# SAFARI-Taxi: **S**ecure, **A**utonomic, **FA**ult-**R**esilient, and **I**ntelligent Taxi Hailing System

Mohammad A Hoque*, Phil Pfeiffer†, Sanford Gabrielle‡, Edward Hall§, and Elizabeth Turbyfill¶
Department of Computing, East Tennessee State University, USA
{hoquem*, phil†, gabrielle‡, hallea1§, blevinsea¶}@etsu.edu

***Abstract*—The Secure, Autonomic, FAult-Resiliant and Intelligent Taxi Hailing System (SAFARI-Taxi), currently undergoing prototyping, will broker rides between taxi drivers and spontaneous taxi users, or *hailers*. SAFARI-Taxi will leverage anticipated growth in connected vehicle infrastructure (V2X) as enabled by dedicated short range communications (DSRC) technology to replace line-of-sight street hailing with automated dispatch, via public kiosks and smart phone apps. Hailing will be managed with a novel protocol, based on *Hailing Request, Service Offer, Hailer Response,* and *Ride Cancellation* messages. Threats to its operation will be mitigated using distributed dispatch; provisions for assuring correctness, timeliness, and appropriate content; and account lockouts, "hailing deposits", and ticketing. Preliminary results indicate that the system will reduce the time to match hailers with taxis. The project's goals align with the U.S. Dept. of Transportation's vision for dynamic mobility applications, including Integrated Dynamic Transit Operation, which specifically targets integration of taxis with public transportation through a citywide connected infrastructure.**

***Index Terms*—DSRC, Taxi Hailing System, Connected Vehicle, Architecture, RSU, OBU, V2X.**

## I. Introduction

Taxis are an essential component of urban transportation in metropolitan areas such as New York City, where 13,000 yellow cabs average 470,000 trips per day, carrying nearly 241 million passengers per year [1], and Singapore city, where 27,534 taxis [2] in December 2016 carried an estimated 954,000 riders per day [3]. Taxis are summoned by various means, including street hailing: the flagging down of a passing taxi with no prior reservation and no other communication with dispatchers or drivers. This has well-known drawbacks for hailers and drivers: the average time to complete a street hail can range from less than 5 minutes in downtown areas to more than 30 minutes in suburbs [4]. During rush hours, hailers can be left waiting while nearby taxis cruise, out of sight, searching for passengers [4].

Alternatives to street hailing include taxi stands and contacting a dispatcher to request a taxi at a specified pickup point. Other changes in taxi service include the gradual replacement of human dispatchers with automated dispatching systems. Most of these systems use Global Information System (GIS) positioning and the 3G/GPRS cellular network to dispatch the nearest available taxi to a prospective customer [5].

SAFARI-Taxi is an implementation of a system proposed by Hoque et al. [4], [5]. Like existing systems, SAFARI-Taxi will support smartphone-based hailing, the monitoring of passenger pickups, and the re-hailing of taxis when drivers are delayed or cancel pickups. Unlike other implementions described in the literature, SAFARI-Taxi will use dedicated short-range communications (DSRC) for hailer-to-driver communication. DSRC, an encrypted, IEEE 802.11p/Wave Short Message Protocol (WSMP) protocol, is the primary communications protocol for the U.S. Dept. of Transportation (USDOT) intelligent vehicle initiative [6]. This initiative will support DSRC-based vehicular communication between DSRC-enabled in-vehicle transceivers and roadside units (RSUs). RSUs are USDOT-standard communications relays that are being installed in major U.S. transportation corridors [7]. Having bypassed operators and middleware, RSU-enabled DSRC could reduce hailing latencies from minutes to as little as 5 seconds, which is evident from our empirical analyses using taxi trajectories.

SAFARI-Taxi will support hailing through a novel protocol that employs Hailing Request, Service Offer, Hailer Response, and Ride Cancellation messages. This is to include hailing via kiosks: RSU-enabled public interfaces. This will enable hailing by the 23 % of Americans who, according to a November 2016 Pew survey, don't own smartphones, as well as users who lack a smartphone app for a taxi service of interest [8].

Threats to SAFARI-taxi's operation will be addressed by using distributed dispatch to mitigate backbone failures; an anticorruption layer to detect and address communication failures; and measures for discouraging inappropriate use, including nominal, refundable "hailing deposits"; the disabling of accounts following bad-faith hails; and ticketing to enforce first-come, first-served service for kiosks in high demand.

