehicles during a search and rescue operation. This also includes helicopters.	✓	
Presence Management / Localisation				
MR-14-A-01-00	1	The current locations of MR personnel and vehicles will be displayed on-screen and updated whenever new location updates are received.	✓	Provided by the CaC software and the PMS.
MR-14-A-01-01	2	GPS coordinates are the preferred format for location information passed between applications.	✓	
MR-14-A-01-02	1	Where GPS coordinates are not possible, the applications must be able to translate other formats (e.g. Ordnance Survey grid references).	✓	
MR-14-A-01-03	2	Update frequencies will differ according to the owner of the GPS module sending the updates (e.g. walkers, dogs, helicopters etc). Therefore update frequencies must be tuneable according to the nature and average speed of the owning entity.	✓	
MR-14-A-01-04	1	The location display must be overlaid onto a 2D geographical map.	✓	
MR-14-A-01-05	2	The scale of the map display should include 1:25000, 1:50000 and 1:100000 versions.	~	Only the 1:25000 scale

Reference	Pri.	Description	Achieved	Comments
				is currently used.
MR-14-A-01-06	2	Photographic maps should also be used where possible.	✓	Provided by the CaC software.
MR-14-A-01-07	3	If possible, the application should be able to display locations overlaid onto a 3D terrain map. This should be linked in with existing GIS, rather than a new system built from scratch.	✓	
MR-14-A-01-08	2	If a rescuer or vehicle has not updated its location within the designated timeframe, the application/middleware must attempt to contact the relevant device to establish its location. The last known location should be logged. A suitable warning should be shown on the screen to indicate a possible problem.	✓	Provided by the CaC software and the PMS.
MR-14-A-01-09	2	The application/middleware must be able to differentiate between stationary people/vehicles and loss of contact. For example, the display should have different icons (or colours) for the same entity when it is stationary and when contact has been lost.	✓	
MR-14-A-01-10	1	Users must have the ability to choose what is seen on the screen and add/remove details. For example, select rescuers, vehicles, helicopters or team leaders etc. to be displayed (or not displayed).	✓	Provided by the CaC software.
MR-14-A-01-11	1	When a casualty is located, the application must respond to the 'casualty located' signal in the appropriate manner.	✓	Provided by the CaC software and the PMS.
Databases				
MR-14-A-02-00	1	The middleware at HQ/112 Centre must contain several knowledge databases to aid in search and rescue operations.	✓	Provided by the CaC software.
MR-14-A-02-01	1	The middleware must contain a personnel database containing information of all rescue personnel, their availability, their relevant skills, experience and location/contact information.	✓	
MR-14-A-02-02	1	The middleware must use the personnel database to automatically page or SMS the relevant personnel in response to the details of an emergency call.	✓	
MR-14-A-02-03	1	The middleware must contain a Geographical Information System (GIS) giving information on the respective	✓	

Reference	Pri.	Description	Achieved	Comments
		geographical area.		
MR-14-A-02-04	1	The middleware must contain a database of previous incidents and relevant statistical data.	✓	
MR-14-A-02-05	1	The previous incidents database must be updated after every search and rescue operation. This should be done an easy manner and automated as much as possible.	✓	
MR-14-A-02-06	1	The middleware should contain a communications database which shows what permanent wireless communications are available throughout the geographical area.	✓	
MR-14-A-02-07	2	The middleware should use search theory algorithms to suggest suitable search patterns from all information available and by cross-referencing the knowledge databases.	✓	
MR-14-A-02-08	1	By utilising the information logged in the databases, the middleware must be able to provide a reconstruction of a previous search and rescue operation.	✓	
MR-14-A-02-09	2	The middleware should use the communications and previous incidents databases and cross reference with search theory algorithms. The middleware can then suggest optimum locations to which rescue vehicles should be despatched with respect to: • the information from the emergency call • likely locations from statistical evidence • road access and available paths • available communications	✓	
MR-14-A-02-10	2	The middleware should suggest an assignment of available personnel to the different search locations and their roles in the search and rescue operation.	✓	
MR-14-A-02-11	2	The users of the middleware (controllers, team leaders) must be able to accept, reject or modify suggestions made by the middleware.	✓	
Team Leaders				
MR-14-A-03-00	1	The Team Leaders must have application software on their personal devices to help	✓	Provided by the CaC

Reference	Pri.	Description	Achieved	Comments
		them monitor, manage and control their subordinates during search and rescue operations.		software and the PMS PDAs.
MR-14-A-03-01	1	The Team Leaders' software must update the HQ/112 Centre of their location every 30 seconds.	✓	
MR-14-A-03-02	1	Team Leaders must be able to view their current location overlaid on a 2D geographical map on their personal devices.	✓	
MR-14-A-03-03	2	The scale of the map display should include 1:25000 and 1:50000 versions.	~	Only the 1:25000 scale is currently used.
Rescue Workers				
MR-14-A-04-00	1	The rescuers must have application software on their personal devices to help them monitor and manage their search and rescue operations.	✓	Provided by the CaC software and the PMS PDAs.
MR-14-A-04-01	1	The rescuers' software must update their Team Leaders and HQ/112 Centre of their location every 30 seconds.	✓	
MR-14-A-04-02	1	Rescue workers must be able to view their current location overlaid on a 2D geographical map on their personal devices.	✓	
MR-14-A-04-03	2	The scale of the map display should include 1:25000 and 1:50000 versions.	~	Only the 1:25000 scale is currently used.
Rescue Vehicles				
MR-14-A-05-00	1	Middleware located with communications equipment in rescue vehicles must monitor the locations of rescue workers on the temporary network.	✓	Provided by the CaC software and the PMS.
MR-14-A-05-01	1	Middleware in the vehicles must aim to provide maximum possible coverage to all rescue workers as they move during search operations.	✓	Provided by the CaC software and the PMS. Sector antennae can be moved automatically.
MR-14-A-05-02	2	The middleware in the vehicles may be connected to intelligent and moveable antennae to help with maximising network coverage.	✓	
MR-14-A-05-03	1	An appropriate display located with the vehicle should show information pertaining to rescue workers connected to the temporary network and other networks	✓	Provided by the CaC software and the PMS.

