A Simple Voice Communication System For Human-In-The Loop Air Traffic Control Simulations

Florian Hafner, Mohamed Mahmoud

Center for Applied Air Traffic Management Research

Embry-Riddle Aeronautical University

Daytona Beach, Florida

KEYWORDS

Human-in-the-loop, ATC Real-Time Simulation, VoIP, Voice Communication System, UDP multicast

ABSTRACT

This paper presents the work preformed by the Embry-Riddle Aeronautical University Center for Applied Air Traffic Management Research (CAAR) in developing a Voice Communication System (VCS) to support a human-in-the-loop real-time simulation facility. The VCS was developed for use in an air-traffic control (ATC) simulation mimicking the actual VHF communication system used by air-traffic controllers, but using software-based VoIP technology within a UDP broadcast based peer-to-peer architecture. Not only does the VCS system facilitate voice communication between aircraft pseudo-pilots, it also allows for direct controller-controller voice communication. The VCS was originally implemented and tested on a LAN based ATC simulation platform but has been demonstrated as a viable way to connect widely distributed sites including real-time links to live aircraft over SATCOM, all with no appreciable delays [2]. Some additional features have been implemented to comply with the FAA Next Generation Air-Ground Communication VDL3 standard to reduce Controller/Pilot workload including Step-on prevention and controller pre-emption in addition to automated change of frequency from one controller authority to the next.

Tests on the delay of voice communication using the VCS over a 100 Mb/s LAN have shown a latency of less than 550ms. When used with additional features such as step-on prevention, this delay is entirely acceptable for this type of application [4].

INTRODUCTION

The Embry-Riddle Center for Applied Air Traffic Management Research (CAAR) has an in-house designed and built object-oriented Real-Time Distributed ATM Simulator (RTDS). This simulator is used to test new concepts or simulate specific scenarios in ATM. The design concept behind the RTDS is to use a User Datagram Protocol (UDP) Multicast messaging system which mirrors the real world object behavior in terms of widely distributed sites communicating with aircraft. This broadcast approach allows the addition of new components without adding to the communication load or bandwidth; thus, new controller or pseudo-pilot positions can be added as needed, making the system highly scalable and flexible.

Having a simulator of this size requires an efficient voice communication capability between controllers and pseudo-pilots. Originally, a system similar to a Public Switched Telephone Network (PSTN) had been developed and installed that provided basic analog voice communication. However, it was not expandable, nor was it simulating radio transmissions in a realistic fashion.

Therefore, after investigating different voice communication concepts, an implementation utilizing digital VoIP technology was developed to compliment the RTDS. The VCS had to be light-weight, easy to implement and potentially had to run on multiple platforms at widely distributed locations. These requirements suggested the use of Java, since it is a multi-platform capable language and has high native support for multi-media applications. Even though Java is widely considered to be slower than equivalent C/C++ applications, test results from this research have shown that the levels of delay in processed voice data are acceptable and that the voice quality is comparably high.

VCS BACKGROUND

Traditionally, most VoIP applications were based on the client-server architecture layout depicted in *Figure 1* where the system is composed of server and client applications. Clients rely on servers to route sound packets between clients. Despite a number of benefits such as a high level of control over individual communication channels, this model has the disadvantage of a single point of failure: if the server fails, the whole system ceases to be operational. It also means that if additional clients are added to the system, the server incurs additional workload by having to route additional sound packets.

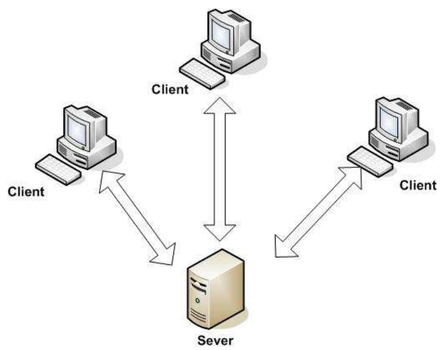

Figure 1: Client-Server layout

Another network communication model of interest for VoIP communication uses a peer-to-peer architecture model. This type of architecture provides several advantages over client-server, which strongly suggested the use for a distributed VoIP communications system:

- Expandable over any multicast enabled network
- Content and resources can be shared from the center and the edge of the network
- A network of peers is easily and almost limitlessly scaled and more reliable than a single server
- Reduced network traffic as VCS uses UDP multicast for communication
- Fault tolerance, in case of failure of one or more peers the system will still be operational

An architectural overview of network communication in a peer-to-peer infrastructure is presented below in *Figure 2*: To implement this peer-to-peer VoIP infrastructure, the UDP protocol using multicasting was utilized.