It is anticipated that the prototype, once implemented, can be deployed and integrated with the New York City Department of Transportation connected vehicle (CV) pilot project funded by USDOT. The prototype, once developed, will be released through the USDOT's Open Source Application Development Platform (OSADP) [9].

The rest of the paper is organized as follows. Section II reviews previous work related to taxi hailing. Section III and IV describe SAFARI-Taxi's physical architecture and communication module, respectively. System operations are described in Section V, followed by threat management and fault-resilience in Section VI and studies of potential system performance in Section VII.

## II. Related work

Alternatives to street hailing date to the 1980's, when taxis were first equipped with radio paging systems [10]. Electronic systems for reserving taxis in Singapore in use by the early 2000's—including phone service, preset buttons in public phones, SMS messaging, online booking via computer, and taxi order terminals—were described by Liao [11], [12], [13].

The idea of fully automated taxi dispatch is at least as old as 2001 [14]. Automated dispatch systems are currently implemented by many large taxi companies. Most use GIS positioning and the 3G/GPRS cellular network to dispatch requests to the nearest available taxi.

Various history-based strategies have been proposed for positioning taxis in areas of need. Hoque et. al used GPS traces to study taxi mobility patterns in San Francisco, based on factors that include instantaneous velocity profiles, spatio-temporal distribution, the connectivity of vehicle-to-vehicle (V2V) communications, clustering, hotspots, trip duration and empty cruise interval [15]. This work was later extended to study metro-wide multi-hop V2V connectivity, partitioning, degree of reachability, and their temporal variation over a period of 24 hours [16]. Phithakkitnukoon et al. used a Bayesian classifier to predict taxi distribution in Lisbon, Portugal, based on time of day, weekday, and weather [17]. Lee et al. used a clustering analysis of spatio-temporal data from taxi traces for Jeju, South Korea to direct taxis to likely areas of need [18], [19]. Ma et al. sought to minimize taxi travel by combining spatio-temporal trace data with a lazy algorithm for shortest path computation [20]. Others, including Zhang et al. [21], have examined driver strategies for maximizing clientele.

The use of DSRC for taxi hailing was first proposed by Hoque et al. in 2012 [4], [5]. Since then, studies like those cited below have focused on specific aspects of taxi dispatch.

Chim et al. proposed three measures for assuring the safety of drivers and passengers [22]. They include requiring drivers to authenticate before providing service, authenticating location messages using OBU and RSU signatures, and using pseudonyms to allow for the tracing of taxi locations while protecting the identities of drivers and passengers.

Dow et al. propose zone queueing and path planning for managing dispatch [23], [24]. They divide coverage areas into contiguous regions based on usage, each with its own dispatcher. Their path algorithm dispatches drivers to locations in their current zones, alerting drivers when they enter inter-zone boundary regions. Dow et al. argue that their algorithm increases driver revenues while minimizing ping-ponging: i.e., repeated handoffs of taxis between dispatch servers.

Miao et al.'s *Receding Horizon Control* algorithm for taxi dispatch uses historical and real-time data from a taxi network to minimize idle mileage and passenger wait times [25]. The authors use simulations based on trace data to characterize their algorithm's effectiveness.

Wang et al. optimize data packet transmission in DSRC networks, using Markov Chain models to predict individual and group-wide driving patterns [26]. Based on a 365-day-long trace of 4303 taxis, the authors argue that their algorithm predicts taxi movements with twice the accuracy —40% vs. 20% —of comparable algorithms.

In [27], Li et al. describe a heterogeneous backbone network that uses LTE Device-to-Device (D2D) technology for communications between vehicles involved in public transit and DSRC for communications with private vehicles. Their intent was to use DSRC to reduce costs while using LTE to improve coverage and support high-speed terminals. Their work includes a novel vehicle routing protocol, TRPN, which uses fuzzy logic to route messages based on vehicle type, hop count, network traffic density, and location.

To the best of our knowledge, none of these studies addresses our key concern: the implementation of a comprehensive, DSRC-based system that supports users with and without ready access to mobile devices, while providing assurances for the system's continued operation in the event of partial failure.

