Virtual Cellular ISPs

Talal Ahmad
New York University
ahmad@cs.nyu.edu

Lakshminarayanan Subramanian
New York University
lakshmi@cs.nyu.edu

ABSTRACT

Achieving reliable cellular connectivity in rural areas is a challenge for several reasons: lack of economic incentives for telecom providers; high deployment and maintenance costs; low purchasing power; poor and unreliable power infrastructures; poor backhaul support; and lack of ground expertise. This paper proposes the design, implementation and deployment of the *Virtual Cellular ISPs (VC-ISP)*, a ground-up cellular network architecture that enables third-party vendors to offer cellular coverage and mobile services to rural areas in an extensible manner. Unlike conventional monolithic cellular network architecture, the VC-ISP model enables a distributed (and potentially disjointed) collection of open, programmable cellular base stations to communicate using a novel intermittency-aware naming and addressing mechanism. This system enables VC-ISPs to provide reliable service with graceful recovery from failures under harsh network conditions. VC-ISP offers support for deploying local applications. Here, we demonstrate the technical viability of the VC-ISP model using a real-world 3-node VC-ISP deployment, including installation of a solar-powered cell tower in rural Ghana and additional emulation experiments in controlled environments.

CCS CONCEPTS

•**Networks** → Network reliability;

KEYWORDS

Cellular Network, Naming and Addressing, Reliability

ACM Reference format:
Talal Ahmad and Lakshminarayanan Subramanian. 2017. Virtual Cellular ISPs. In *Proceedings of SMARTOBJECTSfi17, October 16, 2017, Snowbird, UT, USA.*, , 6 pages.
DOI: http://dx.doi.org/10.1145/3127502.3127515

1 INTRODUCTION

Cellular networks have achieved significant penetration levels in developing regions during the past decade. Much of this growth has been predominantly urban-centric with relatively low rural presence [9, 10, 23], and unfortunately existing cellular connectivity model is not economically viable for rural settings for a multitude of fundamental challenges in rural contexts:

SMARTOBJECTS'17, October 16, 2017, Snowbird, UT, USA.

DOI: http://dx.doi.org/10.1145/3127502.3127515

Power: Rural regions lack stable and reliable grid power. Cellular networks are inherently power hungry in rural settings, and seek to blanket large areas, often relying on diesel-powered generators for power, which is a highly expensive proposition [4].

Lack of Financial Incentives: While cellular networks incur high cost of capital and operational expenses, the demand often does not match the cost due to low user-densities and low purchasing power. For years, low demand and high infrastructure costs have been locked in a vicious cycle. Bharti Airtel and Reliance Communications, two of the largest cellular service providers in India, announced their intent to pull out of rural markets due to unfavorable economics [2].

Reliability: Maintenance of rural, wireless networks is difficult due to a myriad of reliability issues, power-related problems, and a lack of local expertise to solve them. Lack of clean power is known to frequently trigger device failures in rural networks [23].

In this paper, we present the design, implementation and deployment of *Virtual Cellular ISPs (VC-ISP)*, a new ground-up cellular network architecture that enables third-party vendors to offer cellular coverage and mobile services to rural areas in an extensible manner. The basic building of the VC-ISP network architecture is a Virtual Cellular Node (VCN) which are open, programmable cellular base station platforms [1, 14, 21] that enable easy deployment of low-power software defined base stations that function efficiently over an IP backplane. Given that VC-ISPs are designed for areas with limited cellular coverage, the second building block of the VC-ISP model is the use of highly directional backhauls to extend backhaul connectivity to the VCNs. We assume that a backhaul is available for this paper and built the communication mechanism used by the VCNs to form a cellular network.

The VC-ISP network architecture is built upon on the existing work on community cellular networks[3, 5, 12, 24], which can be viewed as stand-alone installations of virtual cellular nodes. In VC-ISP architecture, we aim to address the following question: *How do we enable a distributed and potentially disjointed collection of open, programmable cellular base stations to act in unison as a single virtual cellular ISP?* Given the lack of incentives for conventional cellular providers to provide rural connectivity, the VC-ISP model empowers third-party providers to build a cellular network comprising of several distributed VCNs connected only by an Internet backhaul to the cloud. In addition, due to the programmable nature of individual VCNs, we show how VC-ISPs can provide new forms of distributed mobile edge services where computation, storage and application state can be moved to the mobile edge.