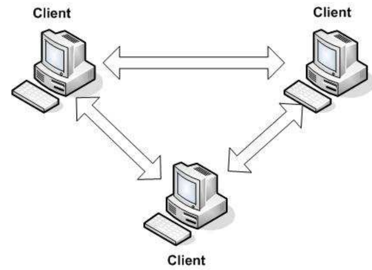

Figure 2: Peer-to-peer layout

UDP is preferred in this application as it is stateless, doesn't require the retransmission of dropped packets and has smaller headers, which increases the amount of audio data in each network packet.

VOIP VERSUS VHF

Currently air traffic controllers are using communication equipment that has not significantly advanced in the last 50 years, and which still has the same restrictions on communications capabilities. As aircraft cockpits are becoming revolutionized with the latest technology, communications between pilot and controllers remain the same. While flying in line of sight of land, pilots communicate with controllers using Very High Frequency (VHF) radios. The number of VHF frequencies available is barely sufficient for the number of sectors. This has led Europe to reduce the separation between frequencies in use, at considerable expense to air traffic service providers and aircraft operators.

Transoceanic flights rely upon High Frequency (HF) radio communications. In-flight pilots contact Oceanic Air Traffic Control Centers (OATCC) via HF radio frequencies. These pilots are not always communicating directly with the controllers. Frequently, the requests and transmissions are relayed through Aeronautical Radio, Inc. or international flight service station, finally arriving to the OATCC. Utilizing a relay station causes delays in transmissions.

While controllers are relying upon out-dated equipment to provide guidance to aircraft in the skies, businesses and individuals are increasingly having 'telephone' conversations and conferences utilizing voice over Internet Protocol (VoIP). A solution to the limited number of radio frequencies, crowded frequencies, utilizing relay stations and many other limitations, would be to transition air traffic communications to VoIP technology. This technology would enable pilots and controllers to communicate via voice over data-link that could be carried on VHF Data-Link (VDL) or over satellite links. VoIP could be a solution to the lack of VHF radio frequencies while decreasing controller workload.

Also, air traffic controllers in some sectors experience high workloads due to the increase in air traffic movements. That high workload can be attributed to the handover/takeover process and the associated radio frequency changes and check-in. A transition from analog VHF to digital VoIP communication would provide the ability to integrate voice communication with other control functions, potentially allowing the automatic switching of '*frequencies*' – or VoIP channels – when aircraft are handed from one control facility to the next.

VCS ARCHITECTURE

The architecture model chosen for VCS matches the RTDS in that it uses multicast peer-to peer communication using the UDP protocol, which allows two or more clients to exchange data with each other without a central server. Unlike a client-server architecture where data is routed through servers and replicated for each interested client, a multi-cast peer-to-peer network is uniquely decentralized and allows computers to communicate directly with each other and share data without replication. *Figure 2* previously and *Figure 3* below depict the information flow based on this architecture specifically adapted to pilot-pilot and controller-pilot communication in a human-in-the-loop Air Traffic Control focused simulation.

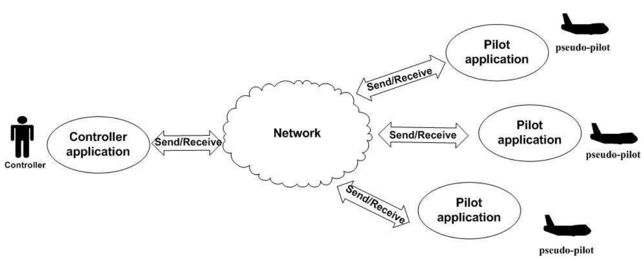

Figure 3: VCS Context Diagram

The controller-pilot communication structure in the above diagram may be replicated to include as many controller and pilot applications as required for the scenario. The controller and pilot applications are separate processes that are selectively executed based on the scenario requirements. Each application in itself has different components and threads.

