QoS-Aware Voice-Over-IP Conferencing Service using Composable Service Overlay Network[*]

Xiaohui Gu[†], Zon-Yin Shae[‡], Rong N. Chang, Klara Nahrstedt

Abstract

As Internet evolves into a service delivery infrastructure, IP telephony has become feasible and will be expected to meet the toll quality of traditional telephone service. In this paper, we present Venus, a novel VoIP conferencing system with quality-of-service (QoS) management. Venus is constructed as an application-level service overlay network, where each overlay node provides voice mixing service as well as application-level data routing. Venus can dynamically compose a voice mixing path for each conferencing session based on both multi-constrained QoS requirements (e.g., delay, loss rate, bandwidth) and global system optimization goal (e.g., maximizing system utilization). Venus provides runtime voice mixing path maintenance to achieve robust VoIP conferencing service. Large-scale simulation results illustrate the practicality and efficiency of the Venus system.

1 Introduction

Internet has evolved into an indispensable service delivery infrastructure instead of merely providing host connectivity. IP telephony is a promising Internet service, particularly because of the significant revenue it can generate. A simple VoIP system includes two participants. The original voice signal is periodically sampled, encoded into a constant bit rate stream, and sent over the Internet to the receiving end where the packet is first stored in the playback buffer and then decoded into the speech signal. Previous assessment study [5] has indicated that the user perceived quality of VoIP service is mainly affected by the end-to-end delay and packet loss rate of the VoIP system.

In this paper, we consider a more advanced VoIP service: multipoint conference, which includes three or more participates. The traditional design of the VoIP conferencing system is to use a centralized multipoint control unit (MCU), which mixes the voice signals of all other participants and sends the mixing speech signal back to each conference member. However, the centralized approach suffers from a number of limitations. First, the centralized MCU has the scalability problem, where a single MCU can be short of resources for a large conference including hundreds of participants. Second, the centralized approach presents poor reliability due to the single point of failure. Third, the centralized approach can lead to degraded quality-of-service (QoS) when conference members are far away from the centralized MCU. Another alternative VoIP conferencing system design is to employ either IP-layer or application-layer multicast [2]. Although the multicast approach can theoretically save network bandwidth, the construction and maintenance of multiple conference trees are often too complicated for practical use, especially when we consider multi-constrained QoS requirements. To this end, we propose a simple yet efficient VoIP conferencing system *Venus* based on the composable service overlay network [3].

The Venus VoIP conferencing system is constructed as an application-level overlay network, where each overlay node provides both voice mixing and application-level data routing. Given a conferencing request, the system dynamically composes a *mixing service path* based on (1) the number and locations of conference members and (2) the multi-constrained QoS requirements (e.g., delay, loss rate). The mixing service path consists of a number of overlay nodes, which is used for resource aggregation and improved reliability. The system employs multi-dimensional failure recovery for the mixing service path to provide failure resilient VoIP conferencing service. The rest of the paper is organized as follows. Section 2 introduces the Venus system model. Section 3 describes the system design details. Section 4 presents the performance evaluation. Finally, the

[*] This work was partially supported by the NASA grant NASA NAG 2-1406, NSF grant 9870736, 9970139. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF, NASA or U.S. Government.

[†] Xiaohui Gu and Klara Nahrstedt are affiliated with Department of Computer Science, University of Illinois at Urbana-Champaign. Email: {xgu, klara} @ cs.uiuc.edu

[‡] Zon-Yin Shae and Rong N. Chang are affiliated with IBM T.J. Watson Research Center, Hawthorne, NY. Email: {zshae, rong} @ us.ibm.com

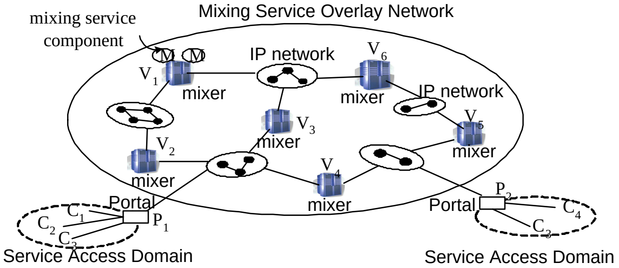

Figure 1. Illustration of the mixing service overlay network.