We demonstrated the effectiveness of the VC-ISP model using a real-world 3-node VC-ISP deployment system, which includes a solar-powered cell tower installation in rural Ghana. Our Ghana based installation also supports different types of distributed mobile services (as outlined earlier) that have been used by rural users within the community. We also evaluated the effectiveness of specific aspects of the VC-ISP architecture using emulation and simulation experiments. In summary, we believe that VC-ISP represents a radical departure from the conventional approach of designing cellular networks, and could enable ground-up innovation, allowing several budding rural entrepreneurs to launch competing cellular network services in rural localities while seamlessly co-existing with existing operators.

2 RELATED WORK

Software-based microcells: Universal service obligations have been pushed by many countries to facilitate the expansion of communication into rural areas. At the same time, recent advances in hardware and open-source software has made available inexpensive cellular equipment broadly accessible. For example, OpenBTS, an open-source GSM base transceiver station (BTS) implementation has enabled a wide range of projects aimed towards building small-scale VCNs. Heimerl et al.[12] demonstrated the viability of independently run, locally operated cellular networks. Similarly, Rhizomatica[3] has deployed several community-run cellular networks in Oaxaca, Mexico with a short-term experimental spectrum license. Zheleva et al. [24] deployed a similar system for purely local communications in Zambia.

Intermittency aware networks: VC-ISP significantly benefits from prior work on Delay Tolerant Networks [13, 22]. One of the first real world DTN deployments was DakNet which provides low-cost digital communication to rural areas using physical transportation links [19]. There have been several DTN-based routing algorithms like Encounter based routing [18] and MobySpace [17]. [6, 7, 15] also use DTN centric "store and forward" algorithms to progressively forward messages until they reach their destination. The intermittency awareness abstraction of VC-ISP is fundamentally different from the traditional DTN abstractions since VC-ISP is primarily tailored and designed for cellular network context and primarily focuses on higher application-centric "store and forward" abstractions; the DTN routes for VC-ISP messages are preset by the network structure which changes infrequently.

3 VC-ISP ARCHITECTURE

VC-ISP is a new model for designing and rethinking the evolution of cellular network providers. The network architecture comprises of a distributed collection of software-defined VCNs powered by emerging open-source cellular base-station platforms, which together provide the abstraction of a single virtual network with the entire gamut of features offered by a traditional cellular provider. Using this architecture, we wish to extend coverage beyond existing boundaries, moving a wide variety of services to the extreme edge. The VC-ISP network architecture aims to enable third-party vendors to offer cellular ISP services in a decentralized manner without requiring modifications to mobile devices, and allowing cooperation with traditional cellular networks.

Conventional cellular networks continue to use highly inflexible and expensive platforms with complex control-plane protocols. This makes the network nearly impossible to configure or manage. On the contrary, VCNs enable us to program and easily integrate a wide range of other middlebox services and functionalities. The

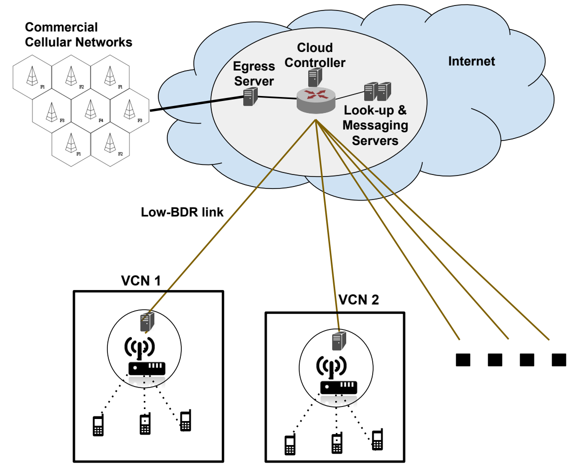

Figure 1: VC-ISP Architecture

strawman architecture is illustrated in Figure 1; a distribution of VCNs provide access to a geographically disjoint set of regions and collectively provides the abstraction of an unified VC-ISP. We envision all of the nodes to be connected to the Internet using various types of backhaul technologies such as wireless backhauls, satellite links, balloons [11], leased lines, telephony, fiber etc. Owned by a single party, a centralized cloud controller would interconnect all nodes and is in charge of administering this cohesive network. This controller resides at an appropriately placed geographic location to normalize communication latency. To achieve such an architecture, we discuss design specifications and the high level goals we wish to achieve. To summarize, the architecture should be able to:

- *Goal 1:* Operate in regions where there is high latency, low bandwidth and intermittency.
- *Goal 2:* Reduce processing at the core and migrate relevant application processing to the edge to further reduce latency.
- *Goal 3:* Avoid placing heavy reliance on conventional sources of power (including fossil fuels).

Addressing these goals would require us to tackle several research challenges. The first and foremost challenge involves the exploration of a variety of operational conditions. Additionally, majority of developing regions have poor cloud infrastructure and a weak backhaul, resulting in low Bandwidth-Delay Ratio (BDR) regimes. Traditionally designed applications running in these regimes face several issues including frequent timeouts, operational bandwidth insufficiency and various other inconsistencies. Dealing with these issues and obtaining *Goal 1* requires rethinking application requirements. In VC-ISP mobile clients are connected to a *local* server placed in the proximity of the base station. This local server has complete autonomy in decision-making for a certain class of events but also must abide to directives from the central controller. Therefore, the local server ensures greater responsiveness to clients. The VC-ISP will provide a separate intermittency-aware substrate where all messages are tagged with session and communication group specific identifiers that enable nodes to store and deliver messages to their corresponding recipients in a delay-tolerant manner. Mobile applications in VC-ISP can also leverage this substrate to specify how to handle application specific messages in an intermittency condition.

This centralized cloud infrastructure provides mechanisms for the now unified (nodal) network to communicate with non-member networks. This infrastructure comprises of 2 key components:

- *Egress nodes*, which are responsible for forwarding traffic between either VCNs or between VCNs and the conventional cellular network.
- *Lookup and messaging servers*, which play a critical role in enabling the VC-ISP to offer connectivity and a variety of cellular services across nodes.

Each VCN in the network operates independently and communicates with other VCNs using the egress nodes, and the lookup and messaging servers. We partially obtained *Goal 2* as the local servers helped facilitate this process in most cases because they have a cached version of the naming and addressing table. Because these servers also have administrative autonomy, they are able to act on the behest of the centralized controller, and can provide additional services for applications at the extreme edge. Different types of edge services will be explained in subsequent sections.

The modular nature of the various components required to build the VC-ISP ensures minimal power consumption. Therefore, we obtain *Goal 3* by predominantly powering the nodes using solar panels. By building a compact, mobile base station unit, we are able to avoid the reliance on the unstable grid power in these regions.

4 NAMING & ADDRESSING

To successfully utilize the various services provided by cellular networks, even from outside the network, it is imperative to have an efficient scheme for device naming and addressing. We wished to achieve three goals using our identity management service, namely (a) flexibility in identities held, (b) seamless user experience, and (c) uniqueness.

4.1 Flexibility

A user can have an authenticated identity which permits it to use services offered by the VC-ISP. However, this identity should not facilitate communication with external users (outside the VC-ISP). Similarly, a user can have another identity whose sole purpose is to enable communication with the outside world. To this end, the VC-ISP issues two types of identities: *(a) Local identities* that are only usable/valid inside the VC-ISP framework, and *(b) Global identities* which make it possible for devices outside the framework to connect with those inside. It is the duty of the identity management service to multiplex the functionalities accordingly based on the nature of the identity.

Users with local identities can communicate with others within the same VCN or in the same VC-ISP. In contrast, global identities provide all the functionalities supported by local identities and enable connection both to and from outside the VC-ISP. The importance of this form of flexibility stems from the fact that providing access to an identity outside the network has an associated cost. Globally identifiable names need to be purchased (or leased on a monthly basis) from organizations selling VoIP services, like Twilio.