Components Behavior

The core VCS application process is composed of a number of components which are present in both controller and pilot executables. *Figure 4* below shows a conceptual component view of the VCS architecture with its internal structured components and their interactions with the execution platform including the sounds card and the network devices:

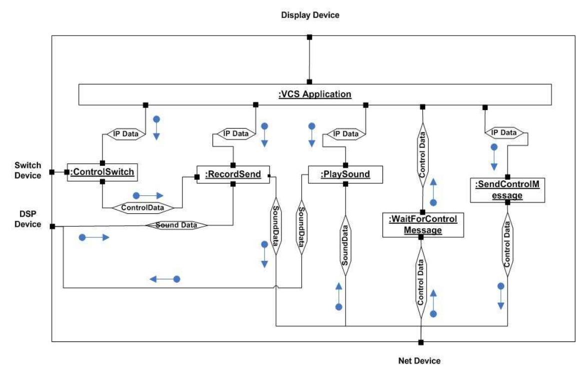

Figure 4: VCS Conceptual View Diagram

The following is description of each component:

- <u>VCSApplication</u>:
  Starts VCS GUI and initializes/establishes communication with the system devices
- <u>RecordSend</u>:
  Interfaces with the Digital Sound Processing (DSP) device, capturing the audio data and sending it over the Network device.
- <u>PlaySound</u>:
  Receives sound data over the Multicast voice channel from the network device. Upon receiving the sound data, it is sent it to the DSP device for playback
- <u>WaitForControlMessage</u>
  Receives control data messages over the Multicast data channel through the network device, including commands to change channels or to 'mute' to allow controllers to override pilots
- <u>SendControlMessage:</u>
  Sends control data to the Multicast data channel

From the core VCS architecture described above, controller and pilot specific derivatives are generated, which mainly differ in the Graphical User Interfaces (GUIs) and in the control messages they process.

Controller VCS

The Controller Application provides voice communication between the controller and the pseudo-pilots in the sector being controlled by the air traffic controller. Each sector has a unique multicast IP address and a port number that replicates the controller radio frequency and allows all controllers and pilots in the sector to communicate with each other as they would with conventional VHF radio utilizing a push to talk switch.

In Addition to the Multicast voice channel the VCS uses to send audio packets between controllers and pilots, a second multicast data channel is established to provide for the dissemination of control messages for the GUI application. This control channel also provides for the means of sending text messages between the controller and pseudo-pilot controlling aircraft in his sector, which is commonly referred to as "Controller Pilot Data Link Communication" or CPDLC. *Figure 5* below provides a diagrammatic overview of the multicast channel setup and communication.

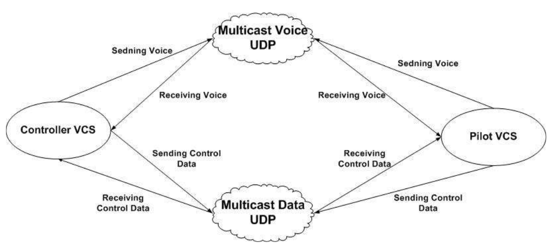

Figure 5: Multicast UPD Layout

The use of CPDLC is a novel idea currently being evaluated for use in real-world ATC applications with the intention of reducing VHF frequency usage and reducing the number of misunderstandings of instructions between controllers and pilots.

Additionally, the VCS controller application provides for communication between two controller entities. The need for this arises fairly infrequently, but does need to be provided for in case a multiple controller simulation scenario requires controller-controller negotiation. Each controller listens to yet another unique multicast IP address and port number for communicating person-to-person with any other controller in the simulation. This communication channel can be established through the graphical user interface (GUI) running on a touch sensitive screen, similar to the Voice Switch Control System of the Display System Replacement (DSR) application used in En-Route Air Traffic Control Centers throughout the United States. This GUI allows the controller to initiate or accept a point-to-point "telephone" communication with other controllers. *Figure 6* shows the touch screen GUI application for controller RIDDLEWEST (see title bar of GUI) with one other sector – or controller – RIDDLEEAST logged on to the VCS.

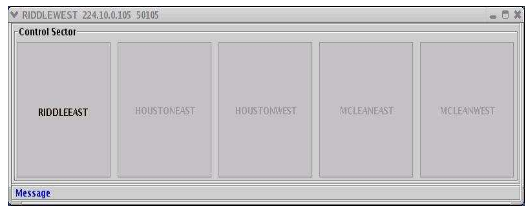

Figure 6: Controller GUI

By simply pressing the GUI button of the intended recipient sector controller, the current controller in sector RIDDLEWEST can request a point-to-point conversation with another controller. This causes the button to flash in blue and yellow repeatedly until the communication is accepted by the recipient controller (see *Figure 7*).

