CrowdITS: Crowdsourcing in Intelligent Transportation Systems

Kashif Ali, Dina Al-Yaseen
School of Computing,
Queen's University,
Kingston, ON, Canada
{kashif, alyaseen}@cs.queensu.ca

Ali Ejaz, Tayyab Javed
ECE Department,
Queen's University,
Kingston, ON, Canada
{ali.ejaz, 9tj9}@queensu.ca

Hossam S. Hassanein
School of Computing,
Queen's University,
Kingston, ON, Canada
hossam@cs.queensu.ca

***Abstract*—Intelligent Transportation Systems (ITS) and their applications are attracting significant attention in research and industry. ITS makes use of various sensing and communication technologies to assist transportation authorities and vehicles drivers in making informative decisions and provide leisure and safe driving experience. Data collection and dispersion are of utmost importance for the proper operation of ITS applications. Numerous standards, architectures and communication protocols have been anticipated for ITS applications. However, existing schemes are based on assumption that vehicles and roadside devices are equipped with sensing and communication capabilities. One of the major gaps of these approaches is their inability to capture events that can easily be logged by drivers using their mobile phones. In this paper, we propose to fill the gap by the use of Crowdsourcing in ITS namely, CrowdITS. In CrowdITS human inputs, along with available sensory data, are collected and communicated to a processing server using mobile phones. The basic idea is to use the Crowd with smart mobile phones to enable certain ITS applications without the need of any special sensors or communication devices, both in-vehicle and on-road. Alternatively, we integrate and aggregate human inputs with multiple information sources, and then selectively disseminate the aggregated information based on the driver's geo-location. Conceptually, the major change is to integrate human inputs, with multiple information sources, aggregate and finally it is localized according to the driver's geo-location. We describe the design of CrowdITS, report on the development of key ITS applications using Android and iPhone mobile phones, and outline the future work in the development of crowdsourced-based applications for intelligent transportation systems.**

I. Introduction

[1] The projected estimate of 8.5 million driving-related deaths by the year 2020 is one of the motivations behind numerous academic, commercial and governmental engagements in adapting technology for transportation industry. Communication and sensing technology used in the transportation industry is widely termed as Intelligent Transportation System (ITS). Technology made its way into ITS in the forms of Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) paradigms [1]. These paradigms rely on certain communication and sensing technology, e.g., WiFi, WiMax, short-range radar, active RFID tags, etc.

Inter-vehicle and vehicle-to-infrastructure communications are two diverse but complimentary paradigms to acquire events information from outside sources. The first scheme is the vehicular ad-hoc wireless communication and involves only vehicles, i.e., the V2V model. The second scheme is the vehicles to infrastructure and involves fixed infrastructure-based wireless communication, i.e., the V2I model. Application of these schemes, and ITS in general, include real-time safety and convenience applications (forward collision warning, blind spot detection, etc.) and non real-time (or near real-time) management and leisure applications (congestion avoidance, traffic reporting, etc.). In this paper, our research focus is on the non-real time ITS applications.

Information collection and dispersion are key to non real-time ITS application, e.g., avoiding traffic congestion. Diverse approaches have been investigated in the literature, employing these schemes for efficient data dissemination. Sophisticated routing, (e.g., location-aided routing [2], trajectory based routing [3], gossiping-based routing [4] and flooding), relaying protocols [5], opportunistic and delay-tolerant approaches, (e.g., spray-and-wait [6], epidemic [7] and message ferrying methods [8]) have been studied extensively within the context of ITS and its applications.

In this paper, we propose Crowdsourcing in ITS; namely, CrowdITS. Crowdsourcing enables the utilization of the Crowd to find solutions which otherwise would be hard or impossible to resolve, for some monetary reward or personal satisfaction [9]. The basic idea is to use the drivers community with smart mobile phone (Crowd) to assist in enabling non real-time and near real-time ITS applications. Motivation is to enable these applications without the need for deployment of sophisticated sensors on the roads or complex communication devices within vehicles.

Conceptually, the major change is to collect human inputs and aggregate it with other readily available sources, e.g., traffic reports from Ministry of Transportation Ontario (MTO). The aggregation is handled by the CrowdITS processing server(s) which pull-in the information from numerous sources, e.g., databases, micro-blogging, etc. The CrowdITS server pushes out the events of possible interest to the driver of each mobile phone. The push-out events are selected according to the geo-location of each member of the Crowd. Google's Cloud to Device Messaging (C2DM) service is used by CrowdITS for events delivery. We also have designed a key

[1]This research is funded by a grant from Natural Sciences and Engineering Research Council of Canada (NSERC)

ITS application; path re-routing to avoid traffic congestions. We report on the initial development of the application for the iPhone and Android smart mobile phones.

