

Ad-hoc Social Network: A Comprehensive Survey

Dr. Sapna Gambhir, Nagender Aneja

Abstract – Ad-hoc Social Network (ASN) is a social network based on ad-hoc network. The paper describes the work done for implementation of ASN, to determine user experience while using ASN on mobile network, architecture of ASN and different profile matching algorithms. The paper also proposes some of research issues and challenges to be solved for ASN.

Index Terms – Ad-hoc Social Network, Routing, Profile, Middleware, Tuple Space, Network Directory, Location

----- ◆ -----
established.

1. Introduction

Ad-hoc Social Network (ASN) is a blend of social network and ad-hoc network. Today most of mobile phones are equipped with Bluetooth, Wi-Fi, and Cellular radio and so are capable of supporting ad-hoc communication mode. ASN is a social network where people of similar interests connect with each other using ad-hoc communication mode of mobile devices. ASN has its own advantages as it uses infrastructure-less network for communication. ASN is more meaningful as users are interested to know which of other users have similar interests and are within communication range so that they can either chat or personally talk on current interests.

Ad-hoc Social Networks (ASNs) have been proposed as anytime, anywhere social networks which use mobile ad-hoc network for its establishment and basic communication. Present social networking solutions suffers with disadvantage that they provide social networking in presence of Internet only. ASN is a solution to applications which provide anytime, anywhere social computing like sharing a cab at airport for a common destination. Mobile social networking is also an emerging area but it differs from Ad-hoc social networks in terms of optimized mobile ad-networks needed to support social applications for mobile devices like smart phones. Many middle-wares for mobile social networking have been proposed.

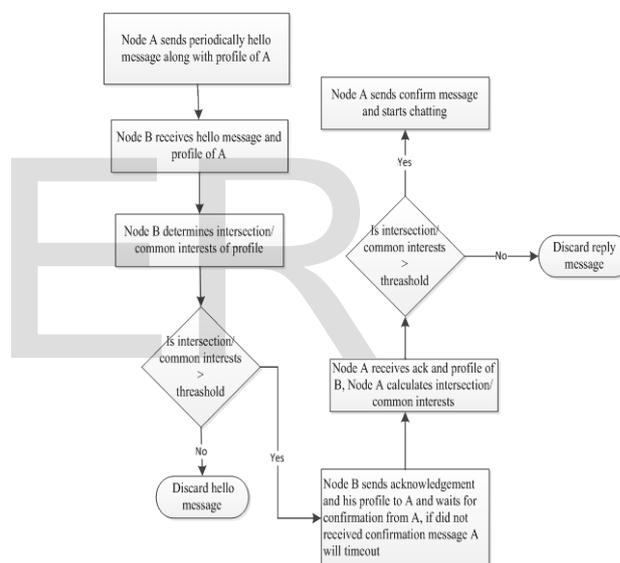


Figure 1: Elementary solution for profile matching in ASN

We conducted a survey of Ad-hoc Social Network to determine feasibility, user experience, and research challenges to form ASN.

2. Literature Survey

Sarigol et al. [1] presents AdSocial, a social platform that supports social network applications in an ad-hoc network. AdSocial targets small-scale scenarios such as friends playing a game on a train or co-workers sharing calendar information. AdSocial uses data piggybacking mechanism to share data by using any of many existing routing protocols for ad hoc networks without requiring any modification to the protocols themselves. Conference participants will be able to establish voice-video calls, chat, or play games while moving around thus configuring a mobile and multi-hop ad

Figure 1 describes a process as how ad-hoc social network is

- Dr. Sapna Gambhir is Associate Professor in Department of Computer Engineering in YMCA University of Science and Technology, Faridabad, Haryana, India. E-mail: sapnagambhir@gmail.com
- Nagender Aneja is a research scholar in Department of Computer Engineering in YMCA University of Science and Technology, Faridabad, Haryana, India. E-mail: naneja@gmail.com

hoc network. AdSocial was demonstrated in an ad hoc network of 10–15 Nokia N810 handhelds. Users of AdSocial can retrieve profile of nearby buddies in order to start chat or voice communication based on sip address. Information about nearby buddies such as their presence status, SIP address, or interests is retrieved directly from a web server of the corresponding buddy in the ad hoc network. To locate a buddy and to determine the address of a buddy's webserver, AdSocial uses MAND (Mobile Ad hoc Network Directory), a distributed directory service for ad hoc networks.