Reference	Pri.	Description	Achieved	Comments
		connected to the vehicle.		
MR-14-A-05-04	1	It must be possible to manually change the parameters of the temporary network (e.g. antenna position, transmission power, data rates etc.).	✓	Provided by standard configuration interfaces.
General				
MR-14-A-06-00	1	When any rescuer locates a casualty, their device software must transmit a 'casualty located' signal, including relevant location data, to all MR personnel, vehicles and HQ/112 Centre.	✓	Provided by the CaC software and PMS.
MR-14-A-06-01	1	Displays on rescue workers' equipment must be simple and show functions selected and the communication mode (1-to-1, group, broadcast).	✓	Provided by PMS and voice service applications.
MR-14-A-06-02	1	Call setup should be automatic once the function and person/group is selected by user.	✓	Provided by voice service.
MR-14-A-06-03	2	The middleware should be intelligent enough to provide "content adaptation" inside the network to optimise information flow with respect to available network resources and the number of users.	✗	Not implemented.
MR-14-A-06-04	2	The middleware should contain a voice-to-text capability so that voice semantics can be transferred across the network as text when network conditions will not support the amount of voice traffic required.	✗	
MR-14-A-06-05	3	The middleware should contain a video adaptation capability so that video streamed over the network will be adjusted according to available network resources.	✓	Provided by video service.

10.3. Hardware Requirements

Reference	Pri.	Description	Achieved	Comments
MR-14-H-00-00	1	Communications equipment to be carried by rescue workers must not impede the individual by being too heavy or bulky.	✓	Provided by backpack routers.
MR-14-H-00-01	1	Communications equipment carried by rescue workers should not exceed a total weight of 2.5 Kg, around the same weight as an average laptop.	✓	

Reference	Pri.	Description	Achieved	Comments
MR-14-H-00-02	1	Communications equipment carried by rescue workers should not exceed a size of 200x200x100mm (Length x Width x Depth).	✓	
MR-14-H-00-03	1	Communications equipment carried by rescue workers must be carried inside an existing backpack so that hands are unimpeded.	✓	
MR-14-H-01-00	1	It must be possible for workers to communicate in a hands free manner (e.g. using Bluetooth headsets or similar).	✓	Provided by PDAs.
MR-14-H-02-00	1	All communications equipment carried by rescue workers must be able to operate in all weather conditions.	✓	Provided by backpack routers and PDAs with waterproof jackets.
MR-14-H-02-01	1	All communications equipment carried by rescue workers must be resistant to wet and damp conditions.	✓	
MR-14-H-02-02	2	All communications equipment carried by rescue workers must be able to operate between -30C and 50C.	✓	
MR-14-H-02-03	1	All communications equipment carried by rescue workers must be able to operate in high winds.	✓	
MR-14-H-02-04	1	Screen displays with rescue workers must be readable at night, in low-light conditions and in sunshine. The screens must also be resistant to sun and/or snow glare.	~	PDA screens were often unreadable in strong sunlight or snow glare.
MR-14-H-03-00	1	The batteries of communications equipment carried by rescue workers must provide at least 4 hours of full usage.	✓	Provided by backpack routers and PDAs.
MR-14-H-03-01	1	Communications equipment located in vehicles can take advantage of the vehicle's battery power but must also be able to operate under their own power for at least 6 hours of full usage.	✓	Provided by vehicle routers
MR-14-H-04-00	1	Rescue vehicles must be equipped with appropriate hardware to provide temporary communications for rescue teams.	✓	
MR-14-H-05-00	1	For the user equipment, there should be different functions accessible in an easy 'push-to-talk' manner. Buttons or touch screens must be of sufficient size so they can be easily pressed with gloves.	✓	Provided by PMS and voice service applications.

Reference	Pri.	Description	Achieved	Comments
MR-14-H-05-01	1	The user equipment must be wearable (e.g. on forearm) to minimise disruption to search and rescue activities.	✓	Personal routers are worn in backpacks. PDAs can be worn on forearms if desired.