The remainder of the paper is organized as follows. Section II gives an overview of Crowdsourcing. Section III presents the CrowdITS and explains the architectural components and interfaces. Section IV describes the development of an important ITS application, vehicles re-routing, using CrowdITS framework using iPhone and Android platform. Section V presents future work and conclusions.

II. Crowdsourcing: An Overview

Crowdsourcing, in general terms, is the act of taking a job traditionally performed by a designated employee and outsourcing it to an undefined, generally large group of people (a "Crowd") in the form of an open call [9]. Technically speaking, Crowdsourcing is a distributed problem-solving and production model. In such model, initially, the problems are formulated in a format that can be understood easily by technical and non-technical people. These formulated problems are then broadcast to an unknown group of solvers, i.e., the Crowd, in the form of an open call for their solutions. The Crowd, usually knitted together via web-based technology submits their solutions. The submitted solutions can also be sorted and filtered out by the Crowd itself, in search for finding the best set of solutions. The final solution(s) is then owned by the organization which initiated the problem, i.e., Crowdsourcer. In most cases, the effort put in by the individuals, with one of the best solutions, is awarded monetarily or by some form of recognition. In other cases, the individuals are working out for their personal interests so personal satisfaction might also be sufficient. These steps are depicted in Fig. 1.

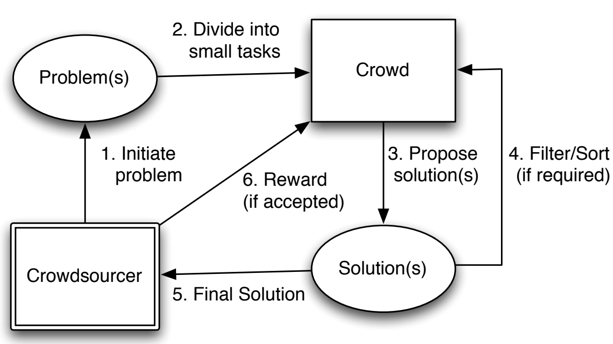


Fig. 1: Overview of main steps in Crowdsourcing

Amazon provides a Crowdsourcing platform, namely Amazon Mechanical Turk (AMT)[1], on which Crowdsourcers broadcast their problem tasks and define a price/reward (minimum is $0.01) for each satisfactorily completed task. The Crowd accepts, works on these tasks and might get rewarded. CrowdDB; a relational query processing system, has been proposed [10]. The CrowdDB makes use of AMT's API to utilize Crowdsourcing to answer queries, in the form of micro-tasks to the AMT Crowd (workers), which would not be otherwise answered by a conventional database. Crowdsourcing has found various other themes such as Crowd funding [11] and Crowd wisdom [12].