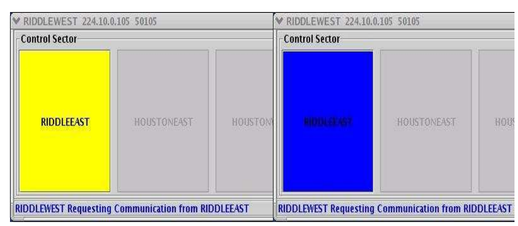

Figure 7: Controller communication with other sector

The controller with whom a point-to-point conversation is being initiated will see the button with the calling controller sector's name flash in red and blue repeatedly, until the incoming call is accepted by pressing the flashing button on the GUI. After the call is accepted, the button of the calling controller will remain solid blue.

Once this controller-controller communication channel has been established, no aircraft being controlled by either controller hear this audio communication. The controller-controller voice communication is essentially done using an open mike (the application continuously transmits and receives audio data between the controllers without the need to depress the push-to-talk switch). While the phone feature is in use, the controllers can still talk to aircraft on their channels by depressing the push-to-talk switch.

Pilot VCS

The pilot VCS application provides a GUI that lists all sectors available for the pilot to select from. The pilot essentially selects the sector for which they would like to provide pseudo-pilot services. It is possible for multiple pseudo-pilots to subscribe to the same sector. This usually occurs when the number of aircraft in the particular sector causes pseudo-pilot workload large enough to warrant multiple pilots. *Figure 8* shows the pilot GUI with two sectors, RIDDLEAST and RIDDLEWEST available to the pilot to initiate communication with. Once the pilot selects a specific sector for communication, the pilot will hear the controller as well as the other pilots in that sector, analogous to selecting a sector VHF frequency. The selected sector will be highlighted in red.

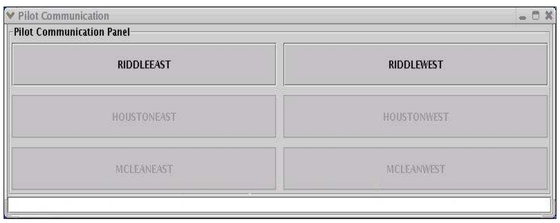

Figure 8: Pilot GUI

The application has two modes of operation, one that allows a pseudo-pilot to be logged on to voice communication for multiple aircraft simultaneously without providing for hand-off capabilities on the pseudo-pilot side, and the other by using a designated flight number, which essentially makes a single pseudo-pilot responsible for a single flight through all control sectors. Figure 9 below shows the pseudo-pilot GUI when logged in for a single aircraft only; in this case the aircraft callsign is CBB1 (see title bar).

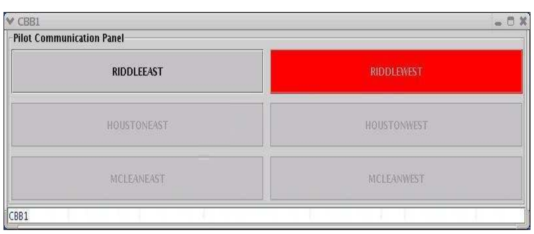

Figure 9: Pilot in Communication

If a pilot is logged-on as an aircraft with a specific flight number as shown above with aircraft CBB1, when the flight is handed over to a new sector the pilot VCS application will automatically retune to the new sector frequency when the hand-off procedure between sectors is completed. The pilot may override the frequency change by selecting a different sector from the GUI.

VCS implementation

The EARL VCS was implemented using the Java core sound API [6] to maintain simplicity, rapid development, and platform independence. The use of UDP multicasting as a network protocol to propagate messages around an entire LAN allows for a very flexible and scalable system. This satisfied an early requirement that the VCS be extendable to provide for pseudo-pilot capabilities for many different types of ATC simulation scenarios with varied numbers of aircraft in the system.

VCS ADVANCED FEATURES

In addition to providing basic voice communication between controllers and pilots, some additional features have been implemented which were aimed at reducing controller workload and the effects of transmission latency caused by the software, the use of a codec, and network communication. These additional features include controller pre-emption, step-on prevention, and a close integration of the VCS with the actual RTDS ATC simulator.