10.4. Failure Requirements

Reference	Pri.	Description	Achieved	Comments
MR-14-H-00-00	1	Communications equipment to be carried by rescue workers must not impede the individual by being too heavy or bulky.	✓	Provided by backpack routers.
MR-14-H-00-01	1	Communications equipment carried by rescue workers should not exceed a total weight of 2.5 Kg, around the same weight as an average laptop.	✓	
MR-14-H-00-02	1	Communications equipment carried by rescue workers should not exceed a size of 200x200x100mm (Length x Width x Depth).	✓	
MR-14-H-00-03	1	Communications equipment carried by rescue workers must be carried inside an existing backpack so that hands are unimpeded.	✓	
MR-14-H-01-00	1	It must be possible for workers to communicate in a hands free manner (e.g. using Bluetooth headsets or similar).	✓	Provided by PDAs.
MR-14-H-02-00	1	All communications equipment carried by rescue workers must be able to operate in all weather conditions.	✓	Provided by backpack routers and PDAs with waterproof jackets.
MR-14-H-02-01	1	All communications equipment carried by rescue workers must be resistant to wet and damp conditions.	✓	
MR-14-H-02-02	2	All communications equipment carried by rescue workers must be able to operate between -30C and 50C.	✓	
MR-14-H-02-03	1	All communications equipment carried by rescue workers must be able to operate in high winds.	✓	

Reference	Pri.	Description	Achieved	Comments
MR-14-H-02-04	1	Screen displays with rescue workers must be readable at night, in low-light conditions and in sunshine. The screens must also be resistant to sun and/or snow glare.	~	PDA screens were often unreadable in strong sunlight or snow glare.
MR-14-H-03-00	1	The batteries of communications equipment carried by rescue workers must provide at least 4 hours of full usage.	✓	Provided by backpack routers and PDAs.
MR-14-H-03-01	1	Communications equipment located in vehicles can take advantage of the vehicle's battery power but must also be able to operate under their own power for at least 6 hours of full usage.	✓	Provided by vehicle routers
MR-14-H-04-00	1	Rescue vehicles must be equipped with appropriate hardware to provide temporary communications for rescue teams.	✓	
MR-14-H-05-00	1	For the user equipment, there should be different functions accessible in an easy 'push-to-talk' manner. Buttons or touch screens must be of sufficient size so they can be easily pressed with gloves.	✓	Provided by PMS and voice service applications.
MR-14-H-05-01	1	The user equipment must be wearable (e.g. on forearm) to minimise disruption to search and rescue activities.	✓	Personal routers are worn in backpacks. PDAs can be worn on forearms if desired.

10.5. Other Requirements

Reference	Pri.	Description	Achieved	Comments
MR-14-L-00-00	1	The emission power levels of all wireless equipment must be within all legal requirements.	✓	Legal power emissions met by all devices. All devices are configurable.
MR-14-L-00-01	1	Legal levels for outdoor use may be too extreme for backpack routers. Therefore, emission power levels for backpack routers must be reduced accordingly.	✓	
MR-14-L-00-02	1	Emission power levels on wireless equipment must be tuneable to the required level.	✓	

Reference	Pri.	Description	Achieved	Comments
MR-14-L-01-00	1	The addition of any equipment and its usage must not contravene and Health and Safety regulations.	✓	

11. Conclusions and Further Work

In this final chapter we provide our conclusions on each of the major components of our mountain rescue communications solution and discuss how work into each of the areas will continue in the future.

11.1. Presence Management Conclusions

After several on-campus, field and on-mountain test the PMS client application was found to behave in a very reliable and stable way and was actually demonstrated in various events, such as the demonstration at Ig, Slovenia and at the Final Review and Demonstration at Grouft, in Luxembourg. Its performance has been found to be vastly improved from earlier implementations and the acquisition of the GPS coordinates is considered very accurate.

Network wise, the PMS client uses IPv6 for transmitting data over the Wi-Fi network that is provided by IAN at the search and rescue region. A disappointing step back was the decision not to use GPRS as a connectivity option, mainly because of the swapping bug of Windows Mobile and the poor GSM signal in the area of the Lake District. Using the GPRS connection for transmitting data from the PMS client has been fully implemented and the team is looking forward to finding ways to include this connectivity option as well in the future, and to increasing the redundancy of the transmission where Wi-Fi connectivity is not available.

The feedback that we got for the PMS client application from rescuers was very positive as they were happy with the user friendliness and ease of operation of the application. However, neutral and negative feedback was received for the actual hardware (PDA devices) that we used for the PMS, due to its non-working under bad weather conditions. The PDA devices that we used as a proof of concept were not waterproof or ruggedised and rescuers were having difficulties in seeing the screen under sunlight or when using the devices whilst wearing gloves. Ideally, we would like to run the PMS client in specifically developed ruggedised hardware that would be wearable, waterproof, could be powered for several hours and be easily operated with gloves in bad weather conditions, which is something that we would explore in the future.

11.2. Backpack Router Conclusions

The backpack router is a key physical component in our overall mountain rescue solution. We have performed extensive tests to determine its suitability for use in mountain rescue scenarios and, in conclusion, believe that the backpack router is suitable for use in trials and demos, however further work is needed on the hardware components to realise a product ready for full deployment. In particular, for further revisions of the hardware design we will specifically aim to reduce the footprint of the device, to improve the waterproofing in general and to make the internal cabling and fixings neater and more robust. Reducing the size of the router will make it more suitable for use in the backpacks that rescue teams already use on a day-to-day basis, and improving the cabling and fixings will make the router more resistant to long term, sustained vibration and shock. Finally, here is a need for further waterproofing which stems from an unsuitable switch design that we chose to incorporate into our enclosure early on in our research. For this particular item we need to go back and reconsider our switch options in general and take more consideration of the intricate properties of the switch itself and not just the way the switch is installed. Overall, we are happy with this outcome, especially since the provisioning of hardware for continuous use in these harsh environments is an extremely specialised field in which we have no previous experience.