## III. System Architecture

SAFARI-Taxi's architecture reflects the U.S. National Architecture for Intelligent Transportation Systems (ITS) [28] and conforms to the USDOT's *Connected Vehicle Reference Implementation Architecture* (CVRIA) [29]. SAFARI-Taxi (Fig. 1) will consist of a database-enabled dispatch server, supporting applications on DSRC Road Side Units (RSUs), Roadside Hailing Devices (RHDs, a.k.a. kiosks), one DSRC-based On Board Unit (OBU) and tablet device per taxi, and client apps for mobile devices. This portion of the design corresponds to the ITS architecture's "Transportation Layer".

The "Transportation Layer" defines a physical and a logical architecture. The physical descriptions and Fig. 1 show the physical architecture. The service descriptions and Fig. 2 illustrate the system's data flows and describe its logical architecture. In this implementation of the U.S. National ITS architecture's "Communication Layer", devices will communicate via the Internet backbone (server↔RSU), Bluetooth or Ethernet (RSU↔Kiosk), DSRC (RSU↔OBU), or Bluetooth or USB cable (OBU↔in-taxi driver device).

*RSUs.* Initially, RSUs will function as gateway devices, exchanging messages between the dispatch server, OBUs, kiosks, and user apps. Driver messages will be folded into IEEE 802.11p-standard SAE J2735 beacon messages. These are time-stamped broadcasts that specify a vehicle's latitude and longitude, speed, and direction of travel at 0.2 second intervals. Extensions to beacon messages will include a vehicle's duty status and availability to accept passengers.

*OBUs.* OBUs will link between tablets running an in-taxi app with other system components, including other OBUs and RSUs. OBUs consist of a radio transceiver, a global positioning system (GPS) and a processor that runs custom code, including implementations of WSMP, DSRC, and SAFARI-Taxi's messaging protocol. OBUs will transmit hailing-related messages via payloads in beacon messages, relaying responses and other received information to the driver's in-taxi app.

*Kiosks.* Kiosks are RSU-integrated tablet computers that accept hail requests. Kiosks will allow prospective passengers to specify a destination via a touch screen, as well as other trip requirements such as the number of passengers, luggage requirements, disability accommodations, and pickup time.