4.1.1 Generation & Dissemination. Unique identifiers are maintained, regulated and generated (sequentially, in the order in which requests are received) by a centralized server separate from the controller, henceforth referred to as a name server. These identifiers (or names) are flat and carry no location information. The flexibility offered in naming can best be utilized when the architecture is optimally designed. To achieve the required performance, we propose two different implementations, namely:

1. Server-Client Model: A single, centralized name server issues both global and local identities as requested. If a new users enters any node, the registration requests are routed to this name server in the order of entrance. In the case of a cellular device that connects through the OpenBTS interface, the request to generate a name comprises of the IMSI number (which is globally unique and non-mutable). This information is then used in the standard GSM challenge-response protocol, when the phone associates with the network. The response from the central name server is a unique identifier. This server keeps track of each identity issued (name) and its current location (address), maintaining consistent state information. This information is required to route calls and messages correctly. In addition, it records the IMSI number of the SIM card to which the identity was issued. If the user moves to a different node, the server still knows that an identity has been issued and only the location of the user needs to be updated. To reduce communication latency, this information is replicated at the local servers at a nodal level and updated on occurrence of a cache-miss.

2. DHT-based Model: In this model name/location mapping requests are answered by a cloud controller chosen by a predefined hash function. These cloud controllers collectively form a DHT for managing identities and application level data storage. This DHT-based model used by several authors [8, 16] deals with failures and provides faster access even when there is a skew in users between different nodes by evenly dividing storage and processing.

Intermittency aware lookup: As explained earlier, the VC-ISP architecture provides an eventually consistent view of the entire name space to the local server at a nodal level. If a user wants to connect with another user, it first contacts the local server of the node it currently is in. On occurrence of a cache-miss at this node, the required information is retrieved using suitable cache-update protocols [20] in a lazy manner. In the absence of a backhaul to the central name server, information at the local server at the nodal level is not consistent but once the backhaul is available consistency is achieved eventually.

4.2 Seamless User Experience

In both of the above mentioned models, naming and addressing information is cached at the local servers at each node. Though not complete, caching ensures a reduction in communication latency in the case of lookups, therefore improving the end-user experience.

Interoperability: It is crucial for the architecture to support the following: (a) communication within the VC-ISP, (b) communication across VC-ISPs, and (c) communication between the VC-ISP and other non-member networks (in both directions). Communication with the architecture has been explained in previous sections, where information obtained from the combination cache hits and misses on local servers and the centralized name server suffices to locate a given identity and ensue communication. Communication across VC-ISPs necessitate a federation manager, an entity responsible for aggregating both names and addressing information across all VC-ISPs. At the expense of a lookup to this manager, members of a particular VC-ISP can locate users at another VC-ISP. In our current hierarchy, the federation manager is located at a level above the cloud controller and central name server in order to provide a better view of the organization below. To better understand the third case, consider a scenario where a client with a global identity wishes to contact someone outside the node who is not part of the VC-ISP i.e a contact in a non-member network. Upon entering the contact's information, the identity management system in the local server generates a cache miss. A subsequent check of the central name server also results in a miss as the contact's naming and addressing information belongs to a third party telecommunication organization. On observing both misses, the local server redirects the communication request to the cloud server, which further forwards this request to the VoIP service provider from whom the global identity was purchased. It is now the responsibility of the service provider to appropriately route the communication request. Thus, any client with a global identity can communicate with a contact outside the VC-ISP architecture without entering any additional information and vice-versa. It is important to note that clients with local identities can effectively communicate with other clients within the VC-ISP using a similar mechanism of cache and central name server lookup to locate the address of an identity. The key difference is that the calls are routed within the VC-ISP architecture using OpenBTS and not a VoIP service provider.

5 IMPLEMENTATION AND EVALUATION

In this section, we describe in greater detail how we implemented different parts to evaluate performance of the VC-ISP system.

5.1 Cellular And VC-ISP Simulator

We simulated a 3-level VC-ISP hierarchy in Python, to compare and contrast its performance with conventional cellular networks in the case of random failure events. The first level in our hierarchy was the cloud controller, which is assumed to never fail. The second and third levels are regular nodes (VCNs). This can be visualized as a tree starting at the cloud controller.