Integration with ATM System

One of the most important features of the VCS system is its capability to interface with the RTDS Human-in-the-Loop simulation platform. The VCS RTDS interface application "listens" to the messages from both the VCS process and the RTDS. Figure 10 shows the flow of data to and from the RTDS VCS interface application that allow the integration of VCS with any real-time simulation platform.

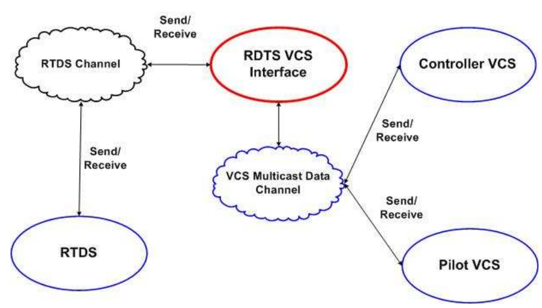

Figure 10: RTDS VCS Interface diagram

For example, when a controller assumes control of an aircraft within the RTDS simulation by clicking on the position symbol or entering the computer ID (CID), the display issues as part of the message an “assume control” message to provide aircraft ID, the handing-over sector and receiving sector. The VCS RTDS Interface process then sends a message to the aircraft VCS application to switch to the multicast IP of the assuming sector. No extra action is required and the transfer of communications is automatic. Pseudo pilots may override automated transfer by selecting a particular sector on the VCS touch-panel GUI.

It should also be of interest to note that even though the focus in this paper has been on UDP multicast messaging for the VCS, a simple interface can be developed to provide for means to communicate with other simulators using TCP/IP messaging.

Controller pre-emption

A common real-world problem with the current VHF communication system is that an aircraft can unintentionally monopolize a VHF frequency. This may be caused due to equipment failures such as a ‘stuck’ transmit switch. This may cause continuous transmission from a single pilot, thereby limiting the ability of other voice communication on this frequency.

The VCS controller application provides controller pre-emption which allows the controller to break into any transmission by a pseudo-pilot at anytime, basically giving the sector controller the capability to interrupt any pilot who is currently transmitting on the multicast channel. This is achieved by sending a mute message to all pseudo-pilots in that specific sector, thereby preventing them from sending their sound packets while the controller is sending a voice transmission.

Step-on Prevention

A major problems with current conventional radios is two aircraft are allows to transmit at the same time, which is known as ‘step-on’. During simultaneous transmission of two aircraft, all that is received by the controller is a ‘squeal’, causing the controller to have to request a repeat of pilot requests or transmissions. This increases pilot and controller workload to an unnecessary degree.

To prevent step-on occurrences within the VCS, whenever the controller application receives a voice transmission from a pilot application, it immediately issues a mute command to all other aircraft applications within the controller sector, thereby preventing them from sending voice message until the transmitting pseudo-pilot has ended their voice transmission. The controller application essentially acts as a server controller for this scenario and gives precedence to the pseudo-pilot voice transmission received first. This functionality prevents more than one pilot from transmitting at any give time.

COMPRESSION

The initial VCS release did not use an audio codec as bandwidth was not deemed a constraint. However, as later results would show, the size of voice packets being transmitted would nevertheless play a role in the latency of the communication.

The initial requirement for VCS was to provide voice communication support for the RTDS simulator platform used by ERAU-CAAR on a 100 Mbits/sec LAN. Even though the VCS voice quality and total performance was deemed very acceptable, the bandwidth required by the application was higher then desired. Initial testing of the VCS without a codec showed that during a one minute voice communication between a controller and a pilot, the transmission rate was 13.2 Kbytes/sec. After implementing the GSM6.10 codec [5] the transmission rate was reduce to 3.8 Kbytes/sec, which lead to a 71 % reduction in bandwidth used by the VCS application. As test results will show later in this paper, this reduction in bandwidth also generated a reduction in the latency observed in the communication.

HARDWARE INTERFACE

The VCS uses standard PCs equipped with industry standard “Sound Blaster Live” Sound cards and stereo headset with a microphone connected to the sound cards.

A press-to-transmit foot/handheld switch connected to either serial/parallel the port to initiate sound recording and transmission.

