By Zefir Kurtisi, Xiaoyuan Gu, and Lars Wolf

ENABLING NETWORK-CENTRIC MUSIC PERFORMANCE IN WIDE-AREA NETWORKS

NMP on the Internet is not only possible, its delay bounds can satisfy the tight requirements involved in human perception while generating high-end audio quality for musicians and listeners alike.

The ubiquitous availability of broadband Internet connectivity at home is driving the use of the Internet for entertainment and other forms of recreation. Paving the way for many demanding multimedia applications over IP, it has accelerated the emergence of new ideas for network-centric collaborative work that was impossible only a few years ago for both technical and economic reasons.

Among many new types of networked entertainment genres, network-centric music performance, or NMP, [3] represents a vision of multi-party musical performance delivered through cyberspace that strives to overcome the inherent limitations in conventional rehearsals and concerts. NMP refers to a system that allows musicians who are physically separated, even over vast geographical distances, to participate in rehearsals and concerts across the Internet with bounded delay and acceptable audio quality. Similar work is exemplified by the Stanford SoundWire Project [2] and the Conductor Driven Scheme [1].

In [3], we focused on building a proof-of-concept NMP system prototype in a LAN. We also investigated delay and audio quality as they are affected by end systems. To make it possible for NMP to go beyond being only a laboratory tool, we've had to address a number of technological challenges. Above all, we've found that meeting its extreme stringent delay bound is a critical prerequisite. Here, we describe the NMP application boundary, presenting some of our evaluation results on NMP operation in wide-area networks.

The figure outlines a set-up of an NMP system consisting of a centralized server acting as the mixer and one or more clients connected through the Internet. The client(s) produce and send audio packets to the server. The server puts packets from each client into a separate queue, applies mixing, and returns the mixed packets to all clients where the final contents are played out through the users' audio playback devices.

In order for the musicians in an NMP session to interact with one another in a natural way, the end-to-end delay must be kept below human perception; 30msec [4] is a widely recognized bound. End-to-end delay here means the total delay (such as between hitting a piano key and hearing the time-synchro-

nous performance of the other musicians) perceptible to a musician.

Due to such a tight delay bound, NMP applicability is restricted to some specific scenarios. The distance between each collaborating client and the server is inevitably limited by the signal propagation speed and link capacity and is further reduced by delays introduced in processing at routers in the networks and at the end systems (the NMP client and server). Since we do not assume a QoS-supported Internet, our optimization strategy is to minimize the end system delays in order to increase the remaining budget for network delay that directly affects the application boundary of an NMP session.

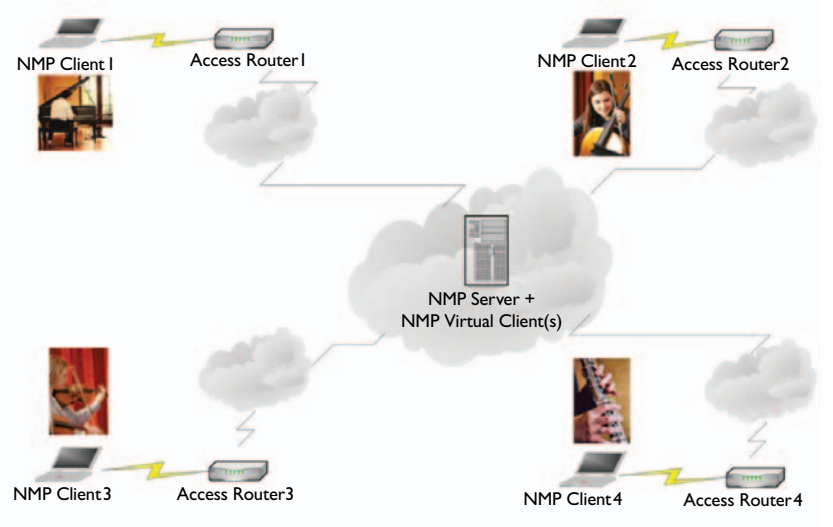
Closer inspection of the processing and data paths in the end systems reveals numerous delay sources that for simplicity can be divided into buffering and computational latencies. The processing scheme in the client's sound card defines the application's buffering granularity and is a major component of end-system delay. To assure continuous recording and playback, sound cards preserve internal hardware buffers before the digital-to-analog converter and after the analog-to-digital converter.

While the converters operate on the front buffer set, the operating system processes the back buffer set. However, the OS is not able to process audio data immediately after the sound card has flipped the front and back buffers, since the sound card's interrupt service routine must be scheduled first. Therefore, at least one additional packet must be buffered by the OS in each direction.

Hence, the sound card (plus the OS) introduce a minimum buffering delay of four buffer units at the client. The size of these units is constant within a session and has a predefined size that is set according to the sampling rate being used. We use a sampling rate of 48kHz with 128 samples per buffer unit. The settings result in a packet buffer delay of 2.667msec. It defines the atomic buffering unit for the application's buffer dimensioning (and is denoted as $t\varphi$ here).

At the server, under realistic network conditions, received audio packets must first be buffered before they are mixed in order to compensate for network jitter; ideal network conditions are assumed for estimating the application boundary. Since in this case de-jitter buffering is not used, the overall latency at the server is only the computational overhead.