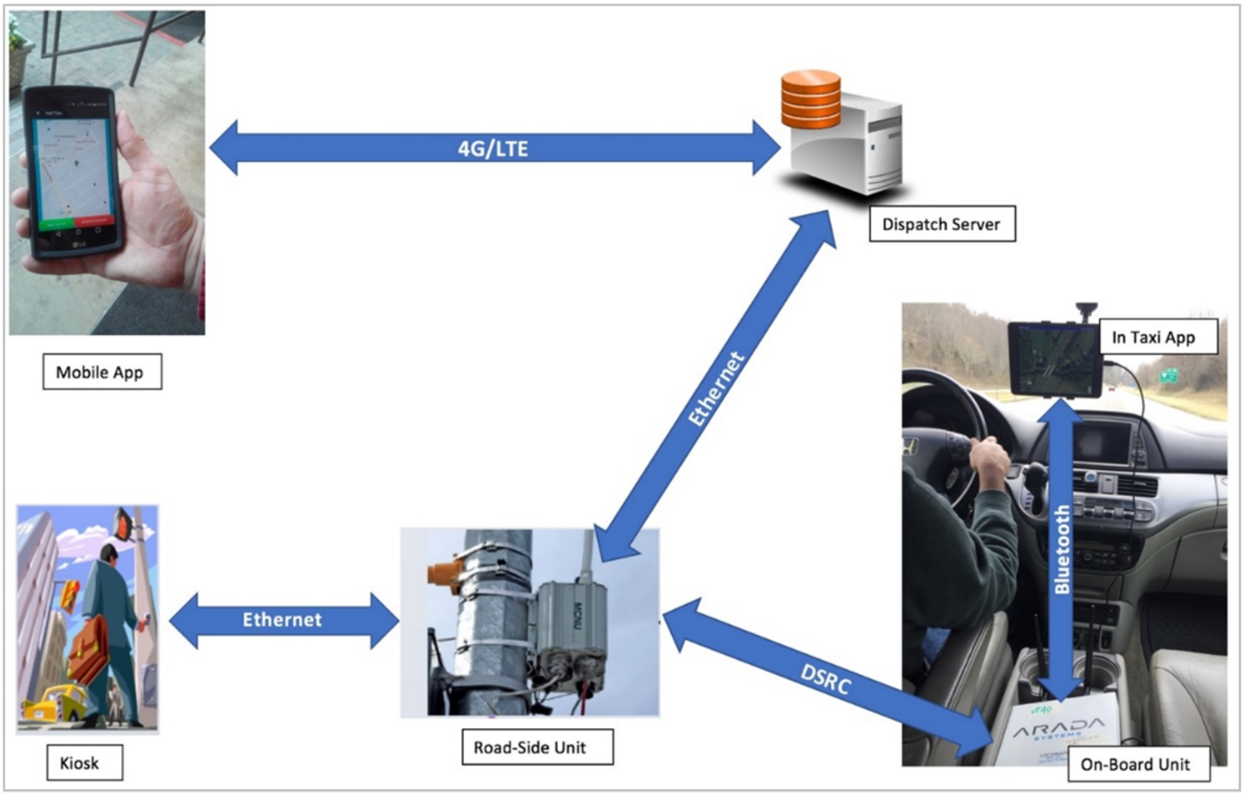


Fig. 1: SAFARI-Taxi Elements and How They Communicate

Users without accounts will swipe a magnetic strip card such as a credit/debit card or a city transit card to confirm requests.

*Dispatch server.* In the system's initial implementation, the dispatch server will match hailers with taxis. It will also track transactions following initial negotiations, log data on transactions and system operation, and collect data on hails and taxi locations from RSUs [23], [24].

*Mobile apps.* An app for SAFARI-Taxi will be developed for Android and iOS. In addition to providing kiosk-like features, the app will allow users to create accounts, edit personal information, change their passwords, recover lost passwords, specify payment options, unregister from accounts, configure default options, and specify them as part of a hail. It will use a phone's location services to identify a user's location at the time of a request. Once a ride is complete, the app will afford users the chance to rate their experience, via e-mail.

## IV. Communication Model

### A. Hailing Process

SAFARI-Taxi will support the following six-step, HTTPS-based procedure for managing hailing transactions:

1) An app or kiosk sends a hail request to the server.
2) The dispatch server selects a taxi to dispatch based on proximity to the client and client/taxi preferences.
3) If taxis are unavailable, the server ends the negotiation. Otherwise, it forwards the request to the chosen taxi, with a server-generated estimated time of arrival (ETA).
4) If the driver rejects the request or lets it time out, the server forwards the request to another taxi. Otherwise, the in-taxi app sends an offer of service to the hailer, possibly with a revised ETA.
5) The hailer either rejects the offer, explictly or by allowing it to time out, or accepts it.
6) On acceptance, the server monitors the taxi until the pickup is complete or cancelled.

### B. Hailing Request Protocol

Messaging between DSRC devices will entail four basic message types, all of which conform to the IEEE 802.11p (WAVE) transmission standard.

*Hailing Request.* Hailing Request messages are broadcast from RSUs. They contain optional fields that specify trip requirements, preceded by six mandatory fields:

- *Req_ID:* a unique identifier for each request; used to detect duplicate messages and avoid broadcast storms.
- *Comm_Mode:* a single bit that specifies whether the communication mode is vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) (1) or V2I only (0)
- *Kiosk_ID:* for kiosk-based hails, a unique identifier for the kiosk that originated the request.
- *Pickup_Location:* this location's latitude and longitude.
- *ETA range:* a time frame within which the taxi should arrive, as determined by the dispatch server.
- *Timestamp:* them when the request was first made.