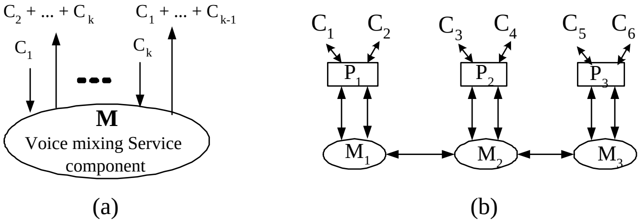

Figure 2. Illustration of voice mixing service component and voice mixing path.

paper concludes in Section 5.

2 VoIP Conferencing System Model

We now introduce the Venus system model that consists of (1) mixing service overlay network, (2) mixing service component, and (3) mixing service path.

Mixing service overlay network. The mixing service overlay network (MSON) is the substrate of the Venus system, which is illustrated by Figure 1. MSON interconnects distributed overlay nodes into an application-level overlay network. Each overlay node encapsulates both functions of an MCU and an application-level router[1], which is called a *mixer*. Each mixer can be owned by a third-party service provider who provides a service level agreement (SLA) specifying the resource and QoS properties of the mixer. For QoS provisioning, we introduce *portals* and *monitors*, which form the management plane of the MSON. The portals are the service access points for conferencing clients, which also define the QoS assurance boundary of the Venus system. The monitors are co-located with each mixer and reports the local network states to all portals. Thus, we assume that each portal has the global view of the MSON[2].

Mixing service component. Each mixing service component takes k input voice signals and generates k mixing signals, which is illustrated by Figure 2 (a). If the service component has k input voice signals, c_1, ..., c_k, the output to $c_j (1 \leq j \leq k)$ is a mix of voices $\sum_{i=1}^{k} c_i - c_j$. Each mixer can simultaneously instantiate multiple service components for different conferencing sessions. Each mixer has a resource capacity defined by the maximum number of input/output connections.

Mixing service path. We can compose a set of mixing service components into a mixing service path[3], which is illustrated by Figure 2 (b). We can prove that each client connected to the mixing service path receives a mixed signal of all other conference members using the induction proof on the length of the path. Due to the space limitation, we omit the proof details here. The QoS values (e.g., delay , loss rate) of the conferencing service is defined as the QoS values of the mixing service path between two end portals (e.g., P_1 and P_3 in Figure 2 (b)), which include both network delays/loss rates and mixing service delays/loss rates. The rationale of the definition is that the portal-to-portal QoS values of the mixing service path is within the controllable range of the Venus system, which also define the major parts of the voice qualities perceived by the end-user.

3 VoIP Conferencing System Design

In this section, we present the design details of the Venus system. For simplicity, we only consider the pre-scheduled conferencing where the number of participants and their locations are known in advance.

3.1 Mixing Service Path Composition

Each client accesses the VoIP conferencing service through the local portal. Thus, The clients in one edge network domain form a client group[4] g_i and will connect to the same local portal P_i. For example, in Figure 1, clients c_1, c_2, and c_3 connect to the portal P_1, and clients c_4 and c_5 connect to the portal P_2. Each portal P_i selects a number of candidate mixers for each client group. The candidate mixers can be selected based on different heuristics: (1) *Random approach*, where P_i randomly selects a number of

[1] For simplicity, we assume that overlay data routing uses the shortest path routing algorithm based on delay metric only.

[2] Such a assumption is practical for Venus since it targets to Enterprize overlay network with a few hundred nodes.

[3] We are aware that mixing service components can be composed into other topologies such as trees, which however requires more complicated construction and maintenance algorithms. For simplicity, we limit ourselves to the case of composing mixing service path in this paper.

[4] If the size of the client group exceeds certain threshold, the system will split the original group into several smaller groups.