Simulation Procedure: The nodes at the second level contribute 60% of the simulated traffic, while the nodes at third level make up the remaining 40%. At periodic time intervals, random failure events occur, comprising of link failures between the two nodal levels or link failures between a node and the cloud. Thus, we are able to estimate the traffic that is dropped because of the corresponding failure event and the component it affects. Using Monte Carlo simulations, we were able to plot the average amount of failed/dropped traffic. This experiment was repeated after varying the number of nodes in various levels of the hierarchy. In the case of conventional cellular networks, it is assumed that the serving gateways and packet gateways are located in the cloud, and traffic is a value proportional to the total number of nodes.

In addition to this simulator, we implemented the two identity management models to compare them across the time taken to converge in issuing identities with different volumes of user activity.

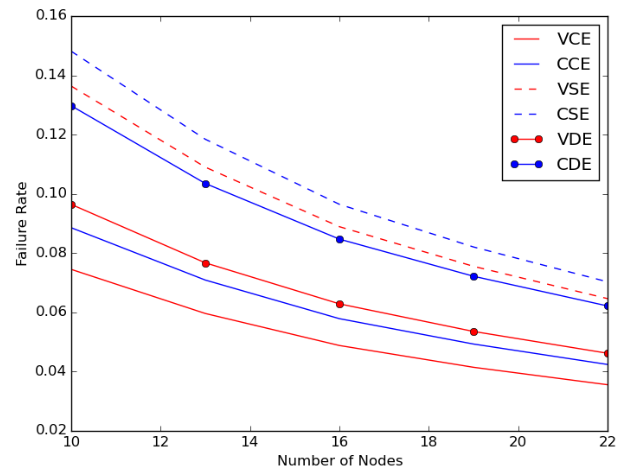

Figure 2: VCISP vs Conventional Cellular

Metric	Definition
VCE	VC-ISP Call Error
CCE	Cellular Call Error
VSE	VC-ISP SMS Error
CSE	Cellular SMS Error
VDE	VC-ISP Data Error
CDE	Cellular Data Error

Table 1: Metrics used in Figure 2 and their definitions.

We have created a 10-node AWS deployment with nodes in different geographic localities to simulate multiple cloud controllers. The locations of the AWS datacenters include California, Oregon, Virginia, Singapore, Frankfurt, Sydney, Tokyo, Ireland and Sao Paulo. Each of the nodes run a Python-based HTTP server, and HTTP `GET` requests are used to simulate inter-node communication.

5.2 Simulator Evaluation

We discuss the results of the implementation in §5.1, where Figure 2 contrasts VC-ISP and conventional cellular networks. The latter have serving gateways and packet gateways far from the actual base stations. Once the link to the base station is down everything breaks. In the analogous case, while the link between the cloud controller and the node is down, the node continues to provide local access. Therefore VCE, VSE and VDE (Table 1) values are always lower than their corresponding cellular counterparts. This general trend suggests that lower error rates are a consequence of a larger number of sites and users. This increases the total calls, ergo reducing the impact of a single failure (simulated within each time interval).

The box and whisker plot in Figure 3 compares the client-server model to the DHT-based model in terms of issuing a new global identity to a user. Issuing a new global identity entails issuing a serialized HTTP query to Nexmo. In the DHT-based model, the