In addition to the physical attributes of the backpack router itself, we found the effective wireless communication range, that was achievable, to be better than expected, and therefore very positive overall. Wireless signal propagation is again a very specialised subject and so far have only used simple, generic 5dbm omni-directional antennas with the backpack router. These are inexpensive, off-the-shelf antennas that are used in everyday indoor wireless scenarios and we have found them to perform much better than originally expected. For this reason, this is another area where potentially we expect to be able to make further gains in the future. At present our main focus is on the development of our networking protocols and software, with the hardware considerations being secondary. However with the further stabilisation of our networking protocols over time we will be able to commit more effort to researching very specific aspects of our hardware solution to make further gains. Antenna design is certainly one of those areas that we will focus on. Through the use of more specialised and higher quality equipment hopefully we will be able to increase the effective wireless communication range of the backpack routers to be significantly better than the already satisfactory levels we are currently achieving.

In general, one of the major benefits of developing a solution for the harsh and difficult environments that a mountain rescue team operates in, is that our solution is applicable to many other, less taxing scenarios. For example, in everyday emergency services scenarios where the router can be expected to be housed in a relatively stable vehicle or a backpack that is infrequently exposed to heavy rain or persistent vibration, then our solution can already be seen to be potentially suitable. This means that by initially setting out to solve one of the most difficult examples of mobile networking, we have simultaneously developed a solution that is suitable for use in many other important use case scenarios.

11.3. Satellite and Backhaul Links Conclusions

The results of the satellite and backhaul link tests demonstrated to us that the network performance of both approaches were suitable for use as part of our overall communication solution. The throughput achieved in each case was easily enough to support the lightweight transmission of localisation data, as well as multiple voice call streams or degraded video service output. However, aside from the networking considerations, it was the logistic factors surrounding the two wide area communication technologies we tested that provided the most problems. The relatively heavy, bulky and power hungry equipment required to establish these links was deemed to be inappropriate for use in a real world mountain rescue deployment. Therefore a more feasible solution to providing the mountain rescue IANs with dependable Internet connectivity must still be found.

Our most immediate focus in this area will be on trialling the use of satellite services offering the ability to use much smaller dishes in the North of the UK (specifically Inmarsat dishes). In addition to experimenting with this much smaller dish solution, we intend to use a bracket mounted, moveable satellite dish arrangement that can be permanently fixed to the top of a rescue vehicle. Finally we intend to use a dish solution which can automatically determine its location using GPS and position itself automatically to align with its appropriate satellite service correctly. Using this type of solution will address all of the problems identified with the wide area connectivity technologies we previously trialled, as it would not take up additional inside space because of its roof mounting and it also would not consume precious rescue team member time as it would align itself automatically. In addition, since all equipment, apart from the dish, could be specially housed in the rescue vehicle, the issues of weatherproofing and power supply would also be addressed.

11.4. Command and Control Software Conclusions

The CaC interface is a very substantial piece of software that combines many different services ranging from alerting rescuers to monitoring them and to providing video, picture and messaging services during emergency scenarios. Its implementation is of substantial size including tens of thousands lines of mainly C# code and presents two basic forms to the user. Its functionality is considered to be stable, taking into account that we were not looking to develop a commercial product but a prototype to be used in our Mountain Rescue scenarios.

CaC has been tested and evaluated thoroughly using many different types of tests and was found to run in the expected manner. The alerting functionality provided by M-Plify was found to be very useful by the rescuers who currently use a paging system to inform them of an incident. Future work could be done to upgrade our CaC client implementation to version 5 of AlarmTilt.

The video and picture service from within CaC was also found to be operating very nicely and although network constraints (low bandwidth and high delay) degraded these services, they were still found to be very useful. Future work should be done to provide a better interface for providing video feed from multiple cameras that should be linked more easily with the rescuers carrying them. Although the detached functionality of the windows that provided the video service was found to be useful, there is some consideration as to whether the mission coordinator gets distracted by multiple different windows on the screens. This needs to be explored more by doing more user tests and by monitoring how the mission coordinator interacts with the software.

The presence management service of CaC was greatly appreciated by the CMRT and was found to be very useful. The interactive and highly detailed Ordinance Survey maps were of high importance to the coordinator who was able not only to monitor the rescuers in real-time, but also to dynamically add and remove information being displayed on the maps. One concern that has been partially addressed, is that due to the high resolution OS maps that are being used, the software is resource intensive although the use of resources is now way better than it was at its initial implementation. Future work can be done to refine even more the way maps are displayed and provide a quicker navigation. Further work could also include recording the video, pictures and audio feed that is being received and tight it with the recording of the movement of the rescuers at the search and rescue field.

In conclusion the CaC interface has been a major part of our Mountain Rescue Solution and of significant importance to the CMRT. Its objectives have been met beyond expectations, but future work could be done to improve it, especially on how the information is represented to the mission coordinator, in order to advance its knowledge for the mission in a more concise and more manner.

11.5. MANEMO Conclusions

By integrating NEMO techniques with existing MANET technology our UMA protocol is able to provide a comprehensive solution to providing global connectivity to MANET scenarios. The UMA approach has been designed to support an entire mobile network of hosts. In doing so it ensures that all hosts connected to any UMA enabled mobile networks are not required to take part in any form of mobility signalling themselves, as the UMA enabled mobile router will perform this functionality on their behalf.