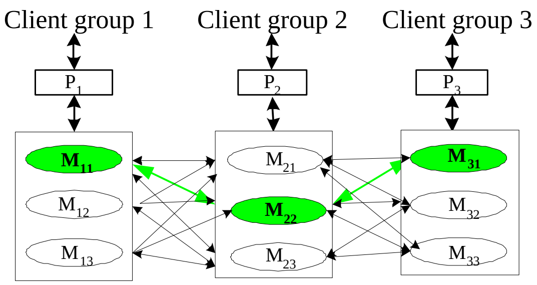

Figure 3. Illustration of mixing service path composition.

mixers; (2) *Greedy approach*, where P_i selects a number of nearby mixers based on some distance metric; and (3) *Hybrid approach*, where half of candidates are selected from nearby nodes and the other half are randomly selected. Venus adopts the hybrid approach in order to achieve best tradeoff between performance guarantee and load balancing. For example, in Figure 3, each portal selects three candidate mixers for each client group.

We define the source portal as the portal to which the conference initiator is connected. The source portal is responsible for composing the best qualified mixing service path using the above candidate mixers. We can define the problem of mixing service path composition as a multi-constrained optimization problem. We want to find the mixing service path that optimizes the system resource utilization subject to the multi-constrained QoS requirements (e.g., delay and loss rate) for the conferencing service. In [3], we have proven the above problem to be NP-complete and provided a modified Dijkstra algorithm for the above problem, which can achieve near-optimal performance. The key idea is to introduce an aggregated cost metric that considers multiple factors including multiple QoS metrics and resource utilization. To satisfy multiple QoS constraints, we modify the Dijkstra algorithm by adaptively adjusting the importance weights of different factors in the aggregated cost metric based on their constraint pressures. More details of the algorithm can be found in [3].

An interesting property of composing a mixing service path is that the path can be any permutation of the voice mixing service components. Thus, Venus explores all permutations of the mixing service path selected by the above modified Dijkstra algorithm. For example, in Figure 3, the final mixing service path can be either $M_{11} \leftrightarrow M_{22} \leftrightarrow M_{31}$ or $M_{11} \leftrightarrow M_{31} \leftrightarrow M_{22}$. We formalize the above problem into a travelling salesman problem (TSP), which is to find a cheapest way of visiting all the selected mixers and returning to the start mixer. The cost definition is the same as the above modified Dijkstra algorithm. Since the TSP is also NP-complete, Venus uses a heuristic algorithm for finding the best mixing service path from all permutations.

After deciding the mixing service path, Venus creates a voice mixing service component for each client group on the selected mixer. Then, Venus sets up the conferencing session and notifies all conference members that the conferencing service is ready for use.

3.2 Mixing Service Path Maintenance

During runtime, the conferencing service can experience significant QoS violations or service outages due to the failures of network links or overlay nodes. To achieve robust VoIP conferencing service, Venus provides runtime failure detection and recovery mechanisms to maintain the availability and QoS of all active mixing service paths. The source portal of a VoIP conferencing session is responsible for monitoring and maintaining the liveness and QoS of the mixing service path for the conferencing session. The failure recovery is performed at multiple layers. First, Venus relies on the overlay data routing to recover the outage failures of network links [4, 1]. For example, in Figure 3, if the overlay path from M_{11} to M_{22} is broken, the overlay data routing layer will dynamically find another overlay path. However, overlay data re-routing cannot recover the failures of mixers. Moreover, the overlay data re-routing only considers the delay metric, which can cause violations of other QoS metrics such as loss rate. Under those circumstances, the source portal dynamically recomposes a new mixing service path to recover the failures. To achieve fast failure recovery, Venus adopts a localized path repair algorithm [3]. The source portal finds a new mixing service path that does not include the broken mixers but has the largest number of common mixers with the old mixing service path.

4 Performance Evaluation

We evaluate performance of Venus using large-scale simulations. We first use a degree-based Internet topology generator Inet 3.0 [6] to generate a power-law random graph topology with 3200 nodes to represent the IP-layer network. We then randomly select N nodes as Venus nodes and $N \cdot 0.3$ nodes as portals. The initial resource capacity and average QoS values of each network link and mixer are uniformly distributed. To simulate performance variations in the Internet, the QoS values are updated per time unit according to a uniform distribution function with the predefined mean value. During each time unit, certain number of VoIP conferencing requests are generated. Each request includes 2 to 10 client groups whose sizes are between 5 to 20 clients. Each conferencing session lasts 5 to 30 time units. In our experiments, each portal selects 10 candidate mixers for each client group.