[1]https://www.mturk.com/mturk/welcome

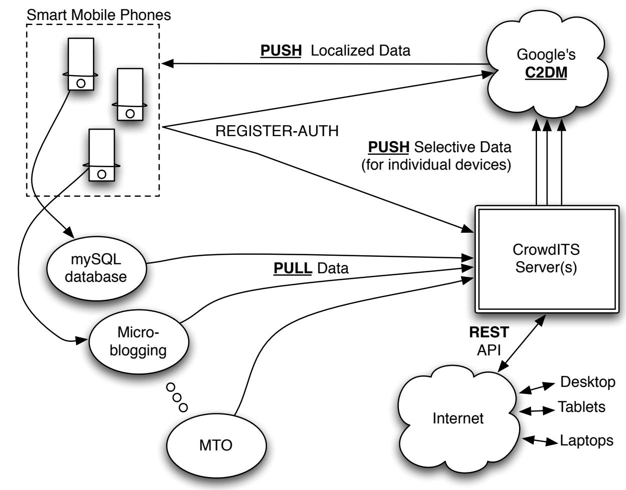


Fig. 2: System architecture of the CrowdITS

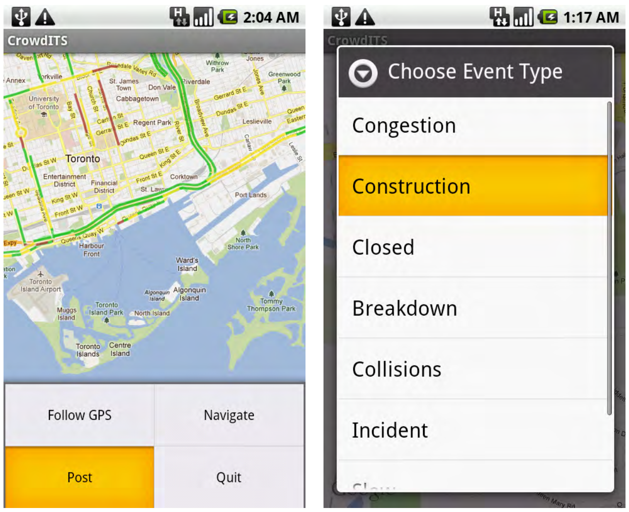
III. CrowdITS System Architecture

CrowdITS is a Crowdsourcing platform for near real-time and non real-time ITS applications, involving data collection from large number of users. Traditionally in ITS, the tasks of identifying various traffic related statistics, events and incidents are handled by specialized sensory equipments. For instance, the loop wires and CCTV on the major highways are used to meter traffic and congestion, respectively. In CrowdITS, these tasks along with others are performed by the Crowd, i.e., the driver or passenger(s) of the vehicle, using either interactive mode (e.g., voice), ubiquitous mode (e.g., GPS logging) or both. Crowd, as an appreciation, receives localized and groomed information of their potential interests. Unlike Crowdsourcing, wherein the tasks are solely handled by the Crowd, the CrowdITS is a hybrid system as it integrate Crowd-based sensing with conventional data gathering, e.g., both on-road and in-vehicle sensors.

The CrowdITS architecture is composed of three major components. These components are Crowd sensing and interface, information processing, and localized device messaging. The system architecture of CrowdITS is shown in Fig. 2. In the rest of this section, we explain the components of CrowdITS architecture.

A. Interface

Interface is of great importance in smart-phone enabled ITS applications. The interface can be interactive, pervasive or both. In interactive mode, the Crowd can enter events in two ways. First, an interactive screen, shown in Fig. 3-b, wherein a list of events are available to choose from. As a single-touch interface it avoids significant interference for the drivers.



(a) Main interface (b) Event posting interface

Fig. 3: CrowdITS interface for posting an event, navigating and GPS follow-up on an Android phone

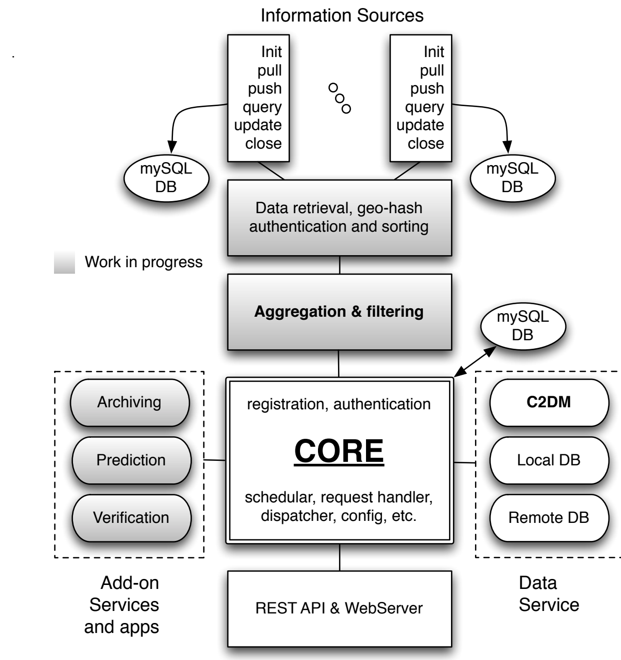


Fig. 4: Framework for the CrowdITS processing servers

Second, using voice commands wherein pre-configured sets of commands are available. Using the voice commands the Crowd can specify the event, its severity, its expected timeout and other metadata (e.g., personal message). An event, as a consequence of using either input method, is auto-stamped with current time and geo-location. The logged time is used to filter out and to set an expiry time for irrelevant entries. The geo-location, converted into geohash, is use to group the events and sent notifications to the Crowd's mobile devices. The driver can freely roam around and be notified of any potential problems ahead or can navigate to a preset destination. The main interface is shown in Fig. 3-a. In pervasive mode, the mobile phone periodically enters its sensory data (e.g. GPS) which can then be used to determine vehicle's speed and direction hence,traffic congestion. Currently, the traffic layers in Google maps are obtained using the pervasive approach. This approach however, has limited reporting facilities, i.e., reporting of only a certain type of events can be made possible. Currently, we are further evaluating the technical and non-technical aspects of different reporting approaches.