Supporting this capability ensures that any nodes connected to UMA enabled mobile networks (such as Personal/Vehicle Area Network nodes) need only be standard IPv6 hosts. This in turn ensures that nodes connected to UMA enabled mobile networks can communicate constantly across the Internet using the same address regardless of their physical location, without their TCP sessions being dropped whenever a roam takes place. It also means that nodes in the MANETs can be directly contacted from anywhere in the Internet, without having to establish a prior flow of packets. In addition to these benefits, we also

strived to ensure that the UMA approach did not affect the current Internet architecture by requiring any augmentation of the core infrastructure or in any access networks. By achieving this aim we are able to present UMA as a mobility solution that is immediately suitable for use across IPv6 enabled networks. This is an important consideration since the number of different providers offering Internet access is already significantly large. For this reason, a solution which relies on Access Routers in these provider networks being augmented to support its functionality could be excessively difficult to deploy. As well as a large number of Internet access providers, there also exists numerous different technologies that can be used to connect to the Internet that each possess differing network characteristics. The ability to simultaneously use as many of these connectivity options as possible through the use of a Multihoming approach can provide significant improvements to the robustness of mobile networking scenarios.

For this reason UMA was designed and implemented in a manner which inherently supports this capability through the use of multiple simultaneous network location bindings, and this will be explored further as part of our future work. Using a testbed configured to replicate a realistic UMA communication scenario we also carried out a performance evaluation of the experimental implementation of our protocol. The results of this experimentation were considered to be very encouraging. Our intention for the UMA protocol was to design an approach which could provide global reachability for MANET nodes with a handover performance that was as good as or better than shown by the NEMO Basic Support protocol with individual mobile networks. Through the results of our testing with the UMA protocol we have shown that in every configuration that arises when using UMA we achieve that goal and in certain cases, notably improve on the performance of NEMO BS. In addition to highlighting the actual performance of the UMA protocol we also configured a second testbed using wide area Internet access technologies and distributed Home Agents that was intended to demonstrate UMAs suitability for immediate deployment over the existing Internet infrastructure. Using the UMA protocol we were able to provide MANET nodes with the benefits of global reachability via access networks including a satellite communication link and a HSDPA cellular connection. This capability would simply not be possible using any other existing proposed solutions to this problem as it would require permission to install experimental software on the Access Routers of the respective networks.

Therefore by combining the properties of both MANET and NEMO techniques we feel we have been able to produce and demonstrate a versatile and efficient approach to extending the functionality of MANETs that is immediately deployable without any alterations required to the existing Internet architecture.

11.6. Voice Service Conclusions

Our voice service trials and the results we gathered from our tests were deemed to be extremely positive. Our decision to develop a bespoke Voice-over-IPv6 implementation for mobile handsets has, in hindsight, turned out to be a very good choice. As our testing demonstrated, the low throughput requirements of our applications coupled with the relatively high throughputs achievable using our MANEMO solution have permitted us to offer very clear, stable voice communications over very complicated mobile network topologies. A particular example of this functionality was the final set of tests performed in the voice service testing section where we were able to conduct perfectly suitable and clear two-way conversations over network topologies containing multiple wireless hops through a number of backpack routers and then transmitted across an elongated path via a server in Hong Kong. This in total tested our voice application's ability to perform over lossy links with high latency and still the service was more than satisfactory. In addition to the performance of the application, developing our own implementation meant that we were able to customise the functionality of the application very easily as we identified different and new requirements.

In particular, we have been able to adapt the functionality of our voice application to specifically suit the needs of a mountain rescue team, taking into account the individual nature of our on-mountain wireless network infrastructure. This was highlighted in our testing phase when we observed that existing pieces of software (such as Linphone) detect that a connection has been lost on the remote end and subsequently shut down the call on the local side. This is a very significant design decision and has important consequences in our mountain rescue scenario. If we were to try and use a piece of voice software that implemented this type of functionality, the rescue workers would be required to constantly re-establish calls whenever they went out of range with an access point or even handed over between networks (depending on the length of time of the handover). With our application the software continues to transmit (whilst it is in "Talk" mode) indefinitely, this means that if a rescue worker has moved out of range of a network temporarily, as soon as they return and establish their Wi-Fi connection the voice call will resume. In addition to the network layer benefits, our application offers, developing the code in-house also means that we can adapt the interface according to our needs as well. This is of particular importance when we consider the requirements of multi-person, group calls. With most openly available VOIP applications that support many-to-many calling functionality, the interface is designed to be controlled by a mouse and keyboard. List menus of group members available to be added to a call are manipulated by selecting each individual person to be included with a mouse pointer. This is obviously not feasible for rescue workers in the field of operation, and even if touch-screen interfaces are used it represents a lot of effort and time wasted initiating a call when all the rescue worker wishes to do is speak. With this in mind, we are able to design our interface to incorporate "Group" buttons where all the rescue worker has to do is press one button and a call will be initiated with a number of predefined people at once. For example, a group button can be added to call all members of the rescue worker's separate search party, or another can be added to immediately call all search coordinators, etc.

As for our future direction with our voice service, it is clear that we are happy with the performance and functionality of the core software as it stands, so our efforts will be focussed on the software peripheral to its core functionality. For example we will aim to improve the way that the voice application is automatically identified and registered as part of the IAN and facilitate the coordinator based in the HQ to very quickly be able to establish multiple calls. In addition we will also research possible ways to initiate and stop calls in a completely hands-free manner, allowing the rescue worker to use a hands free headset (which is currently supported) and stop and start calls without ever touching the mobile handset.