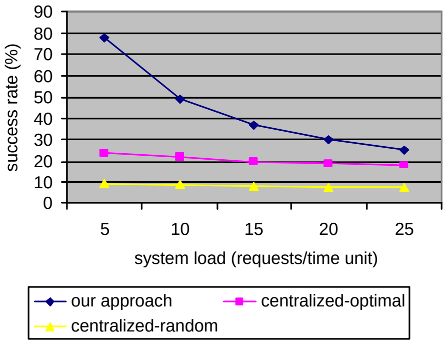

Figure 4. Illustration of service success rate comparison.

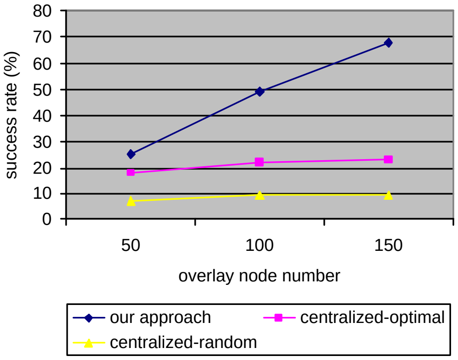

Figure 5. Illustration of scaling property comparison.

We define the metric *service success rate* for evaluating the performance evaluation. A QoS-aware VoIP conferencing service is said to be provisioned successfully if and only if (1) the system has enough resources for the conferencing session, and (2) average QoS values (i.e., delay, loss rate) measured over the whole conferencing session satisfy the required QoS values. For comparison, we also implement the *centralized-random* algorithm that randomly select a mixer for each conferencing request, and *centralized-optimal* algorithm that selects the best single mixer for each conferencing request.

Figure 4 illustrates the service success rate achieved by the three different algorithms under different system loads on an MSON with 100 mixers. We observe that Venus can achieve much higher success rates than the other two algorithms by efficiently aggregating resources of distributed mixers and finding best mixing service path under delay and loss rate constraints. Figure 5 shows the service success rate comparison under the same request rate (10 requests per time unit) on different MSONs with sizes 50, 100 and 150 nodes respectively. The results demonstrate that the Venus system presents much better scaling property than the other two approaches. The system performance increases almost linearly as we increase the number of mixers.

5 Conclusion

In this paper, we have presented a novel QoS-aware VoIP conferencing system *Venus* using an application-level composable service overlay network. Venus does not require any IP-layer or application-layer multicast support. Venus also achieves better scalability and QoS provisioning than the traditional VoIP conferencing service that uses a centralized MCU. Large-scale simulation results demonstrate the efficiency of the Venus system. The future directions of this research include (1) comparing the performance and overhead of composing mixing service components into different topologies, and (2) exploring other multi-constrained path finding algorithms.

References

[1] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris. Resilient Overlay Networks. *In Proc. 18th ACM SOSP 2001, Banff, Canada*, October 2001.

[2] J.-C. Chang and W. Liao. Application-Layer Conference Trees for Multimedia Multipoint Conferences Using Megaco/H.248. *IEEE International Conference on Multimedia and Expo (ICME 2001), Tokyo, Japan*, August 2001.

[3] X. Gu, K. Nahrstedt, R. N. Chang, and C. Ward. QoS-Assured Service Composition in Managed Service Overlay Networks. *Proc. of IEEE 23nd International Conference on Distributed Computing Systems (ICDCS 2003), Providence, RI*, May 2003.

[4] N. Kamat, J. Wang, and J. Liu. A Delay-Efficient Re-Routing Scheme for VOIP Traffic. *IEEE International Conference on Multimedia and Expo (ICME 2003), Baltimore, MD*, 2003.

[5] A. Markopoulou, F. Tobagi, and M. Karam. Assessment of VoIP Quality over Internet Backbones. *IEEE Transactions on Networking*, October 2003.

[6] J. Winick and S. Jamin. Inet3.0: Internet Topology Generator. *Tech Report UM-CSE-TR-456-02 (http://irl.eecs.umich.edu/jamin/)*, 2002.