A TELEPHONY APPLICATION FOR MANETS
Voice Over a MANET-extended JXTA Virtual Overlay Network

Luis Bernardo, Rodolfo Oliveira, Sérgio Gaspar, David Paulino and Paulo Pinto
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, P-2829-516 Caparica, Portugal

Keywords: Internet Telephony service, Mobile Ad-hoc Networks, Peer-to-Peer computing.

Abstract: This paper presents MANET-VoVON, a new Internet application for mobile ad-hoc networks (MANETs) providing voice over virtual overlay networks. A MANET-enabled version of JXTA peer-to-peer modular open platform (MANET-JXTA) is used to support user location and optionally, audio streaming over the JXTA virtual overlay network. Using MANET-JXTA, a client can search asynchronously for a user, and delay the call setup until a path is available to reach the user. The application uses a private signalling protocol based on the exchange of XML messages over MANET-JXTA communication channels. Nevertheless, it is fully interoperable with normal SIP clients through an embedded gateway function. This paper describes a prototype implementation of the proposed application and of the MANET-JXTA, and presents some performance measurements.

1 INTRODUCTION

Internet telephony services based on SIP (*Session Initiation Protocol*) (Rosenberg, 02) were designed for stable networks. The original IETF's approach relied on a registration service to handle client's mobility and availability. The call setup is forwarded by a proxy server, which queries the registration service for the current user location. Peer-to-peer (p2p) services were proposed for replacing the centralized registration service: Skype (Skype, 06) (Baset, 04) uses Kazaa's p2p infrastructure for storing the user's location, and P2P-SIP (Sing, 05) uses a Chord p2p infrastructure for the same purpose. However, both p2p approaches were designed for the Internet and assume complete and stable connectivity.

Unstable mobile ad-hoc networks (MANETs) present a challenge for the existing applications because they are composed by scattered islands of nodes, with limited connectivity. Under these conditions it is costly to create and maintain a Chord distributed hash table (DHT) with a ring topology that does not match the MANET topology. A Chord link may connect two nodes that geographically are several hops distant. It is also costly to create and maintain a virtual overlay network (VON) connecting Kazaa supernodes. (Oliveira, 05) shows that for these conditions the p2p infrastructures above collapse for high node's speeds. For a MANET it is also not acceptable to base the telephony call's success upon the connectivity to a centralized node. Instead, a new asynchronous tolerant approach is needed. The "location service" must be capable of storing the user's "call desires", and opportunistically establish the call when the network topology allows it. Therefore, the call establishment is not immediate but can take some time after the user gives the command.

We have implemented MANET-VoVON (MANET Voice over VON), a partition tolerant telephony service. The prototype is based on a new p2p architecture (Oliveira, 05b) that was implemented as an extension of the JXTA platform version 2.3.2 (JXTA, 06) optimized for unstable MANETs (we call it MANET-JXTA in this paper). This service interoperates with external SIP clients through a gateway, and within the MANET using MANET-JXTA services. It can stream audio directly end-to-end, or through the p2p VON. An evaluation of the MANET-JXTA overhead is presented in this paper.

The remainder of this paper is organized as follows: Section two introduces MANET-JXTA main characteristics. Section three presents the proposed internet telephony architecture, and its interoperation with SIP based services. Section four gives an overview of the prototype software implemented, and some performance measurements. Finally, some conclusions are drawn in section five.

Bernardo L., Oliveira R., Gaspar S., Paulino D. and Pinto P. (2006).
A TELEPHONY APPLICATION FOR MANETS - Voice Over a MANET-extended JXTA Virtual Overlay Network.
In *Proceedings of the International Conference on Wireless Information Networks and Systems*, pages 227-233

2 MANET-JXTA

MANET-JXTA defines an extended set of the JXTA core p2p protocols and services (JXTA, 04), on top of which applications can be implemented. These protocols provide the basic functionality for peer and resource discovery, communication and organization.

2.1 JXTA

Applications run on a JXTA VON, which is a set of peers grouped on peer groups created by exchanging advertisements. All entities and resources (peers, groups, pipes, endpoints, queries, services, etc.) in JXTA are represented by advertisements. Advertisements are XML documents having unique IDs. JXTA core p2p services manipulate advertisements caching them and using searching procedures when caching fails.