Sarigolet et al. [2] presented a distributed tuple space for social networking on ad hoc networks. The authors presented a novel approach for storing and searching data in ad hoc networks. The tuple space abstracts the underlying network as a common memory space in which nodes can store and lookup tuples. Each application configures its own "shared memory" rather than all tuples residing in a single shared memory. Moreover, an application can control the propagation of its tuples both in space and time. A presence application uses a tuple like the following to advertise the name of a buddy, its status, and IP address.

```
tuple-presence = {  
  key = "adsocial-presence"  
  value = "15.10.5.2:80 | Patrick | busy"  
  version = 32  
  owner = "15.10.5.2"  
  scope = 4  
  lifetime = 30  
}
```

The key is the identifier of the tuple and the value represents its content. The owner identifies node generating the tuple. The scope field specifies for how many hops at most the tuple must be disseminated in the network. The lifetime field specifies for how long a tuple is stored locally at a node. Finally, the version field specifies a replacement schema among tuples with the same key. Therefore, a tuple is uniquely identified by the pair < key; version >. The authors also describe primitives for insertion, deletion, and lookup of tuples.

The authors also used the tuple space to implement a buddy presence service that allows users to view all buddies in their proximity as well as search for buddies with specific interests. Such an application could be used, for instance, in a conference venue to allow people to set up discussion groups. Every user creates a profile that among personal information contains a list of interests. A conference attendee may, for example, express in his profile an interest for "social networks & information filtering". If another conference participant wishes to meet with researchers working in his field, he can initiate a search with some relevant keywords and retrieve a list of users with matching interests.

Dario et al. [3] emphasize need of shifting application focus from virtual to physical social space since physical proximity increases likelihood of social relationships. The authors proposed that success of anytime, anywhere social computing depends on middleware solutions that separate social-network management concerns from application requirements. Socially Aware and Mobile Architecture (SAMOA), integrates a set of common management facilities for personalizing location-dependent social networks, and for propagating social networks' visibility up to the application level. The authors proposed that building social-networking application on top of network layer can be tedious and error-prone because of issues like device mobility, intermittent connectivity, and availability. SAMOA supports creation of anytime, anywhere semantic context aware social network i.e. the logical abstractions that group together mobile users who are in physical proximity and share common affinities, attitudes, and social interests. SAMOA lets mobile users create roaming social networks that, following user movements, reflect at each instant all nearby encounters of interest. SAMOA roaming social networks center on a user (the ego user), and are based on two kinds of context visibility that is place visibility and profile visibility. SAMOA models represent context data in terms of semantic metadata (profiles) and exploits semantic matching algorithms for analyzing profiles and inferring potential semantic compatibility.

Maattila et al. [4] TWIN is an ad hoc social networking system which offers applications for social presence, mobile multimedia sharing and ad hoc community-based communication. The authors presented results of a large scale user study of TWIN. In this field study, 250 study participants used TWIN for nine weeks. The aim was to investigate user experience of social proximity-based ad hoc communication. It was found that the users felt TWIN to be more fun and entertaining than a useful tool for achieving pre-defined goals. Furthermore, users appreciated the possibility to find and chat with both familiar and unfamiliar persons nearby. Privacy concerns did not rise as a significant issue in user experience.

Seada et al. [5] presented social networks as a killer application for wireless ad-hoc networks. They proposed improvements and various components required in basic architecture for ad-hoc network that can support social network applications like boot strapper of social network, friend locator, location locator, security and privacy, and connectivity to the Internet.

Podobnik et al. [6] proposed that ad-hoc social networking as a platform in comparison to social networking as a service based on benefits, applications, and ease of building ad-hoc social network. They also discussed two different methods called open and closed for building ad-hoc social networks. In a closed ASN initiator's social network (e.g. Facebook) members are only the members of initiator's ad-hoc social network while in case of open ASN a non-SN member can also join the ad-hoc social network. A

registration of non-SN members must be approved by initiating user.