Geohashing is used to map and link multiple events together, from different sources. Geohash is a latitude/longitude geocode system based on a hierarchical spatial data structure which subdivides space into buckets of grid shape[2]. The obtained hash, a string, is used as a unique identifier to represent a point anywhere in the world. Geohash has arbitrary precision, due to its encoding mechanisms which allow having variable-size strings, and hence, flexible precision. Due to this property, any nearby places will often (but not necessarily always) present similar prefixes to the geohash string. In other words, the longer a shared prefix is, the closer the two places are.

[2]https://www.geohash.org

B. Information Processing

The CrowdITS processing server provides three main services. These include information retrieval and storage, registration and authentication, and localized messaging to mobile devices. Furthermore, additional services can also be developed, e.g., archiving and traffic prediction, information verification, web service, etc.

Numerous information sources are available in ITS. The number of sources and data volumes demand scalable and expandable architecture. We propose a plugin-based framework for the centralized CrowdITS processing server that will provide a scalable and expandable utilization of the numerous information sources and data volumes available for ITS. The overall framework of such an architecture is shown in Fig. 4. Each stream of information is handled by its own plug-in which processes incoming data packets, archives them into individual repositories and provides APIs for the core unit to pull or push, query, and update data. To balance the heavy workload of Crowd, different plug-ins can be configured to handle sub-sets of geo-location, i.e., only a range of geohash. The information archived into various databases is retrieved, filtered, and is aggregated into meaningful information by the core component of CrowdITS server. Information related to events is geo-hashed which will then be pushed to the subscribed devices using Google's Cloud to device messaging service. Using REST API, the updated information, integrated with Google maps and traffic layers, is also viewable using standard web browser. Our system is implemented using PHP. For repositories and maps, we make use of MySQL in PHP and PHP-Google-map-api version 3.0 in Google maps. Currently, we are investigating algorithms for data aggregation and traffic

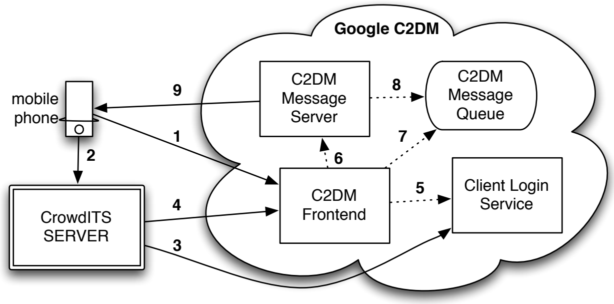


Fig. 5: Integration of the Google's C2DM with CrowdITS

predictions.

C. Localized Device Messaging

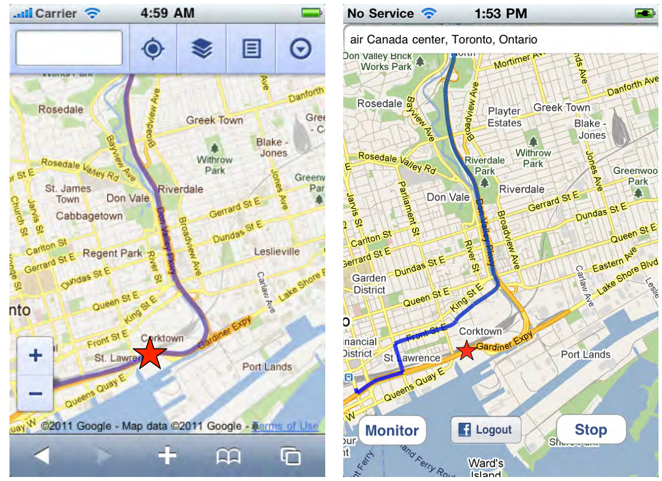
Crowd generates large volume of information, which is then aggregated with other sources to generate meaningful events for any geohash grid. The Crowd subscribes to their respective interested grid(s) for potential notifications. These notifications can be pulled by the mobile phone or pushed by the processing server to the mobile phones. In CrowdITS, we adopt the push mechanism since it is energy-efficient and has lesser communication overhead for mobile phones. To enable this we integrated the CrowdITS with the Cloud based messaging service provided by Google, namely Cloud to Device Messaging (C2DM), shown in Fig. 5.