JXTA defines a set of protocols adapted for searching advertisements on the Internet (JXTA, 04). Basic query–reply message exchanges are supported by the Resolver Service (RS), which implements the Peer Resolver Protocol (PRP). RS is then used to implement other application specific resolution services. When no cache information is available, PRP uses a search service to look for advertisements. The most basic search service is the Rendezvous service and besides PRP other higher level search services also use it (e.g. Peer Discovery service, Pipe Binding Protocol). Additionally, JXTA supports a loosely-consistent DHT of advertisements (Rendezvous peer view service) implemented by a subset of Rendezvous peers. A Rendezvous (RV) peer is the equivalent to the Kazza's supernode: it acts as an advertisement concentrator supporting searches for a subset of non-RV peers.

Inter-peer communication in JXTA is usually implemented by exchanging XML messages over a communication channel created by the pipe service. The pipe service defines a one-to-one or one-to-many non-reliable peer-to-peer channel that transport messages through the VON connecting the peer-nodes, crossing NAT routers and firewalls. JXTA 2.0 defines two additional communication layers that are not adapted to telephony applications: endpoint service implements static connections; JXTA sockets implement reliable communication.

Pipes use logical identifiers at system level that are associated and resolved to physical addresses by the Endpoint Routing Protocol (ERP) at run-time, and when the connection is lost. Therefore, they identify univocally an application and a peer, independently of how many network interfaces the peer's machine has, or if their IP addresses change. However, pipes introduce an important message overhead that results from all the JXTA headers included by the JXTA core protocols and the application protocols. The minimum set includes the Endpoint Router header, which has the source and destination endpoint addresses, and a tentative route to the destination. (Antoniu, 05) evaluates the JXTA's communication performance on a Fast Ethernet, and shows that JXTA overhead can be very high for small packet sizes. On the other side, a minimum of five percent throughput degradation is achievable on a Fast Ethernet for large packet sizes.

2.2 MANET-JXTA

MANET-JXTA handles queries preferentially using searching, because cached information tends to be outdated too fast on MANETs. It is also capable of using the original JXTA mechanism to interoperate with pure, non MANET-JXTA peers. MANET-JXTA adds the MANET rendezvous protocol (MANET-RVP) (Oliveira, 05b), a p2p protocol that uses a cross-layering flooding approach for propagating query messages through rendezvous peers in real-time. Query messages are flooded directly into the wireless LAN (WLAN) using IP multicast. It does rely on any existing VON topology. That might not be adapted to the physical topology. Due to peer movement, several RV peers may be located on the same network region. A clustering protocol is used to reduce the flooding overhead on these conditions. Each RV peer sends beacons periodically. Using the beacon information, RV peers are capable of selecting a subset of the RV peers as broadcast group leaders, responsible for broadcasting the query messages through all RV peers. Beacons are also used to detect the appearance of new neighbours. Notice that for a MANET almost all the peers must be RV peers because the topology may change continuously.

MANET-JXTA also modifies the Resolver Service (RS), responsible for answering advertisement queries. A TTL (*Time-To-Live*) parameter was added to the query message, allowing applications to specify time limited asynchronous queries.

After the query message reaches a RV peer, it is successively flooded using MANET-RVP protocol until it reaches all connected RV peers, within the specified maximum number of network hops. The response message is sent back to the query originator reversing the query message path. If the path is broken (a peer in the path is out of range), the route resolver protocol uses RS to search for another path using the destination endpoint addresses. Besides answering to the query with local information, RV peers add the query message to a finite length active query table. When a RV peer previously out of range (peer 3 in figure 1) is detected, the queries in the table are repropagated to it, and to the RV peers in the island it belongs to, eventually being stored by all RV peers that ever get in touch with a covered peer island.

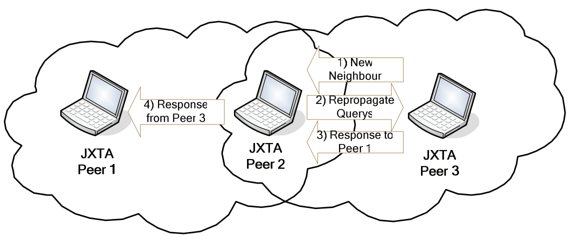

Figure 1: Query resolution after user 3 discovery.