Gurecki [7] investigated Wi-Fi tethering software to configure G1 phone in ad-hoc wireless mode. He established an ad-hoc network with a laptop and developed a scheme for discovering other ad-hoc devices to provide social networking over ad hoc network. He also developed software for displaying GPS location and text messages received from other ad-hoc peers on G1 phone. The software was also able to transmitting & receiving photos (time permitting). Further, he adapted the discovery protocol to also discover other G1 phones on a WiFi network managed by an access point (time permitting).

Stuedi [8] proposed MAND (Mobile Ad-hoc Network Directory) a distributed system built on the top of network layer for the storing and the information in MANETs. Authors also built an application ADSocial on top of MAND that allows users to exchange messages, communicate common interest and set up VoIP calls in ad-hoc networks. They also claim that even if MAND is built on top of network layer but it does not require any modification to existing routing protocols. This approach restricts the reuse of ad-hoc support in different application domains. In addition, it can be error prone and tedious because it requires dealing explicitly with all issues related to user and device mobility, and availability.

Dodson et al. [9] proposed the Junction protocol which is an application-level communication protocol for ASNs on which applications of ASN can be developed. Junction is an application-level communication protocol and library designed for writing mobile applications for ad hoc groups without centralized application servers. Applications may be built using a generic switchboard service for communication. Each dynamic instance can designate a unique switchboard, hosted by a peer or by a third-party, for the session. Invitations to sessions can be carried over NFC, Bluetooth, or QR codes. Junction is available as open source for Android, iPhone and Javascript platforms. Over ten applications in multimedia sharing, games, communication, education, and for enhancing security for online transactions have been developed in Junction.

Li et al. [10] proposes a concrete, generalized, and novel framework to develop a fully functional mobile ad hoc social network (MobiSN). MobiSN proposes a semantics-based matchmaking scheme for discovering similar users. It uses ontology to analyse user profiles and application data that enables the inference of relationship and similarity between users and resources. The authors also proposed a Semantics-based Distance-Vector (SDV) query routing algorithm, in which a node can make routing decisions by knowing only its immediate neighbours and limited resource information.

Li et al. [11] proposed, semantics based social network (SMSN), in which mobile devices store user social profiles including their interests, and hobbies. Some data of the profile can be extracted from the applications being executed on mobile devices. By the analysis of the profiles, communities of users of similar interests are discovered and automatically constructed. The authors proposed semantics-based matchmaking that adds machine accessible semantics to the system. It uses ontology to analyse user profiles and application data that enables the inference of relationship and similarity between users and resources. SMSN allows exchange of heterogeneous social data without loss of meaning between various applications and/or user profiles of diverse types, and greatly improves the interoperability between diverse devices and data.

Lee et al. [12] proposed a system to create a user profile automatically based on collected data from a mobile device and build an Ad-hoc social network. The authors proposed a method of inferring a user's interests in real-time. User's interests are inferred by extracting meaningful words from recorded URLs. The authors used a Hierarchical model for interest inference. It consists of a set of nodes and directed edges. A node in the Hierarchical model has a keyword and a value. Keyword is the name of category extracted from URLs and value is degree of interest level. The system builds ad hoc social network based on inferred interests. In order to establish ad hoc social network, mobile device A broadcasts a neighbour discovery message including user's interests. When mobile device B receives neighbour discovery message from A, it calculate similarity between A and B and send response message if similarity is high. If mobile device A receives the response message, A and B form a virtual link to interact.

Chen et al. [13] discussed a file-sharing algorithm for an ad-hoc network consisting of 'n' fixed nodes. A node is allowed to input geographic distance parameter to limit number of hop counts in multi-hop ad-hoc network. Node that wishes to have a desired file initiates a request to its neighbouring nodes. Request is forwarded till it reaches a node that can serve the request. The serving than schedule according to the number of requests it received for the same file. It has been shown that if a serving nodes serve at most 6 nodes at a time than every node's request can be served in $sq(n)$ scaling factor.

Podobnik et al. [14] also proposed to extend social network model to ad-hoc social network to capture the interest of users physically located in a limited geographical area with common goal or similar dynamic interests.

He et al. [15] addressed the problem of message propagation in ad-hoc-based proximity mobile social networks (PMSNs). They suggested proximity-based search instead of using location of mobile user for establishing, information sharing and interaction among users of PMSN. To accomplish the aim using proximity search, each node keeps the received message and retransmit according to parameter times-to-

send (TTS) embedded in received message header. Node computes time for next transmission of same message and decrements TTS by one to control the termination of message propagation. When TTS reaches zero, it deletes the message from its message repository. A message filter component embedded with options of filtering rules like keywords, friend list and virtual credits is also suggested for reliable message delivery. Authors claim to obtain threshold value 6 of TTS for a network of 600 users over an area of 500X500 to deliver the message to more than 98% of the users.

