iSNAC: Infrastructureless Social Networking at Academic Conferences

Yuanzhu Chen[*†], Walter Taylor[†], Sam Coxon[†], and Jason H. Moore[†]
[*]Department of Computer Science, Memorial University of Newfoundland, Canada
[†]Computational Genetics Laboratory, Dartmouth Medical School, Dartmouth College, USA

I. Introduction

Researchers on Mobile Ad Hoc and Delay-Tolerant Networking have been on a quest of the "killer applications" for such a technology to be useful in real civilian scenarios [1]. On the other hand, support of ubiquitous social networking is a major thrust driving research and development of mobile apps. We have identified the need for Mobile Ad Hoc and Delay-Tolerant Networking services at venues of academic conferences. In this case, users can produce messages on site and exchange them with their colleagues. The conference organizers can also use this platform to disseminate announcements as well.

Typically, large academic conferences attract researchers from academia and industry from all over the world. Many such conferences have simultaneous symposia or tracks, or a large number of posters presented the same time. It is thus desired that conference attendants are able to communicate with each other to facilitate swift information exchange. Unfortunately, existing technologies do not have such capabilities for a few reasons.

1) Despite that more and more people carry smart phones or other types of mobile devices with data services, the premium of using such services in a foreign country is often prohibitive.
2) Not all events provide complimentary Wi-Fi access. Even if they do, Wi-Fi networks are often clogged up by users who must process their e-mails or look up information from the Web.
3) A few examples of specialized equipment have been churned out of research labs for short-range data communication in the ad hoc fashion, but they aren't general enough to appeal a large audience. For any killer app to be successful, it needs to accumulate enough critical mass, and the key here is to be generic and open.

To meet this need, we developed an application on the Apple iPad [2] using the Bluetooth short-range radios on the iOS devices. This app is dubbed iSNAC (Infrastructureless Social Networking at Academic Conferences) since it does not rely on any infrastructure at the conference venue, such as Wi-Fi access points or cellular base-stations. It is a scenario where Mobile Ad Hoc and Delay-Tolerant Networking is rendered useful. More importantly, iSNAC is a user-space application built-on the iOS GameKit Framework, and it does not require modification to the operating system itself, which makes deploying it simply a matter of putting it on the App Store. This project extends latest explorations based on device proximity information or using dedicated devices [3], [4] by transporting actual data in the network. We tested iSNAC at the 61st Annual Meeting of the American Society of Human Genetics (ICHG/ASHG 2011) in Montreal. Since ICHG is typically a large convention, with over 10,000 attendants and over 3,000 posters every year, it is ideal for us to understand how the app functions in an electronically crowded environment.

II. The iSNAC Application

The iSNAC app has a map-centric user interface and runs on iOS 4.2 or later. It allows users to browse information about posters presented at the conference and facilitates commenting on the posters and sharing these comments in a peer-to-peer fashion. As it is used, it records how messages propagate among devices. At the end of the conference, the user can submit the collected data to a server side program.

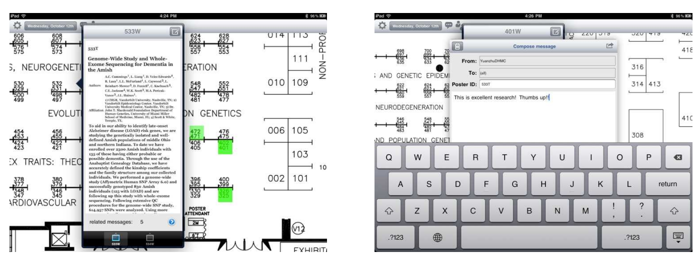

Fig. 1. iSNAC user interface

A. User interface

The center piece of the user interface is a map of the conference venue. It supports touch gestures including pinch zooming, panning, and tapping. The user can reposition the viewport by zooming in and out and by panning. The poster areas respond to tapping by presenting a pop-over window containing the poster's abstract. In addition, it also shows how many messages there are related to this poster (Fig. 1 left). It can also disclose a list of such messages for further reading. A user can compose a message about this particular poster from here (Fig. 1 right).

iSNAC has a toolbar for the user to access the Message Center and Peer List. The root level of the Message Center lists all user-generated messages and provides a message composition window. Messages composed here are generic and are not associated with a certain poster. The Peer List provides a list of which other users are on and off line. This allows the user to show or hide messages from a particular user on the map user interface.

Users are grouped by group names. For example, devices from our group share the same name of “MooreLab”. Other groups attending the conference can use their own group names to establish intra-group messaging. Messages are intended for all other members of the group regardless poster-specific or generic. Thus, communication here is one-to-all.