For MANET-JXTA, the VON connecting the peers is created on-demand after MANET-RVP is used to answer a query. VON links are defined by the path elements contained in the answer message headers. Therefore, after a query resolution it matches the physical MANET topology, with minimum communication overhead. All neighbour peers in the VON are within radio coverage. If peers move from their original position, MANET routing protocols can be used to maintain the VON links as long as the application pipes are alive.

JXTA reference implementation (JXTA, 06) only supported TCP and HTTP transport protocols. MANET-JXTA added an UDP transport protocol module. The use of UDP links eventually reduces the end-to-end jitter on the application pipe's communications over 802.11 error prone WLANs (Zhang, 04). However, no quality of service guarantees are provided by the existing 802.11 MAC protocols on ad-hoc mode, and the implemented UDP module does not support any traffic differentiation mechanisms either.

3 MANET-VOVON APPLICATION

The MANET-Voice over VON (VoVON) application implements a telephony application on a MANET enabled version of JXTA service. Figure 2 shows the overall VoVON architecture.

Within MANET-JXTA enabled peers, VoVON supports asynchronous telephony call setup to a remote peer. VoVON application publishes a VoVON ModuleSpecificAdvertisement, which specifies the control receiving pipe advertisement. Therefore, using MANET-JXTA resolver service, other VoVON peers can search for VoVON advertisements, and can establish a call using the pipe advertisement.

If a MANET-JXTA RV peer exists interconnecting the MANET to the Internet (forming a meshed wireless network), the application supports synchronous call setup using standard JXTA protocols and services.

The model also includes a JXTA-SIP gateway (figure 2), which allows standard SIP clients to make calls to JXTA peers and to receive calls from JXTA peers, without modifying the user agent. The gateway associates a DNS domain name to the VoVON's JXTA name space, and encapsulates the RTP audio stream into XML messages, when needed. It supports direct interaction with SIP users, or indirect interaction, through SIP proxies.

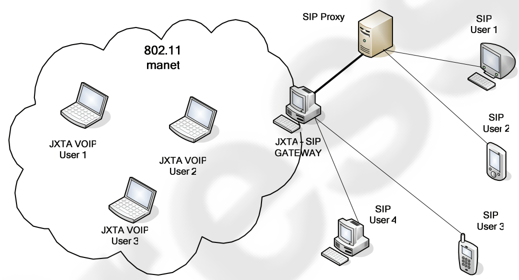

Figure 2: MANET-VoVON architecture.

3.1 Signalling

The VoVON's signalling messages were defined as a XML mapping of the SIP messages. In order to handle JXTA pipe unreliability, all VoVON messages are confirmed. A timer is used to retransmit them if no acknowledge is received and up to nine attempts are made. Table 1 presents a subset of the messages supported by VoVON.

Call setup original three-way hand-shake was mapped into a four-way VoVON message hand-shake over a pipe channel. *Call_Req* message includes the classical INVITE fields (e.g. Session Description Protocol (SDP) field), but adds a TTL field that defines for how long the call request is valid. *Call_Reply* message acknowledges the reception of *Call_Req*, and is equivalent to SIP's "100 trying" message. *Call_OK* message implements the "200 OK" SIP's message, with the corresponding fields (including the SDP field), and the SIP codec negotiation rules are followed. VoVON adds two *Call_Connect* messages to initialize audio streaming on each way, containing private pipe advertisements (see figure 3).

The SDP field was modified, to allow the definition of either a RTP endpoint (RTP protocol, IP address and the port number); or a JXTA pipe (receptor pipe advertisement string). Through pipes audio is transmitted over the VON links crossing firewalls and NAT routers, reaching any peer within JXTA VON. Pipes usage also minimizes the network setup time on a MANET – no MANET routing overhead exists. On the other hand, RTP streams introduce an initial routing overhead – it is necessary to discover the IP route between both peers before sending the audio samples. Depending on the MANET routing algorithm (Perkins, 01) discovering an IP route for a remote peer several hops away may mean: flooding the entire MANET with a "route request" packet for on-demand protocols (e.g. AODV, DSR, etc.); or continuously updating a routing table for proactive protocols (e.g. DSDV, OLSR). However, pipes may introduce a huge bandwidth overhead (Antoniu, 05).