The mobile phone registers with C2DM, using an email address, and receives a registration ID (Step-1). The Registration ID (R-ID) is unique for each device and application pair therefore allowing multiple applications to utilize C2DM services. The R-ID is sent to CrowdITS server (Step-2) where it is logged into central database. The CrowdITS server authenticates to C2DM server (step-3) and receives an AUTH token. In case of notification for any mobile phones, measured by any updates on the subscribed geohash by its associated R-ID, the CrowdITS server pushes the event data onto C2DM using its own AUTH token and mobile phone R-ID (step-4). The C2DM internally authenticates the tokens, routes the message to appropriate server and subsequently queues it (steps 6-8). The message is eventually pushed out to the mobile phone using a persistent TCP/IP session (step-9). Currently, the C2DM service is available only for the Android devices.

IV. Real-time Congestion Free Path: An Application

In this section, we describe the development of an important and fundamental ITS application, i.e., traffic re-routing to avoid congestions.

It is estimated that the annual cost of congestion for each driver is approximately \$1000 and \$200 for large and small cities, respectively. Traffic reporting, beside other solutions, is in-use to assist commuters to anticipate and avoid traffic problems. Conventionally, video cameras and radio/TV for messages collection and their broadcast respectively have been used. However, such messages are usually macro-level information, outdated, and do not provide drivers with an alternate congestion free path. Certain mobile phone-based navigational applications, e.g., Google maps, provide traffic layers that may not be up-to-date or complete. These shortcomings are due to their pervasive mode of operation as only the certain kinds of events are reportable. Fig. 6 shows snapshot of GPS traces, using GPS with Google maps and CrowdITS congestion-free path re-routing application. An event, e.g., collision, shown as red star, is logged by the Crowd minutes before reaching it. GPS-based navigational applications fail to recognize any such conditions (Fig 6-a) whereas CrowdITS finds an alternate congestion free path and guided the driver promptly (Fig 6-b).



(a) GPS Routing (b) CrowdITS Routing

Fig. 6: Comparison of the Google maps and CrowdITS path. Shown are GPS traces using, red star is a collision reported seconds before the vehicle reaches

We have developed the congestion free path re-routing application for both Android and iPhone platform. The main interface snapshots are shown in Fig. 3 and Fig. 6-b for Android and iPhone, respectively. To provide congestion free path re-routing, the application makes use of Google Maps, CrowdITS, CloudMade and C2DM APIs. Using the main screen, shown in Fig. 3-a, user enters (with auto-complete) the destination address. The entered address is communicated to Google maps server, as HTTP request, and in return receives JSON formatted responses containing all possible matching addresses. The user current GPS position and the final picked address, from the matches, is used as the path source and destination, respectively.

Using REST, the source and destination pair along with other necessary options is sent to the CloudMade server. The CloudMade server returns JSON response that includes navigational information in the form of latitude longitude pairs for the desired destination. These pairs are converted to their respective geohashes which, along with the R-ID (obtained from C2DM; explained in Section III-C), is used to subscribe

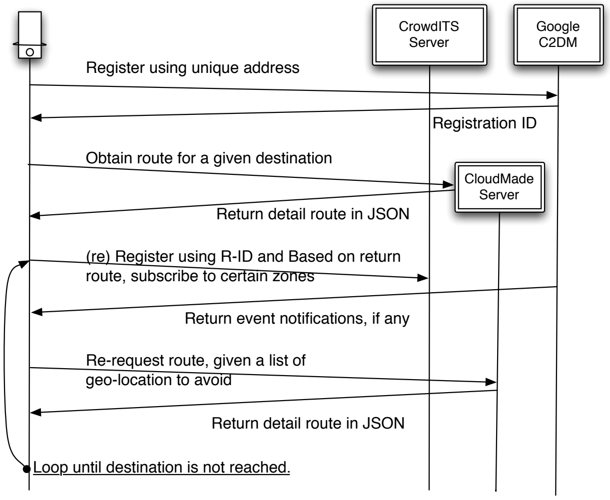


Fig. 7: Timed flowchart of vehicle re-routing using CrowdITS

for event notifications with CrowdITS. The driver navigates using the obtained path information, as the application speak out the directions at appropriate geo-locations. The parsed JSON responses are also used to plot the route using Google maps. Driver, in case if it see an event of interest, can report using either command or touch interface, as explained in Section III-A.