Touch panel displays were used to run the controller and pilot VCS applications. This setup was intended to mimic the Display System Replacement (DSR) voice communication system. Setup currently in place in many En-Route Air Traffic Control Centers in the US.

TESTING METHODOLOGY

Perhaps the most important factor in a controller and pilot voice or any VoIP communication is the quality of the service ---- known as QoS. ITU-T Recommendation E.800 defines QoS as *"The collective effect of service performance, which determines the degree of satisfaction of a user of the service."* A number of different factors affect the QoS, including packet loss, delay, available bandwidth, network protocol used and jitter.

The VCS application was tested using two different approaches. A subjective test evaluated the quality of the VCS from a user prospective and an objective test which measured the performance of the application in terms of software metrics, such as delay and packet loss.

Subjective rating

The subjective testing of the VCS system was based on the measurement techniques that are defined in ITU-T P.800. These techniques are largely based on the feedback and opinion of pseudo-pilot and controller test subjects who observed the voice communication and rated the quality of the transmission on a 1 to 5 scale with 5 being excellent and 1 unsatisfactory. In doing so, they were briefed about what factors they should consider. These factors included audio loss, circuit noise, sidetones, talker echo, distortion, delay, and other transmission problems.

Objective rating

The most important factor when objectively testing the VCS is the measurement of the latency or delay since this increases the difficulty to have a normal conversation (and therefore workload). The ITU-T standard recommends a maximum of 0.15 seconds unidirectional delay. Specific sources of delay can be identified by examining the routes that audio packets travel from source to destination. Looking at a generic LAN network architecture, the delay budget contains the following [3]:

- Signal encoding/decoding algorithm, typically 15-37.5 milliseconds at both the origin and destination.
- Protocol processing overheads in components to include UDP, and IP information plus echo cancellation, typically less than 5 milliseconds.
- Bandwidth and utilization (loading) of the LAN and WAN channels, which may introduce framing and queuing delays in the range of 5-25 milliseconds, depending on the transmission rate.
- Routing, queuing, and propagation delays across the WAN which depend on transmission media and distance, typically 10-40 milliseconds.
- Jitter, or variation in the arrival rates of the packets comprising the audio or video stream. Since the packetized voice samples may take different paths through the packet-switched network, arrival rates of those packets.

Testing Setup

The VCS[1] was deployed and tested within Embry-Riddle's ATM Research laboratory and the Global Communication, Navigation, and Surveillance Systems (GCNSS) project sponsored by the FAA and Boeing. The project was intended to demonstrate the feasibility of a global satellite-based CNS, using a highly integrated Common Information Network (CIN) architecture. The demonstration of these new concepts involved a single aircraft flying in and out of radar and VHF coverage over the Gulf of Mexico, while maintaining full communication and control using Satellite communications for both position reports and voice communication.

To support the demonstrations, the CAAR RTDS real time simulation including the VCS was widely distributed, from Daytona Beach, FL, with 2 simulator positions each in McLean, VA; Houston, TX; Irvine CA; and in the Boeing 737-400 experimental aircraft flying in the demonstrations. The network architecture was comprised of a mix of dedicated T1 based WAN and SATCOM over Ku geostationary and L band (Iridium) links. *Figure 10* shows the network layout for the GCNSS project. Throughout the demonstration, the VCS provided a full 'party-line' capability with step-on prevention and controller pre-emption. Controllers at Houston were able to transfer voice communications as part of a standard 'silent handover' between two sectors/controllers.

[1] GSM6.10 codec was not available during GCNSS

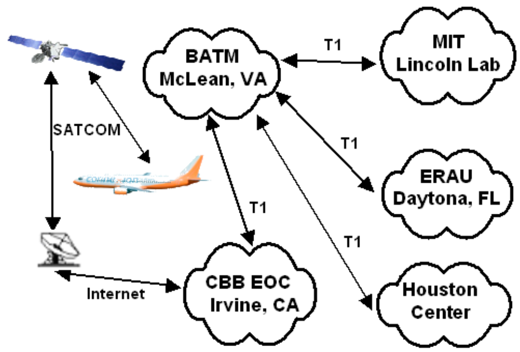

Figure 11: GCNSS Network Layout

The measurement for communication latency within the VCS, was accomplished using the process described in *Figure 11*.