Table 1: MANET-VoVON signalling messages and their mapping to SIP messages.

SIP	MANET-VoVON
INVITE	Call_Req / Call_Reply
200 OK (for INVITE)	Call_OK
	Call_Connect
ACK (for INVITE)	Call_Status_Reply_OK
BYE / 200 OK	Call_End / Call_End_ACK

VoVON call termination is implemented by a two way message hand-shake. No REGISTER message is included in VoVON signalling. Instead, all peer associations are made using MANET-JXTA RS (resolver service), more adapted to the unstable nature of the MANETs.

3.2 SIP Interoperation

SIP interoperation with VoVON is supported by JXTA-SIP gateways. A JXTA-SIP gateway publishes a JXTA-SIP gateway advertisement instead of a VoVON advertisement, which includes additional gateway specific information. Therefore, using the MANET-JXTA RS, other VoVON peers can search for JXTA-SIP advertisements, and can use them to establish a call to a SIP client.

A JXTA-SIP gateway is associated with a DNS domain name and implements the SIP protocol stack to communicate with SIP users and SIP proxies. Each JXTA user is associated with a VOVON uniform resource identifier (URI) within the JXTA network. Thus URI can be associated with a SIP URI identifying the domain when seen through a JXTA-SIP gateway:

vovon:(name JXTA)
sip:(name JXTA) @ domain DNS

When a VoVON client requests a call setup to a SIP URI, the VoVON application uses MANET-JXTA RS to search for VoVON advertisements with the URI's name component. If the search fails, MANET-JXTA RS is then used to find JXTA-SIP gateway advertisements. A bidirectional connection is established to one of the JXTA-SIP gateways.

Figure 3 shows the messages exchanged through the control pipes during a call setup from a MANET-JXTA client to a SIP user agent connected to a SIP proxy. JXTA-SIP gateway runs the VoVON setup message hand-shake at the JXTA interface with the VoVON user, and perform the translations described above.

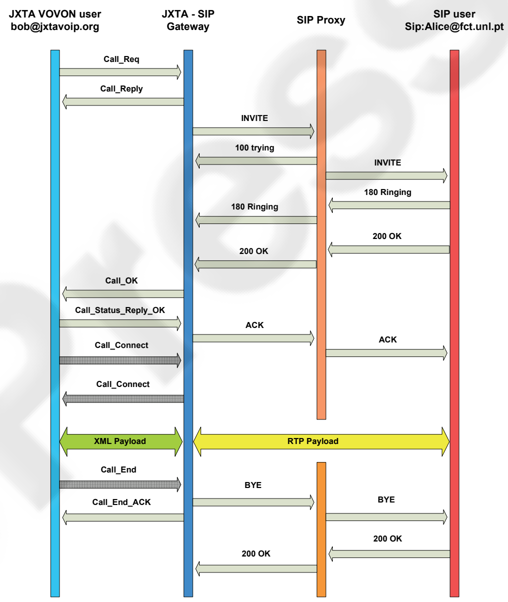

Figure 3: call setup from JXTA to SIP using a SIP proxy.

JXTA-SIP gateway maintains all audio encoding specifications in the SDP audio fields it sends to the SIP proxy or to the SIP user. Although, it swaps all domain names with references to its domain name (IP address) and to a dynamic UDP port, which is allocated in response to a new VoVON client request. Therefore, the JXTA-SIP gateway always intermediates the audio streams from VoVON users and SIP users, converting the encapsulation formats and the addresses between the VoVON user (XML message or RTP packets) and the SIP user (RTP packets). In case of JXTA-SIP gateway failure, all active connections are torn down, and VoVON clients have to select a new MANET-JXTA gateway and create new connections (possibly associated with a different DNS domain).

Calls can be disconnected by any of the call participants. When SIP's "BYE" or VoVON's *Call_End* messages are received, JXTA-SIP end the audio streaming, and completes the disconnection signalling exchanges for each connection.