Upon receiving any notification in the form of latitude longitude pair(s), from the Google C2DM, the CloudMade request is re-initiated. The notifications are uploaded into C2DM by the CrowdITS, as it collect and aggregates from Crowd and other sources. The newly formed CloudMade request contains the initial destination, current GPS location and the notified latitude and longitude pairs. In case of any path change, a freshly calculated geohash zones are used to re-subscribe to the CrowdITS server and the driver are re-navigated accordingly. This process is repeated till the destination. The overall process is shown in Fig. 7. As an alternate to mobile phone notification, the CrowdITS server provide web interface to the aggregated traffic information. Fig. 8 is a snapshot of the CrowdITS webpage showing aggregated view of Crowd posted events (red), MTO logged events (orange) and traffic layers using Google maps.

V. Conclusions

Intelligent Transportation Systems (ITS) is an active area of research and development in government, academic and industrial sectors. In this paper, we advocate the use of Crowdsourcing for non real-time ITS applications. Motivation is to enable these applications without the needs of deploying sophisticated sensors on the roads or complex communication devices within the vehicles. We describe the design of Crowdsourcing in ITS; namely, CrowdITS, explains various architectural components and report on the application development of congestion free path re-routing, for the Android and iPhone mobile phones. Numerous technical and non-technical challenges still need to be overcome in order to adopt Crowdsourcing at a wider scale within ITS applications. Main research challenges are to device efficient data aggregation and verification algorithms, traffic prediction and efficient interface design for Crowd reporting. As future work, we plan to evaluate and report on the wide-scale deployment and performance of the CrowdITS system while developing other exciting ITS applications.

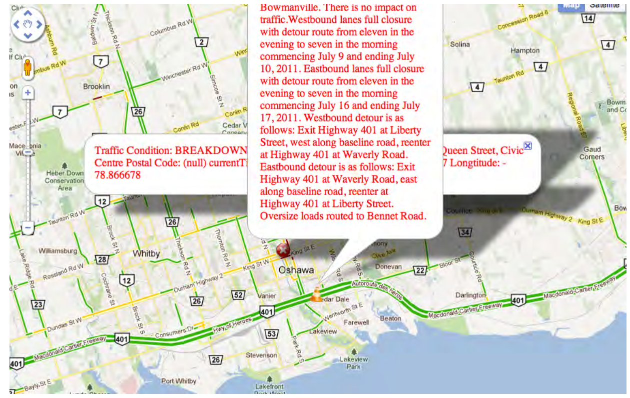


Fig. 8: Webpage snapshot, using Google map view , of integrated Crowd events, MTO logged events, and Google's traffic layers

Acknowledgement

This research is funded by a grant from Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] R. Bishop, *Intelligent Vechicle Technology and Trends*. Artech House Publishers, 2005.

[2] Y.-B. Ko and N. H. Vaidya, "Location-aided routing (lar) in mobile ad hoc networks," *Wireless Networks*, vol. 6, no. 4, pp. 307–321, 2000.

[3] D. Niculescu and B. Nath, "Trajectory based forwarding and its applications," in *Proceedings of the 9th annual international conference on Mobile computing and networking*, 2003, pp. 260–272.

[4] P. Costa, D. Gavidia, B. Koldehofe, H. Miranda, M. Musolesi, and O. Riva, "When cars start gossiping," in *Proceedings of the 6th workshop on Middleware for network eccentric and mobile applications*, 2008, pp. 1–4.

[5] J. Zhao and G. Cao, "Vadd: Vehicle-assisted data delivery in vehicular ad hoc networks," in *Proceedings of the 25th IEEE International Conference on Computer Communications*, 2006, pp. 1–12.

[6] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proceedings of the ACM SIGCOMM workshop on Delay-tolerant networking*, 2005, pp. 252–259.

[7] Z. J. Haas, J. Y. Halpern, and L. Li, "Gossip-based ad hoc routing," *IEEE/ACM Transaction on Networking*, vol. 14, no. 3, pp. 479–491, 2006.

[8] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, 2004, pp. 187–198.

[9] J. Howe, *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*. Crown Business, 2009.

[10] M. J. Franklin, D. Kossmann, T. Kraska, S. Ramesh, and R. Xin, "Crowddb: answering queries with crowdsourcing," in *Proceedings of the international conference on Management of data*, 2011, pp. 61–72.

[11] A. Ordanini, L. Miceli, M. Pizzetti, and A. Parasuraman, "Crowd-funding: Transforming customers into investors through innovative service platforms," *Journal of Service Management*, vol. 22, no. 4, pp. 443–470, 2011.

[12] J. Surowiecki, *The Wisdom of Crowds*. Achor, 2005.