11.7. Video Service Conclusions

In the video service trials we discovered a number of interesting properties concerning the streaming video cameras that we chose to incorporate. One of the biggest impressions was the relatively high bandwidth consumed by our service. At its highest quality the video cameras we currently use require over 5 Mbps of bandwidth to transmit images as they are intended to be viewed. The medium resolution video stream however is still entirely suitable for use in the mountain rescue scenario but, at over 3 Mbps, still requires a lot of bandwidth. As the video stream is implemented to transmit over the TCP transport protocol, it will however back off when multiple different cameras are transmitting over the same saturated link and instead of breaking the stream, the service will simply degrade. This is a very big positive attribute of the video service as even very restricted network conditions will still permit images to be transmitted back to the headquarters (albeit less responsively). The combination of the results from the power consumption tests and the performance tests has led us to conclude that wherever possible, the use of a wired video camera, connected into a mobile router via Ethernet is a much better option in deployment cases and should ultimately be strived for. Wherever this is not possible the wireless video cameras can be introduced as a secondary option.

As future work in this area we aim to acquire different streaming IP webcams from other manufacturers to compare their respective performance with the current batch of Panasonic units that we employ. If we remain unhappy with all the commercially available options we will then consider developing our own

streaming media server ourselves, using a small lightweight main board platform such as a Gumstix board, with a streaming video camera attached to the board via USB. Using this hardware we could then setup an IPv6 streaming media server on the main board and communicate with the on-maintain network using an onboard Wi-Fi interface. By building our custom solution we would then be able to experiment with different video codecs and transmission techniques designed more specifically for use in lossy, lower bandwidth networks.

11.8. Future Research Work

In addition to the improvements and future development work that we intend to carry out in the areas specifically related to our mountain rescue solution, we have also identified a number of mobile networking research areas (highlighted by our work in the MANEMO problem space) that we will attempt to solve in the future. Within the lifetime of the U-2010 project we were able to develop a working implementation of our Unified MANEMO Architecture (UMA) protocol and demonstrate its capabilities and potential using real testbeds and hardware. This implementation is a fully functioning prototype of the basic UMA design we initially proposed and offers all of the beneficial properties related to global reachability and session continuity in MANETs and Multihop Mobile Networks that we originally set out to provide. However, our breakthroughs and advances in this area have in themselves highlighted further areas for research that can bring new, previously unachievable functionality to these mobile scenarios, and it is our intention to continue to solve these brand new challenges in the future.

In particular, the UMA approach introduces potential advantages in areas such as Multihoming and Authentication, Authorisation and Accounting (AAA) for Internet-connected MANETs (iMANETs) and Multihop Mobile Networks, as well intelligent gateway selection and enforcement in iMANETs.

Multihoming

Multihoming in mobility scenarios is a highly useful concept. The use of multiple available connections to the Internet can help improve the resilience and reliability of a node's Internet connectivity as well as provide an opportunity to perform more seamless handovers. Figure 38 on page 101 illustrates a scenario in which a newly attaching MANET node (Node 3) has three available Internet access options. Two indirect connections to the Internet via existing MANETs and one direct connection via a UMTS interface. This type of communication situation could be feasibly expected to arise in many typical MANET scenarios. By leveraging the concept of Multiple Care-of-Address Registration (MCoA) within UMA, the newly attaching MANET node could make use of both of the available in-direct connections to the Internet as well as establish its own direct connection via its UMTS interface to register three simultaneous bindings with its HA (HA3).

Using this approach the MANET node is able to register simultaneous locations that it is reachable at with its HA. Once registered, the MANET node and the HA can then choose which connection to transmit packets via based on policy or connectivity quality. In addition, this approach also would also enable MANET nodes to perform near instantaneous handovers since parallel layer 3 connections can be established and then switched between as and when required, resulting in almost no disruption.

Authentication, Authorisation and Accounting (AAA)

The ability to efficiently and accurately perform Authentication, Authorisation and Accounting (AAA) is a fundamental component of most successful networking solutions.

Performing effective AAA in Mobile Ad-hoc networks is inherently difficult because of the infrastructureless nature of ad-hoc networks. UMA has been designed in a manner which attempts to provide a potential solution to these AAA considerations by leveraging the structured approach of the Inter-HA communication process imposed by the UMA protocol.

This process ensures that there is always static entity available that is directly associated with any MANET node (i.e. the HA). As the HA is always involved in the communication process it is therefore constantly available to authenticate the MANET Node and can be subsequently billed for the nodes service usage if necessary. Accountability is important because if we consider a typical Mobile Ad-hoc Networking scenario whereby nodes in the MANET wish to communicate externally with nodes in the Internet, the Gateway nodes are required to perform an unfair role in the overall communication model. This is because the Gateway nodes will be required to carry the traffic of other nodes in the MANET as well as its own. Arbitrarily requiring nodes to perform this function may be infeasible in certain scenarios, especially if the Gateway node accesses the Internet via a potentially resource constrained or financially expensive access medium.

In these scenarios, the Gateway node will suffer degradation in their its service and possibly incur additional costs. The Inter-HA communication system employed by UMA ensures that the HAs of Gateways are potentially able to maintain accurate records of which other MANET nodes have used the Gateways Internet connection and how much traffic they have transmitted in total. At present our implementation only performs basic Access and Authentication checks, but it is our intention to integrate a comprehensive AAA solution into the UMA model in the future, in order to demonstrate the potential benefits available through using this approach.