3.3 Interaction with SIP Proxy

JXTA-SIP gateway can be configured to register itself on a SIP Proxy. However, the usual SIP procedure of sending a REGISTER message for each VoVON user cannot be used since the SIP-Proxy does not know who is available in the MANET.

Instead, JXTA-SIP gateway sends the SIP Proxy a domain REGISTER message associating itself with all names within a DNS domain name. A SIP proxy handles an INVITE request by trying to match a registered full URI. If none exists, it tries to match a registered domain. In this case, it forwards the call to the gateway and decrements the "Max-header" field in the SIP header.

When a JXTA-SIP gateway receives a call to the JXTA-SIP's DNS domain, it answers back with a "100 trying" message and starts a MANET-JXTA RS query looking for a VoVON user with the URI name. This query carries a zero TTL value, meaning that it only accepts calls when the VoVON user is reachable at that instant. If the user is located, the messages exchange follow a pattern similar to the one presented in figure 3. Otherwise the call is rejected.

4 APPLICATION

A VoVON prototype application is currently being implemented over a prototype implementation of MANET-JXTA. MANET-JXTA current prototype extended a previous prototype presented in (Oliveira, 05b) with the RS and the MANET-RVP presented above.

4.1 VoVON Prototype

We implemented a MANET-JXTA application which combines the VoVON user application and the JXTA-SIP gateway application.

VoVON signalling is exchanged using two control unidirectional pipes: one receiver pipe to receive connection requests from remote client pipes and other signalling messages; one client pipe for connecting to remote control VoVON receiver pipes. Audio can be transmitted using two JXTA pipes (one for each direction), or using two RTP streams over UDP.

Audio is handled using Java Media Framework (JMF) version 2.1.1e (JMF, 06). The application captures and formats audio into RTP format using two JMF modules. VoVON application always tries to select the GSM codec, due to its compression rate (13 Kbit/s), but it is compatible with any format supported by JMF.

Audio is transmitted over a JXTA pipe with the RTP header, encapsulated in a XML message. Therefore the JXTA pipe acts as a tunnel that transparently sends audio samples through the JXTA VON. JMF objects are connected to JXTA pipes using a local UDP socket. JMF objects stream audio to a local UDP socket, where the RTP packets are received and sent through an audio JXTA pipe. RTP packets are extracted from JXTA messages and sent to the JMF objects using the same UDP socket. Therefore, the architecture is not optimised for sending audio over JXTA pipes because it introduces additional jitter due to local socket transmissions at the sender and at the receiver side.

The audio player is capable of compensating some jitter, by storing the received audio samples on a circular buffer and delaying their presentation. However, if an audio sample is received after its presentation time, it is discarded. By default, JMF uses an audio buffer length of 250 ms, but it can be configured to a maximum value of 1000 ms.

The graphical interface, presented in figure 4, supports the application configuration and the call initiation and termination. The user is allowed to define its JXTA name (*My Name*), the JXTA group it joins (*Name*), if it is a rendezvous peer (in *RendeVouz* menu), and the gateway configuration (RTP port and IP address, and the DNS domain name, in the *Gateway* menu). The application presents a list of known VoVON peers in a table, but also allows the introduction of a SIP or VOVON URI (by default is a SIP URI) to start a new call.

The Current MANET-JXTA implementation can only use one of the transport protocols for all JXTA: TCP or UDP. It cannot use both in parallel. When the application is run for the first time, it opens a MANET-JXTA configuration menu where the transport module and the transport parameters are chosen. The application stores the configuration in a ".jxta" directory for the next run.

4.2 Performance Measurements

VoVON application initial performance tests focused on its overhead, compared to other Voice over IP applications, and on its SIP interoperability.

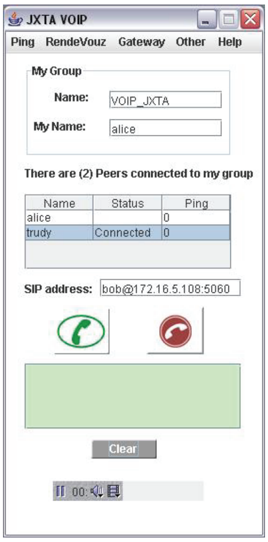

Figure 4: VoVON application user interface.