Compared to buffering delay, processing delay due to computational overhead is a magnitude lower and can be ignored. We measured a total computational overhead for an audio packet of 50μs–180μs. We thus approximated the total end-system delay under ideal network conditions to the buffering delay at the client, or $t\varphi$ = 10.7msec, allowing 30msec–10.7msec = 19.3msec for network delay.



Scenarios involving network-centric music performance.

Under Different Network Conditions

The network delay budget of 19.3msec is used to guide the delay jitter compensation at the end systems. This compensation is typically done by configuring the de-jitter buffers at the server and client with granularity of $t\varphi$ = 2.667msec. More buffers increase the overall tolerable delay but have a better chance of smoothing out the jitters at the network. Trading delay for loss enables an NMP system to operate under different network conditions and to match different user expectations.

Two network set-ups described in the following paragraphs demonstrate that NMP is operable within the defined delay bound. The NMP sessions are formed by three NMP clients connected to a server. Packet loss rate is measured for a given number of deployed buffers in intervals of 1,000 packets for a total period of 20 minutes. Late packets are treated as losses due to the real-time semantics of the application.

The first NMP set-up is for a LAN; the machines are directly connected through a fast Ethernet switch, with a measured round-trip time of (min/avg/max/dev = 272/310/2630/193)μs. The second set-up is for a WAN spanning a one-way distance of about 300km where all the computers are part of the German

In order for the musicians in an NMP session to interact with one another in a natural way, THE END-TO-END DELAY MUST BE KEPT BELOW HUMAN PERCEPTION; 30msec is a widely recognized bound.

Research Network. This set-up is representative of mid-size networks, with all nodes being reachable through a limited number of hops. Here, we measured a round-trip time of (8825/9315/15064/794)µs over 11 hops.

The table summarizes our measurement results. A LAN session can be set up with no additional network buffers, resulting in a total delay of 10.7msec and a mean packet loss ratio of 0.06% suitable for high audio quality. The loss rate is further reduced by adding a network de-jitter buffer with one audio packet at the client and one at the server, respectively. Adding this buffer increases total delay to 16msec but reduces the packet loss rate to 0.015%.

Type	Network Buffer		Total Delay	Packet Loss Statistics		
	[packets]	[msec]	[msec]	mean	max.	dev.
LAN	0	0	10.7	0.060%	2.4%	0.173%
LAN	2	5.3	16.0	0.015%	1.6%	0.101%
WAN	6	16.0	26.7	0.689%	6.9%	1.254%
WAN	10	26.7	37.3	0.216%	3.8%	0.519%

Delay and packet loss in several scenarios.

In the WAN session, a network de-jitter buffer of three packets at both the client and the server is sufficient to keep the packet loss ratio below 0.7%. To further improve audio quality, the de-jitter buffer is enlarged by four audio packets, or 10.66msec, in the second WAN scenario. Here, the total delay is 37.3msec, and the packet loss ratio is reduced to 0.216%.

Along with these statistics and evaluation, we also conducted subjective listening and usability tests with musicians in both scenarios. Their evaluations further confirmed the good objective results.

Conclusion

This work is a significant step toward the general usability of NMP, bringing it out of the laboratory and one research LAN to potentially multiple commercial-scale WANs. We have shown that NMP is not only possible but can satisfy the stringent delay bound of 30msec. With long-distance evaluation, we confirmed that network delay affects NMP behavior, indicating many possibilities for further enhancement.

One aspect of future work is to investigate whether the absolute delay bound of 30msec might be relaxed, with musicians able to adapt to the introduced delay, along with the extent of that relaxation. Relaxing that bound will surely extend the application boundary toward greater coverage in NMP sessions. Another is error concealment and correction schemes that would help application developers deal with packet loss during transmission. Moreover, besides a purely technical perspective, user interface and user evaluation must be improved to prepare NMP for commercial launch and user acceptance. C

References

1. Bouillot, N. The auditory consistency in distributed music performance: A conductor-based synchronization. *ISDM Info et com Sciences for Decision 13* (Mar. 2004), 129–137.
2. Chafe, C., Wilson, S., Leistikow. R., Chisholm, D., and Scavone, G. A simplified approach to high-quality music and sound over IP. In *Proceedings of the COST-G6 Conference on Digital Audio Effects* (Verona, Italy, Dec. 7–9, 2000), 159–164.
3. Gu, X., Dick, M., Kurtisi, Z., Noyer, U., and Wolf, L. Network-centric music performance: Practice and experiments. *IEEE Communications Magazine 43,* 6 (June 2005), 86–93.
4. Schuett, N. *The Effects of Latency on Ensemble Performance.* Master's Thesis, Stanford University, Palo, Alto, CA, May 2002.

Zefir Kurtisi (kurtisi@ibr.cs.tu-bs.de) is a research staff member in the Computer Science Department at the Technische Universitat Braunschweig, Braunschweig, Germany.
Xiaoyuan Gu (xiaogu@ibr.cs.tu-bs.de) is a research staff member in the Computer Science Department at the Technische Universitat Braunschweig, Braunschweig, Germany.
Lars Wolf (wolf@ibr.cs.tu-bs.de) is a professor in and dean of the Computer Science Department at the Technische Universitat Braunschweig, Braunschweig, Germany.