Intelligent Gateway Selection and Enforcement in iMANETs

Consider a scenario consisting of numerous Mobile Routers (MRs) where each MR can connect both to the Internet and to each other. A very simple example of this type of scenario is presented in Figure 41:

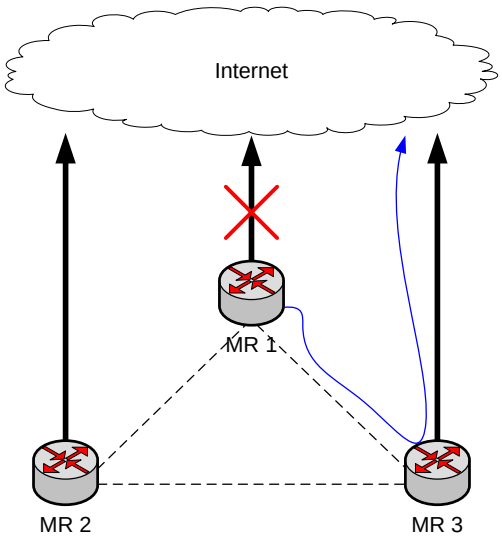

Figure 41 : Gateway Selection Problem (Simplistic)

In this illustration each MR has at first established a connection to the Internet and a connection to the other MRs around it, but then subsequently MR 1 has lost its own direct Internet connection and must seek an alternative method of communicating with nodes in the Internet. In this situation MR 1 has two

available options, it can now either route its packets via MR 2 or MR 3 because these MRs both still have their Internet connections in place. These MRs are then known as "Gateways" because they provide other MRs with a means of connecting to the Internet indirectly through them. The important question then is: Which Gateway should MR 1 choose to connect to the Internet via? Does MR 2 have a link with higher throughput capabilities? Or does MR 3 have a more reliable connection which ultimately may be more long lived? Do either of the Gateways have a cost associated with using their Internet access, if so, what is it?

At present our solution, the Unified MANEMO Architecture (UMA) can support the routing model imposed once this selection is made, but it cannot intelligently make and enforce the actual Gateway selection process itself. At the moment UMA will just blindly send packets addressed for the Internet, which are then collectively routed through the ad hoc network and out a Gateway, which Gateway that is we currently have no control over.

In our opinion researching and solving this problem will have three main phases:

1. Development of a framework for associating static values with Internet connection links that can then be expressed to other MRs considering utilising a Gateway
    - i.e. Wi-Fi link = 1, UMTS link = 2, Satellite link = 3
2. Implementation of the ability for an MR to select the use of a specific Gateway and crucially, ensure that its use is enforced throughout the network. The key thing here is to ensure that the Gateway that an MR has selected is actually used and that packets aren't subsequently forwarded by the routing protocol via a different Gateway.
3. Once phases 1 and 2 are complete we can then start to consider how to solve this problem whilst supporting dynamic values for the Internet connections of Gateways. At this point we would start to try to incorporate dynamic considerations such as the purpose of the network itself (do the requirements of the MRs as a whole change at some point in the network's lifetime, if so how does that change link selection?). Do the costs involved change at different times? Does contention for the link increase and decrease? Etc.

In order to solve this problem comprehensively we then have to start considering the bigger picture. Figure 42 illustrates a more typical topology that could be expected to arise where multiple different Gateways that are different distances away are available to a MR.

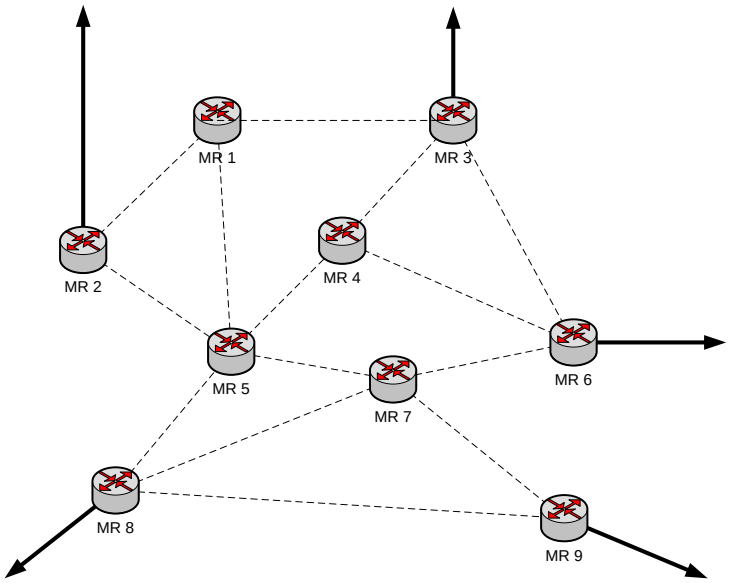

Figure 42 : Gateway Selection Problem (Complex)

Ultimately to support this kind of situation we would expect each MR to maintain a table of available Gateways available to it. At any point in time every MR should record information about every Gateway it has access to. In particular, we would expect that the MR would maintain information about the "Internal" link characteristics between itself and each of the Gateways it can use (things such as the distance the Gateway is from the MR in hops, the quality of those links, etc). Then also the "External" link values i.e. the metrics associated with a Gateway's Internet connection. Figure 43 shows a possible example of such a table for MR 4 in the situation depicted in Figure 42.