Audio can be sent using RTP packets over UDP transport protocol, or using JXTA pipes over UDP or TCP protocols. For GSM audio encoding, eighty audio samples per second are sent. Using the Ethereal protocol analyser (Ethereal, 06), we measured the total number of bytes sent, represented in table 2. JXTA pipes send two IP packet fragments per sample, and TCP sends an acknowledge packet. As expected, JXTA pipes introduce a huge overhead that results from the presence of an Endpoint Router and a VoVON headers, and from the XML encoding overhead. This conclusion is consistent with (Antoniu, 05), confirming that pipe encapsulation is only valuable when bandwidth is not a problem, the connection requires NAT or firewall crossing, or the connection has a short life, dissuading any extra MANET-route discovery delay.

Table 2: JXTA Pipes communication overhead for GSM audio encoding.

Audio Encoding	Sample size [Bytes]
RTP	153 Bytes/sample
JXTA Pipes over UDP	1799 Bytes/sample
JXTA Pipes over TCP	1818 Bytes/sample

Even with this large bandwidth overhead, it is only possible to send audio over a JXTA pipe if the end-to-end jitter introduced by the encoding and decoding process is not significant, compared with using RTP channels. Jitter is usually defined by equation 1 (Schulzrinne, 03). Jitter is the mean deviation (smoothed absolute value) of the difference D between the arrival time and the sender time tag. The smoothing factor of 1/16 is used to reduce the effect of huge isolated deviation peaks, common on 802.11b WLANs.

$$J(i)= J(i-1) + (|D(i-1,i)| - J(i-1))/16 \qquad (1)$$

Table 3 presents the average jitter value measured at the receiver side for two SIP Communicator user agents (SipCommunicator, 06) using RTP streams over UDP, and for two VoVON applications communicating on a network with only two peers, using a TCP JXTA pipe. Jitter was measured using Ethereal protocol analyser (which uses equation 1), capturing the packets just before they are received by the JMF object. The results show that the use of JXTA TCP pipes introduces an average jitter overhead of about 10 msec (52.8 %) for one JXTA hop. This value can be compensated using JMF buffering, and is clearly below the maximum end-to-end delay defined by ITU-T for telephony applications (150 msec).

Table 3: Average jitter for an unloaded Ethernet, measured using Ethereal.

Audio encapsulation and network	Jitter [ms]
RTP	19,23
RTP over JXTA TCP pipes	29,39

Table 4 presents the average jitter value measured for an extensive set of experiments, with two VoVON applications communicating on a network with only two peers, for four different setups. The jitter on the JXTA pipe was measured when the XML audio messages are received from the pipe (excluding the jitter introduced at the last local socket retransmission), using a measuring function included in the VoVON application. The pipes' jitter measurements show that 802.11b network is responsible for a 28% increase on the average jitter for UDP pipes, and a 1% increase for TCP pipes, comparing them to the Fast Ethernet measurements. This behaviour difference is probably due to a high overhead handling fragmented 802.11b UDP packets on MS Windows XP. Unfortunately, it was not possible to test MANET-VoVON on Linux because the kernel does not handle fragmented multicast UDP packets, required for MANET-RVP. It also shows that TCP congestion control effects are not noticeable possibly because traffic bandwidth is very low.

VoVON JXTA-SIP gateway interoperability was tested by connecting a VoVON client to a VoVON JXTA-SIP gateway and to a SIP Communicator (SipCommunicator, 06) user agent, using 802.11b WLANs and Fast Ethernet. The system was also tested with JAIN-SIP Proxy (NIST-SIP, 03). In all cases, it was possible to start calls from the VoVON clients and the SIP Communicator clients.

Table 4: Jitter measurements for unloaded networks using VoVON and JXTA pipes.

Audio encapsulation and network	Jitter [ms]
Pipes over UDP on Fast Ethernet	16,18
Pipes over TCP on Fast Ethernet	16,45
Pipes over UDP on 802.11b	20,75
Pipes over TCP on 802.11b	16,68

5 CONCLUSIONS AND FURTHER WORK

This paper shows that MANET-JXTA peer-to-peer open platform can be used to implement real-time applications on a MANET or on meshed networks. It shows that using MANET-RVP deferred search, it is simple to have call setup triggered by connection availability in a simple way. Other alternative approaches, like using SIP events (Roach, 02), would be complex. A centralized approach would require a connection to a third party: the location server. A decentralized approach would require periodic flooding of searches or replies, controlled by the application.