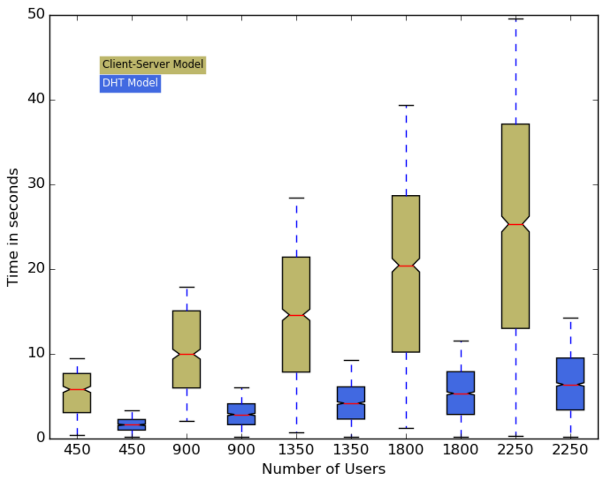

Figure 3: Latency for issuing identities: Single server vs 10-node DHT

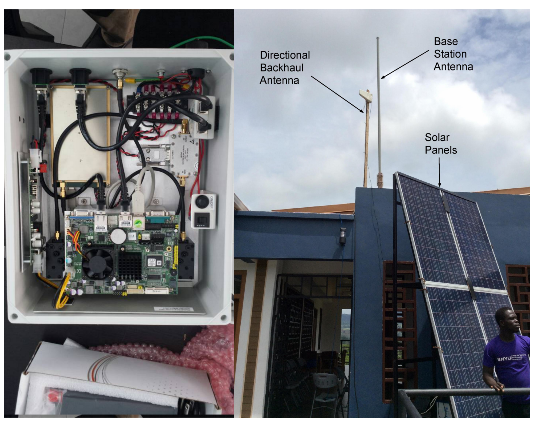

Figure 4: Our VCN in Kumawu Ghana. This VCN was deployed in a region at the edge of existing commercial cellular coverage and was connected to other VCNs using the naming and addressing mechanism explained above. It was also powered by solar panels.

requests were generated by multiple processes in the node at Frankfurt while other nodes were responsible for processing the request and forwarding it to Nexmo. A synthetic hash function evenly divides the requests to the corresponding nodes that process it, without any location bias. In the client-server model, similarly generated requests were both initiated and responded to by the local host server on the Frankfurt node. It can be seen that the DHT outperforms the single server, even when the number of users is low. This is true as the cumulative processing power in a 10-node DHT exceeds that of a single node server.

6 DEPLOYMENT

We built and evaluated a VC-ISP with three nodes, located in Kumawu (Ghana), New York City (USA) and Abu Dhabi (UAE).
VCNs: We used two hardware platforms as Base Transceiver Station (BTS) nodes. The first was a 1 Watt RapidCell from Range Networks [21], and the second was a 10 mWatt USRP B100 from Ettus Research. We positioned one USRP B100 BTS in Abu Dhabi, another in New York, and finally positioned the RapidCell in Kumawu, Ghana (Shown in Figure 4). Nodes ran OpenBTS along with smqueue and sipauthserve which deal with SMS and user authentication respectively. Freeswitch was used for routing calls, messages and data. We placed a PC next to the BTS setup and referred to it as the local server. This PC hosted the local cache. It is important to note that even without this PC, the OpenBTS combined with smqueue, sipauthserve and Freeswitch is able to provide the functionality of a VCN.
Cloud Controller: We used an AWS EC2 instance hosted in Frankfurt, because the location is roughly equidistant from New York, Abu Dhabi and Kumawu. The instance had a memory of 1 GiB and was allocated a single vCPU. Our EC2 instance was running Ubuntu 14.04, and all application support on the cloud was written using Python. The specs of the instance suffice for application support as it met the minimal required processing.
Experience: We successfully used our 3-Node VCN to conduct calls and text messages between mobile devices connected to two different VCNs with acceptable latency and decent call quality. We also conducted several calls and SMS to users connected to conventional cellular networks. As explained earlier for calls and SMS routing, a local cache was first checked and when the location was not in the same VCN, the call was then routed to the cloud controller. The cloud controller then routed the calls to the appropriate VCN. In cases where the network was not available the text messages and voice messages were saved locally at the node until the network was available.

7 ACKNOWLEDGEMENTS

We thank Prof. Yaw Nyarko, Zohaib Jabbar, Fareeha Amjad, Kessir Adjaho, Alfred Afutu, Tiffany Tong, Yasir Zaki, Aditya Dhananjay and other members of the Center for Technology and Economic Development (CTED) in Abu Dhabi and CTED Kumawu members in Ghana for their help in this work. We also thank the anonymous reviewers for their reviews which helped in improving this paper. This work was supported by the NYU Abu Dhabi Research Institute and the Center for Technology and Economic Development in NYU Abu Dhabi.