Gateway	Internal Link Values	External Link Values
MR 2	1 Hop | High Link Quality	Wi-Fi
MR 3	2 Hops | Medium Link Quality	Satellite
MR 6	1 Hop | High Link Quality	UMTS
MR 8	2 Hops | High Link Quality	UMTS
MR 9	3 Hops | Low Link Quality	Wi-Fi

Figure 43 : Gateway Option Table

References

[1] U-2010 Deliverable 1.1.1, "Reference scenarios based on user studies", October 2007.

[2] U-2010 Deliverable 1.1.2, "Functional requirements for networks and services", March 2007.

[3] U-2010 Deliverable 2.1.2, "u-2010 Architecture", June 2009.

[4] U-2010 Deliverable 2.2.2, "Report on u-2010 Mobility Solution", February 2009.

[5] U-2010 Deliverable 3.2.1, "Report on the Presence Management Solution", November 2008.

[6] U-2010 Deliverable 3.2.2, "Prototype of the Presence Management Solution", November 2008.

[7] U-2010 Deliverable 4.1.1, "Prototype of an alarm and emergency communication system based on the developed architecture and services in Luxembourg", November 2008.

[8] U-2010 Deliverable 4.2.1, "Report on the Mountain Rescue Service Concept", July 2008.

[9] U-2010 Deliverable 4.2.2, "Prototype Mountain Rescue Service Trial", September 2009.

[10] U-2010 Deliverable 4.5.1, "Concept of Prototype of Mountain rescue service implemented in Slovenia", September 2009.

[11] U-2010 Deliverable 4.5.2, "Report on Mountain Rescue Service Implemented in Slovenia", September 2009.

[12] Cockermouth Mountain Rescue FAQ, Available From: http://www.cockermouthmrt.org.uk/faq.aspx#faq4

[13] http://www.speedtest.net/

[14] http://www.broadbandspeedchecker.co.uk/

[15] http://www.thinkbroadband.com/speedtest.html

[16] Hurricane Electric Internet Services IPv6 Tunnel Broker, http://ipv6tb.he.net/

[17] Panasonic BL-C101, BL-C121 Product Brochure.

[18] Ubiquiti Routerstation homepage http://www.ubnt.com/products/rs.php

[19] Ubiquiti Routerstation Pro homepage http://www.ubnt.com/products/rspro.php

[20] "Grayrigg Train Crash", http://news.bbc.co.uk/2/hi/uk_news/6391633.stm

Acronyms

2D	2-Dimensional
3D	3-Dimensional
3G	3rd Generation (of mobile phone technology and standards)
ANSI	American National Standards Institute
AP	Access Point
API	Application Programming Interface
AR	Access Router
CaC	Command and Control interface
CANLMAN	Cumbria And North Lancashire Metropolitan Area Network
GIS	Geographical Information System
CLEO	Cumbria and Lancashire Education Online
CMRT	Cockermouth Mountain Rescue Team
CoTS	Commercial off-the-Shelf
CODEC	Coder Decoder
DC	Direct Current
EDGE	Enhanced Data Rate for Global Evolution
ESSID	Extended Service Set Identifier
EU	European Union
GCC	GNU Compiler Collection
GPRS	General Packet Radio System
GPS	Global Positioning System
GSM	Groupe Spécial Mobile (Global System for Mobile Communications)
GUI	Graphical User Interface
HA	Home Agent
HCI	Human Computer Interface
HD	High Definition
HQ	Headquarters
HSDPA	High Speed Downlink Packet Access
HTTP	Hypertext Transfer Protocol
IAN	Incident Area Network
ICT	Information Communication Technology

ID	Identifier
IETF	Internet Engineering Task Force
IOS	Internetwork Operating System
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
JPEG	Joint Photographic Experts Group
LoS	Line of Sight
MAC	Medium Access Control
MANEMO	MANET for NEMO (alternative: MANET and NEMO)
MANET	Mobile Ad-hoc Network
MCM	MANET-Centric MANEMO
MIPv6	Mobile IPv6
MP3	MPEG-1 Audio Layer 3
MPEG	Motion Picture Experts Group
MJPEG	Motion JPEG
MR	Mobile Router
NCM	NEMO-Centric MANEMO
NEMO	Network Mobility
NEMO BS	Network Mobility Basic Support
NEPL	NEMO Platform for Linux
NINA	Network in Node Advertisement
OS	Operating System
OS	Ordnance Survey
PAN	Personal Area Network
PC	Personal Computer
PDA	Personal Digital Assistant
PMS	Presence Management Service
PoE	Power over Ethernet
PoP	Point of Presence
PMS	Presence Management System
PSTN	Public Switched Telephone Network
RF	Radio Frequency
RO	Route Optimisation

RSSI	Received Signal Strength Indicator
RTT	Round Trip Time
SAR	Search and Rescue
SBC	Single Board Computer
SDIO	Secure Digital Input Output
SDK	Software Development Kit
SIP	Session Initiation Protocol
SMRA	Slovenian Mountain Rescue Association
SMS	Short Message Service
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SSL	Secure Sockets Layer
SSID	Service Set Identifier
STA	Search Theory Automatisation
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UK	United Kingdom
UMF	Unified Message Format
UMA	Unified MANEMO Architecture
UMTS	Universal Mobile Telecommunications System
UTP	Unshielded Twisted Pair
USB	Universal Serial Bus
VAR	Voice Activity Detection
VBR	Variable Bit Rate
VoIP	Voice over IP
WAN	Wide Area Network
Wi-Fi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WPA	Wi-Fi Protected Access
WPA-PSK	WPA Pre-Shared Key
XML	Extensible Markup Language