Internet Telephony based on p2p architectures (i.e. Skype and P2P-SIP) has focused primarily on the user's location tracking. This paper proposed and analysed the possibility of also using the p2p network for streaming audio, concluding that due to the huge bandwidth overhead, its use should be restricted to extreme situations: NAT and firewall crossing, or short-lived connections. Results show that the end-to-end jitter is acceptable when audio is sent through a p2p VON.

Future work includes the continuation of the design and implementation of MANET-JXTA and MANET-VoVON. Quality of service can be improved by introducing message differentiation mechanisms in JXTA pipes, and by introducing advanced flow control mechanisms based on network state measurements. Further prototype tuning is also necessary on the JXTA UDP transport module and on the audio pipe communication. The audio pipe's communication jitter can be reduced by connecting the JMF classes directly to the JXTA pipes using the JMF RTP socket interface. Finally, comprehensive MANET multi-hop tests will be done and reported on a future paper.

REFERENCES

Antoniu, G., Hatcher, P., Jan, M., Noblet, D., 2005. Performance Evaluation of JXTA Communication Layers. In *GP2PC'2005, 5th International Workshop on Global and Peer-to-Peer Computing*, May 2005.

Baset, S., Schulzrinne, H., 2004. An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol. Technical Report CUCS-039-04, Computer Science Department, Columbia University, New York, USA, Sep 2004.

Ethereal, 2006. Ethereal: A Network Protocol Analyser. Retrieved from http://www.ethereal.com/

NIST-SIP, 2003. A JAIN-SIP Proxy for the People!. Retrieved from http://snad.ncsl.nist.gov/proj/iptel/

JMF, 2006. Java Media Framework API (JMF). Retrieved from http://java.sun.com/products/java-media/jmf/

JXTA, 2004. Project JXTA: JXTA v2.0 Protocols Specification. Retrieved from http://spec.jxta.org/nonav/v1.0/docbook/JXTAProtocols.html

JXTA, 2006. JXTA: Get Connected. Retrieved from http://www.jxta.org

Oliveira, R., Bernardo, L., Pinto, P., 2005. Flooding Techniques for Resource Discovery on High Mobility MANETs. In *IWWAN'05, International Workshop on Wireless Ad-Hoc Networks*, May 2005.

Oliveira, R., Bernardo, L., Ruivo, N., Pinto, P., 2005b. Searching for PI resources on MANETs using JXTA. In *AICT-SAPIR-ELETE'05, Advanced Industrial Conference on Telecommunications/Service Assurance with Partial and Intermittent Resources Conference/E-Learning on Telecommunications Workshop*, pp. 371-376, IEEE Press, July 2005.

Perkins, C., 2001. Ad Hoc Networking, AddisonWesley.

Roach, A., June 2002. Session Initiation Protocol (SIP)-Specific Event Notification. IETF RFC 3265, June 2002.

Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E., 2002. SIP: Session Initiation Protocol. IETF RFC 3261, June 2002.

Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V., 2003. RTP: A Transport Protocol for Real-Time Applications. IETF RFC 3550, July 2003.

Singh, K., Schulzrinne, H., 2005. Peer-to-Peer Internet Telephony using SIP. In *NOSSDAV'05*, *International Workshop on Network and Operating Systems Support for Digital Audio and Video*, pp. 63-68. ACM Press.

SipCommunicator, 2006. SIP Communicator - the Java VoIP and Instant Messaging client. Retrieved from https://sip-communicator.dev.java.net/

Skype, 2006. Skype Explained. Retrieved from http://www.skype.com/products/explained.html

Smith, J., 1998. *The book*, The publishing company. London, 2nd edition.

Zhang, X., Schulzrinne, H., 2004. Voice over TCP and UDP. Technical Report CUCS-033-04, Computer Science Department, Columbia University, New York, USA, Sep 2004.