REFERENCES

[1] Openbts. www://www.openbts.org/.
[2] RCom, Bharti seek early exit from rural telephony scheme. In *The Economic Times (02/07/2011)*.
[3] Rhizomatica community base station. http://rhizomatica.org/projects/community-basestation/. Accessed: 2015-11-05.
[4] T. Ahmad, S. Kalyanaraman, F. Amjad, and L. Subramanian. Solar vs diesel: where to draw the line for cell towers? In *Proceedings of the Seventh International Conference on Information and Communication Technologies and Development*, page 7. ACM, 2015.
[5] A. Anand, V. Pejovic, E. M. Belding, and D. L. Johnson. Villagecell: Cost effective cellular connectivity in rural areas. In *Proceedings of the Fifth International Conference on Information and Communication Technologies and Development*, ICTD '12, pages 180–189, New York, NY, USA, 2012. ACM.

[6] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networks. In *INFOCOM*, volume 6, pages 1–11, 2006.
[7] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Impact of human mobility on opportunistic forwarding algorithms. *Mobile Computing, IEEE Transactions on*, 6(6):606–620, 2007.
[8] A. Dhananjay, M. Tierney, J. Li, and L. Subramanian. Wire: a new rural connectivity paradigm. In *ACM SIGCOMM Computer Communication Review*, volume 41, pages 462–463. ACM, 2011.
[9] V. Gabale, A. Chiplunkar, B. Raman, and P. Dutta. Delaycheck: Scheduling voice over multi-hop multi-channel wireless mesh networks. In *COMSNETS*, 2011.
[10] V. Gabale, B. Raman, K. Chebrolu, and P. Kulkarni. Lit mac: Addressing the challenges of effective voice communication in a low cost, low power wireless mesh network. In *Proceedings of the First ACM Symposium on Computing for Development*, ACM DEV '10, pages 5:1–5:11, New York, NY, USA, 2010. ACM.
[11] Google Inc. Project loon. http://www.google.com/loon/.
[12] K. Heimerl, S. Hasan, K. Ali, E. Brewer, and T. Parikh. Local, sustainable, small-scale cellular networks. In *Proceedings of the Sixth International Conference on Information and Communication Technologies and Development: Full Papers-Volume 1*, pages 2–12. ACM, 2013.
[13] S. Jain, K. Fall, and R. Patra. Routing in a Delay Tolerant Network. In *ACM SIGCOMM*, pages 145–158, New York, NY, USA, 2004. ACM.
[14] X. Jin, L. E. Li, L. Vanbever, and J. Rexford. Softcell: Scalable and flexible cellular core network architecture. In *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, pages 163–174. ACM, 2013.
[15] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein. Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet. In *ACM Sigplan Notices*, volume 37, pages 96–107. ACM, 2002.
[16] C. Kim, M. Caesar, and J. Rexford. Floodless in SEATTLE: A Scalable Ethernet Architecture for Large Enterprises. In *SIGCOMM '08*.
[17] J. Leguay, T. Friedman, and V. Conan. Evaluating mobility pattern space routing for dtns. *arXiv preprint cs/0511102*, 2005.
[18] S. C. Nelson, M. Bakht, and R. Kravets. Encounter-based routing in dtns. In *INFOCOM 2009, IEEE*, pages 846–854. IEEE, 2009.
[19] A. Pentland, R. Fletcher, and A. Hasson. Daknet: Rethinking connectivity in developing nations. *Computer*, 37(1):78–83, 2004.
[20] T. R. Puzak. Analysis of cache replacement-algorithms. 1985.
[21] Range Networks. Range networks. http://www.rangenetworks.com/.
[22] A. Seth, D. Kroeker, M. Zaharia, S. Guo, and S. Keshav. Low-cost communication for rural internet kiosks using mechanical backhaul. *Proceedings of the 12th annual international conference on Mobile computing and networking*, pages 334–345, 2006.
[23] S. Surana, R. Patra, S. Nedevschi, M. Ramos, L. Subramanian, Y. Ben-David, and E. Brewer. Beyond Pilots: Keeping Rural Wireless Networks Alive. *Proceedings of NSDI 2008*, 2008.
[24] M. Zheleva, A. Paul, D. L. Johnson, and E. Belding. Kwiizya: Local cellular network services in remote areas. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '13, pages 417–430, New York, NY, USA, 2013. ACM.