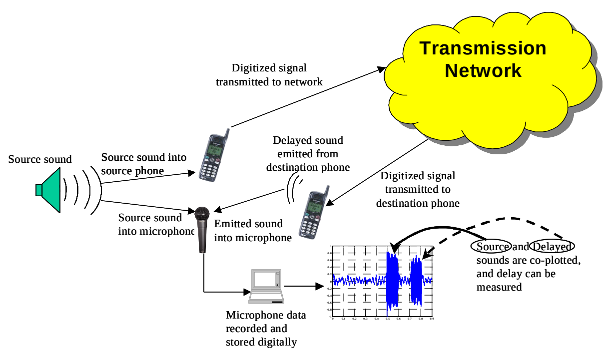

Figure 12: VCS delay measurement setup

Using the above process, a short sound is played into both the source headset microphone and the measurement system (an oscilloscope), which is recording all sounds. In this case, the sound was a sinusoidal tone at 1000 Hz lasting for 100 milliseconds. The sound is digitized by the source headset, encoded, and transmitted over the network, where it is received by the destination hardware, decoded, and played through the headset's speaker(s). This output version of the test sound is also played into the measurement system, which is continuously recording. The test tone was repeated five times for each measurement, with a spacing of approximately 1 second, and recording was then stopped. Using these measurements, the elapsed times between the origination and playback of the tones were recorded [2].

EXPERIMENTAL RESULTS

Subjective results

Voice quality tests were run using experienced controllers from Houston En-Route ATC center, and pilots from Embry-Riddle Aeronautical University (ERAU). The tests included two controllers and eight pilots. These tests essentially evaluated the controllers' abilities to interact with and control pseudo-piloted aircraft using the VCS and any associated delay and voice quality issues.

The test was repeated four times. Keeping in mind that the test subjects are humans, some expected variation in the average final ratings was observed. However, the overall Mean Opinion Scores rating for the VCS was 4 (Good) [3]. Additional comments by the air traffic controllers also re-iterated that the delay inherent in the VCS system had very little effect on their ability to control aircraft.

Objective results

Using the measurement procedure outline earlier, voice latency tests were run for the GCNSS demonstrations, as well as at ERAU after some enhancements to the VCS software.

The voice latency tests performed during the GCNSS project revealed an average latency of 661 milliseconds. This basically means that each voice communication packet reached the recipients headphones approximately 660ms after being transmitted through the microphone.

Following some enhancements to the VCS aimed at improving performance, the same type of process used to evaluate the VCS previously was employed. Test results revealed a mean latency of 536ms with a standard deviation of 11ms using the GSM codec. Interestingly, using a PCM codec generated a mean latency of 550ms and equal standard deviation.

BENEFITS OF VOIP INTEGRATION WITH REAL-TIME SIMULATOR

Any human-in-the-loop simulation is likely to require some form of voice communication between subjects and/or supporting staff. Even though it is sometimes convenient to purchase an off-the-shelf system, the benefits of Voice over IP have shown clear merit in its ability to use the existing network infrastructure and provide high quality sound. Since VoIP data as well as real-time human-in-the-loop data and control messages are both transmitted over the same network, it becomes convenient to link both systems together (this is in fact what select voice recognition software accomplishes on some high-end simulators).

The integration of the real-time simulation and VCS systems creates a setup that reduces workload, and provides dynamic scalability to add/subtract users on the system. It also allows subjects in the human-in-the-loop simulation to manipulate frequencies and channels through the real-time simulation interface without having to manually perform these tasks on the VoIP system. Also, if the need arises to widely distribute the real-time simulation across multiple locations, the VCS requires no modifications to accommodate this.

CONCLUSION

This study has shown that a relatively simple, scalable, and flexible Voice Communication System can be developed fairly rapidly using Java and in particular it's multimedia components.

The VCS provided a high fidelity voice communication system to support human-in-the-loop simulation scenarios by utilizing VoIP technology, mimicking the actual VHF used in controller/pilot communication for use in air-traffic control (ATC) simulation. The VCS was fully compliant with the VDL3 [4] requirements by providing voice and data link with added features such as controller pre-emption, step-on prevention and transmit status indicator.