B. Messaging

The Bluetooth radio on-board of the iPad has a transmission range of up to 100 meters in a not-so-“noisy” environment. At conference venues, however, its range is a fraction of that. In general, two devices can be out of range of each other frequently, depending on where the users are and how they move. Thus, a device will “mechanically” transport messages from one location to another as its user moves around, i.e. the *store-carry-forward* paradigm. When the app detects other group members within range, it swaps messages with them. As a result, a message propagates among devices belonging to the same group until it reaches everybody. Messages delivered as such are “near-instant”.

A device sends two types of packets, “`iSend`” and “`iHave`”, to tell peers in range what they have and what they are providing, respectively. The payload of either of these packets is an array. For the case of an `iSend` packet, the array elements are full messages. For `iHave`, the array elements are digests of messages. The `iHave` packet’s creation and transmission are periodic at every 60 seconds. The `iSend` packet is created and transmitted after reception of an `iHave` packet from a nearby device. That is, it compares the received digests against the messages in its own storage. If missing messages are identified, they are put in an `iSend` packet for transmission. To avoid simultaneous transmissions of `iSend` in response to a single `iHave`, senders of `iSend` back off randomly by up to about a few seconds. In addition, after the user generates a new message, it is also transmitted in an `iHave` packet immediately.

A user-generated message is created when the user comments on a poster or sends a generic message. Furthermore, iSNAC also generates dummy *logbot* messages every 180 seconds for us to monitor how messages propagate at regular intervals. After a message of either type is created, it has multiple opportunities to be transmitted. The first time is right after it is generated. Subsequently, its digest is advertised in `iHave` packets periodically. When the app detects that a peer is missing a message that it has, it will transmit all such messages considered missing in an `iSend` packet.

III. Recorded Data

We deployed 12 iPads to ICHG and 9 of them collected useable data. Of our most interest is the time it took one message to propagate to all 9 devices. Fig. 2 is an example of how long it takes a series of periodic logbot messages from a given device to arrive at the 1st, 2nd, . . ., 9th device. Here, a vertical bar represents one message generated every 180 seconds. We can observe that a message usually arrives at the majority of the devices after a negligible delay.

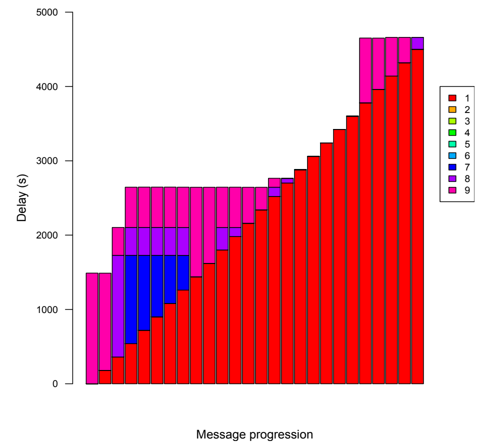

Fig. 2. Message dissemination delay

IV. Discussion

The iSNAC application has provided us a vehicle to study Mobile Ad Hoc and Delay-Tolerant Networking in noisy environments. Given that it is built on the unmodified iOS, it can be easily deployed on a large number of devices and generalized to other venues and types of events. We are interested in how it can be extended in a few aspects.

- Unicast – How do we send a message to a particular user in the group instead of all? The existing routing protocols in Delay-Tolerant Networking can be examined on this platform in such a real scenario.
- Heterogeneous mode – When the Wi-Fi network at the venue is unclogged from time to time, how do we utilize it, and how would this affect the performance of iSNAC?
- Scalability – Given the short range of Bluetooth radios relative to the scope of the conference venue, would more users contribute to better connectivity, and thus shorter delay, in the network?
- Inter-group cooperation – When a number of groups are at the same venue, can they cooperate in transporting messages for other groups? What kind of incentive is needed for them to consume their own power and network resources to help others altruistically?

We intend to port iSNAC to future iterations of ICHG to investigate these extensions to the application.

References

[1] A. Lindgren and P. Hui, “The quest for a killer app for opportunistic and delay tolerant networks (invited paper),” in *MobiCom Workshop CHANTS*, September 2009.

[2] “ICHGPoster: an iSNAC application of iPad,” 2011. [Online]. Available: http://itunes.apple.com/ca/app/ichg-posters/id470010424

[3] V. Vukadinovic and S. Mangold, “Opportunistic wireless communication in theme parks: A study of visitors mobility,” in *MobiCom Workshop CHANTS*, September 2011.

[4] P. Hui, A. Lindgren, and J. Crowcroft, “Empirical evaluation of hybrid opportunistic networks,” in *COMSNETS*, January 2009.