Yan et al. [16] and Chen et al. [17] presented a framework for reputation management and generation system in ASNs. Using this framework (AdChat) a user can see the local reputation of other users during chatting. Sharing cost of some item in a shopping-mall to get discount, sharing taxi for a ride etc. are some application scenarios where AdChat can be used.

Yiu et al. [18] presented an application which locates friends in proximity based on given threshold Euclidean distance. They proposed that application tunes its proximity distance itself according to communication cost.

Dobson et al. [19] presented an algorithm for synchronization of nodes in ASNs while exchanging data for routing. Authors claimed that it can handle scalability of network up to 4096 nodes. They used GMAC (A gossiping TDMA MAC layer) to solve the problem.

Kohno et al. [20] discussed an application of ASNs for sharing photos which have been clicked in an ad hoc network. They proposed a mechanism which synchronize and share the photos among the members of ASN according to chronological order.

Rahnama et al. [21] worked on problem of efficient friend locating mechanism in ASN and extended the work on semantic based pattern matching via adding the context based adaptability. Algorithm based on the idea that as a user changes his social connection patterns, the suggestion for friends which he receives also change accordingly.

Campbell et al. [22] proposed to add sensing capability into social networking applications. They presented a system called CenceMe which collects the information about the neighbouring users and concise the facts which can be used in many applications.

3. Research Challenges and Conclusion

Current research in Ad-hoc Social Networks demonstrated that ad-hoc social network could be an extension to web-based social network or an independent social network formed for specific occasions. Users can detect and verify the proximity of their peers and they can actually know where they actually are within certain frequency and not

only where they claim to be. In certain situations, such as large events or high density, or crowded locations e.g. in a university centralized base station may not be able to handle the load and Ad-hoc network technology is the best way to help users to create a temporary social network.

There are many components to build this architecture starting from bootstrapping or joining a social network, sending or receiving messages for locating people with similar interests with localization aspects. In many scenarios such as large event or crowded spots, high density of nodes poses various challenges for ad-hoc network that have not been considered in past. High density of nodes means a very large neighbour list for mobile device with limited memory and limited processing power. Routing protocols based on flooding mechanism may cause significant overhead in the network. For these reasons, smarter mechanisms for connectivity and topology management are required.

The research also demonstrated that there is need to investigate interest-based matching and topology building. Entering users' preferences or interests manually is research problem and it should be automated. One of the solutions is to use users' browser history but that may not be sufficient. Users' may have different browsing pattern at different geographic locations and creating a single global user profile may not help user to get good friend suggestions.

It is important that intermediate nodes cooperate in processing and forwarding requests and replies despite the overhead. We know from our experience that users are willing to cooperate as long as they are getting some value or incentives for being part of the network. A distributed authentication mechanism is required to verify a node that is actually a friend and has similar interest before any data is shared between nodes.

One of the harder problems is to prevent intermediate nodes and neighbours from knowing a user's friends and communication pattern.

In ad-hoc social network, users are not interested in a reply about any node but about nodes that satisfies certain proximity conditions. Therefore, a routing or matching algorithm should interpret location and distance between nodes or may be direction also.

4. References

- [1] E. Sarigol, O. Riva, P. Stuedi and G. Alonso, "Enabling social networking in ad hoc networks of mobile phones," Proc. VLDB Endow, vol. 2, no. 2, pp. 1634-1637, August 2009.
- [2] E. Sarigol, O. Riva and G. Alonso, "Tuple Space for Social Networking on Mobile Phones," in IEEE International Conference on Data Engineering, California, USA, 2010.