Subjective metrics used to assess the Quality of Service of the voice communication system have shown a ranking of 4, indicating overall 'good' performance. An analysis of voice latency over a standard LAN resulted in a delay of 536ms. While this may seem significant, Sollenberger et. al. have demonstrated that latency greater than 350ms and even up to 750ms is acceptable in ATC simulations providing that controller pre-emption and step-on prevention mechanisms are used [4]. Consequently, the VCS has shown reasonable performance for use in these types of human-in-the-loop simulations.

The system was designed using a pure object oriented "OO" methodology with a peer-to-peer architecture and UDP multicast communications, which add flexibility and scalability to the system.

The simplicity of the VCS system allows it to be used in a wide variety of simulation scenarios; for example the VCS can be changed to simulate airline dispatcher communication with their own aircraft or air-traffic ground control movement.

FUTURE WORK

Current efforts are focused on extending the VCS to use the IEEE 1278.1 DIS standard [9] for message formats, which will provide the ability to interface with other voice communications systems. The VCS is also being modified to allow for CPDLC text messaging between controllers and pilots. Work is also continually on-going to improve the performance of the VCS and reduce voice latency.

REFERENCES

[1] Federal Aviation Administration (2000). *Air traffic control* [7110.65M]. Washington, DC: Author. Hopkin,V.D. (1995).*Human factors in air traffic control.* Bristol, PA: Taylor & Francis.

[2] Boeing Air Traffic Management, 2004, "GCNSS Demonstration Segment B Report", Document number: D794-10141-1, pg. 3.

[3] ITU-T Recommendation G.114, May, 2003.

[4] Sollenberger, R., McAnulty, D., Kerns, K. "The Effect of Voice Communications Latency in High Density, Communications Intensive Airspace", Technical Note DOT/FAA/CT/TN03/04, January 2003

[5] GSM06.10 speech compression, At: http://kbs.cs.tu-berlin.de/~jutta/toast.html

[6] Java Sound API, At: http://java.sun.com/j2se/1.4.1/docs/api

[7] Java Sound Resources, At: http://www.jsresources.org/faq_audio.html

[8] A comparison of internet audio compression, At: http://www.sericyb.com.au/audio.html

[9] IEEE. *Standard for Distributed Interactive Simulation*. IEEE Std 1278.1-1995

BIOGRAPHY

Mohamed Mahmoud received his Bachelor of Science in Aerospace Engineering and Masters in Software Engineering from Embry-Riddle Aeronautical University in Daytona Beach, Florida in 1997 and 2002, respectively. While completing his graduate degree, Mr. Mahmoud worked for Carrier *Electronics*, Farmington, CT for one year developing front end user interface software for controlling AC units. Upon returning back to ERAU, Mr. Mohamed worked as a graduate student assistant teaching the Introduction to Software Engineering course. Later, Mr. Mohamed joined the Air Traffic Management Research lab, as a research assistant, participating in various research projects in the field of ATM simulation. Following his graduation, Mr. Mahmoud was promoted to Software Research Engineer and tasked to investigate VoIP technology and the possibility of using it in the field of Air Traffic simulation. Mr. Mohamed developed the concept of using VoIP technology to mimic VHF communications, which was used in the design and implementation of the VCS system.

Florian Hafner received his Masters of Software Engineering degree from Embry-Riddle Aeronautical University in 2002. While completing his graduate work at ERAU, Mr. Hafner worked for the Air Traffic Management Research Lab, where he spent the majority of his time on TAAM simulation projects and software development for a real-time ATC Human-in-the-Loop simulator. Following his graduation, Mr. Hafner joined Preston Aviation Solutions in 2002 as the Customer Support Manager responsible for providing deployment and technical support for North American users of TAAM, PaxSim, and AADS. In 2003, Mr. Hafner left Preston Aviation Solutions to rejoin Embry-Riddle as a Senior Air Traffic Management Research Associate and Adjunct Professor. In his present position, he is responsible for overseeing a variety of ATM Research projects dealing primarily with fast-time and real-time simulation, modeling, and analysis, Air Traffic data mining and analysis, and the development and evaluation of novel ATM concepts and systems. As an Adjunct Professor, Mr. Hafner also instructs graduate students in the use of TAAM in support of airport design and master planning. Mr. Hafner is currently pursuing a PhD degree in Industrial Engineering at the University of Central Florida, focusing on Simulation, Modeling and Analysis.