- [3] B. Dario, M. Rebecca and T. Alessandra, "Context-Aware Middleware for Anytime, Anywhere Social Networks," *IEEE Intelligent Systems*, vol. 22, no. 15, pp. 23–32, 2007.
- [4] Vaananen-Vainio-Mattila, P. Saarinen, M. Wäljas, M. Hännikäinen, H. Orsila and N. Kiukkonen, "User Experience of Social Ad Hoc Networking: Findings from a Large-Scale Field Trial of TWIN," in *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia*, Limassol, Cyprus, 2010.
- [5] K. Seada and C. Perkins, "Social Networks: The Killer App for Wireless Ad Hoc Networks?," *Nokia Research Center -Technical Report*, August 2006.
- [6] V. Podobnik and I. Lovrek, "Transforming social networking from a service to a platform: a case study of ad-hoc social," in *Proceedings of the 13th International Conference on Electronic Commerce (ICEC '11)*, New York, USA, 2012.
- [7] D. Gurecki, "Ad-hoc Social Networking for the Google Android Platform," <http://www4.ncsu.edu/~dwgureck/csc714/>, Raleigh, NC, USA, 2009.
- [8] P. Stuedi, O. Riva and G. Alonso, "Demo abstract ad hoc social networking using MAND," <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.189.523&rep=rep1&type=pdf>, 2008.
- [9] B. Dodson, A. Cannon, T.-Y. Huang and M. S. Lam, "The Junction protocol for Ad Hoc Peer-to-Peer Mobile Applications," *Stanford University*, USA, December 2010.
- [10] J. Li and S. U. Khan, "MobiSN: Semantics-based Mobile Ad Hoc Social Network Framework," in *Proceedings of the 28th IEEE conference on Global telecommunications*, Piscataway, NJ, USA, 2009.
- [11] J. Li, H. Wang and S. U. Khan, "A Semantics-based Approach to Large-Scale Mobile Social Networking," *Mob. Netw. Appl.*, vol. 17, no. 2, pp. 192–205, 2012.
- [12] J. Lee and C. S. Hong, "A Mechanism for Building Ad-hoc Social Network Based on User's Interest," in *13th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Taipei, Taiwan, 2011.
- [13] Y.-T. Chen, C. Caramanis and S. Shakkottai, "On Sharing Viral Video over an Ad Hoc Wireless Network," *CoRR*, vol. abs/1101.5088, 2011.
- [14] V. Podobnik and I. Lovrek, "An agent-based platform for ad-hoc social networking," in *Proceedings of the 5th KES international conference on Agent and multi-agent systems: technologies and applications*, Manchester, UK, 2011.
- [15] W. He, Y. Huang, N. K. and B. Wu, "Message propagation in ad-hoc-based proximity mobile social networks," in *IEEE 8th International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, Mannheim, Germany, 2010.
- [16] Z. Yan and Y. Chen, "AdChatRep: a reputation system for MANET chatting," in *Proceedings of 1st international symposium on From digital footprints to social and community intelligence*, New York, USA, 2011.
- [17] Y. Chen, Z. Yan and V. Niemi, "Implementation of a Reputation System for Pervasive Social Networking," in *IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Washington, DC, USA, 2011.
- [18] M. L. Yiu, L. H. U, S. Salteris and K. Tzoumas, "Efficient proximity detection among mobile users via self-tuning policies," *VLDB Endowment*, vol. 3, no. 1-2, pp. 985–996, September 2010.
- [19] M. Dobson, S. Voulgaris and M. Van Steen, "Network-level synchronization in decentralized social ad-hoc networks," in *5th International Conference on Pervasive Computing and Applications (ICPCA)*, Maribor, 2010.
- [20] M. Kohno and J. Rekimoto, "Searching common experience: a social communication tool based on mobile ad-hoc networking," in *Proceedings of the 7th international conference on Human computer interaction with mobile devices & services (MobileHCI)*, Salzburg, Austria, 2005.
- [21] H. Rahnama, A. Madni, A. Sadeghian, C. Mawson and B. Gajderowicz, "Adaptive context for generic pattern matching in ad hoc social networks," in *3rd International Symposium on Communications, Control and Signal Processing (ISCCSP)*, St Julians, 2008.
- [22] A. T. Campbell, S. B. Eisenman, K. Fodor, N. D. Lane, H. Lu, E. Miluzzo, M. Musolesi, R. A. Peterson and X. Zheng, "CenceMe: Injecting Sensing Presence into Social Network Applications using Mobile Phones (Demo Abstract)," in http://www.sigmobile.org/mobihoc/2008/download/musolesi_demo.pdf, Hong Kong, May 26-30, 2008.