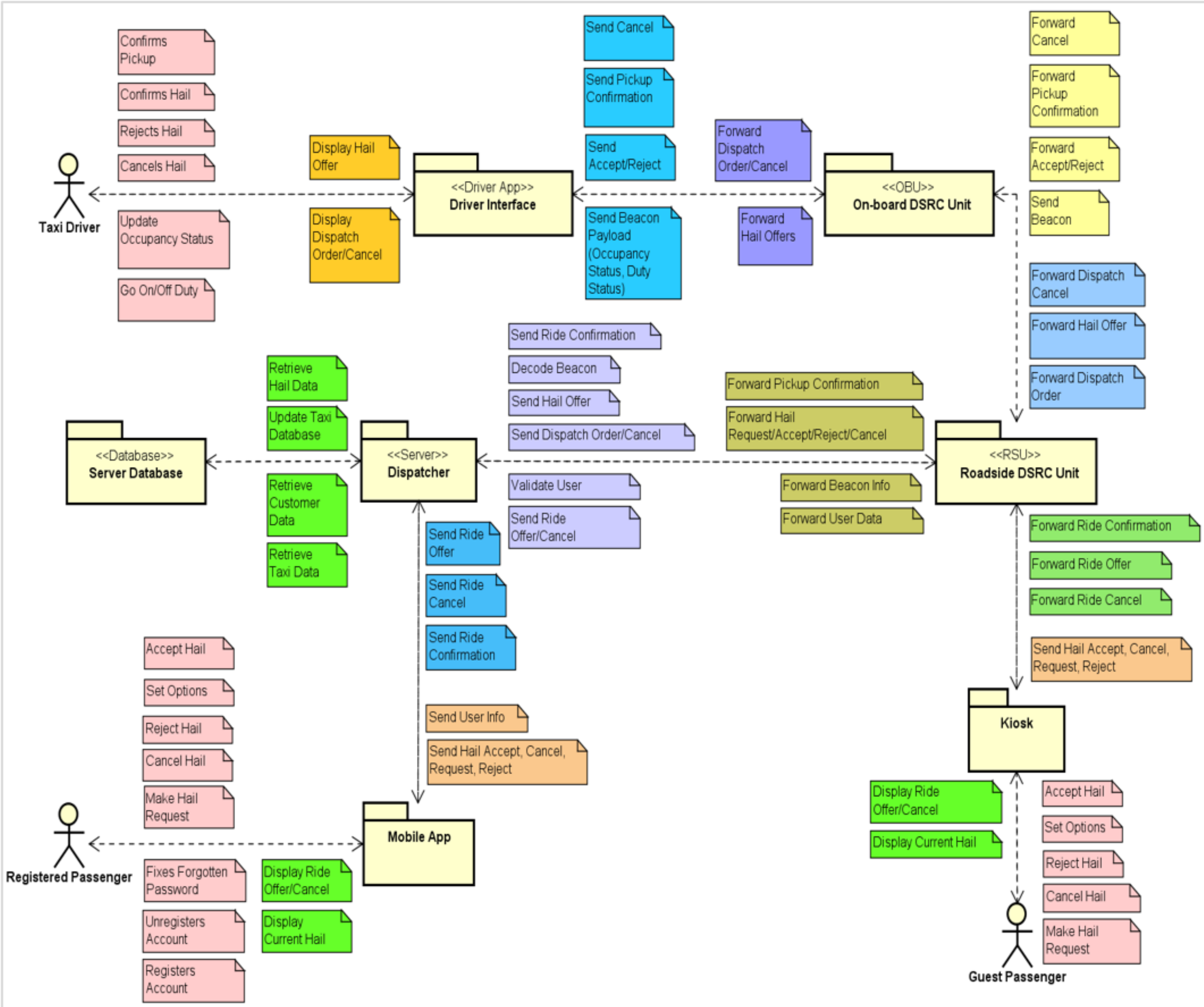


Fig. 2: SAFARI-Taxi Infrastructure, with possible messages between system elements

*Service Offer.* Service Offer messages are sent from OBUs. These messages convey a taxi driver's acceptance or rejection of a taxi request. They consist of six fields:

- *Req_ID:* the Req_ID of the corresponding hail request
- *Comm_Mode:* as above
- *AcceptReject:* a single bit; 1 iff the request was accepted
- *OBU_ID:* the responding OBU's unique identifier.
- *Taxi number:* included when drivers accept requests
- *ETA range:* the initial request's ETA, as optionally adjusted by the driver and OBU.

*Hailer Response.* This message is a subset of the Driver Request message. Kiosks and apps generate it when a hailer responds to a Service Offer message or implicitly rejects a hail by allowing the offer to time out.

*Ride Cancellation* A driver or passenger may cancel a hail before the passenger is picked up, due (e.g.) to unavoidable delays in a taxi's arrival. This message is identical to the Hailer Response message, except for its use of a point of origin bit (0 for hailer, 1 for driver) in lieu of the accept/reject bit.

*Other Message Types* Resync messages will be used to reconfirm transactions following temporary losses of driver-passenger communication. Additionally, the use of an OBU-based mobile *ad hoc* network-(MANET) to backup Internet service would require an OBU_ID field in Hailer Response messages and a Service Acknowledgment message from a driver, due to the need to allow hailers to select from the multiple offers that distributed dispatch could yield [30], [31].

## V. System Operation

Fig. 3 presents an activity diagram for the initial, centralized implementation of the SAFARI-Taxi server. Due to space limitations, kiosk, user app, and driver app/OBU state transition diagrams are omitted. A brief characterization of each of these diagrams follows.

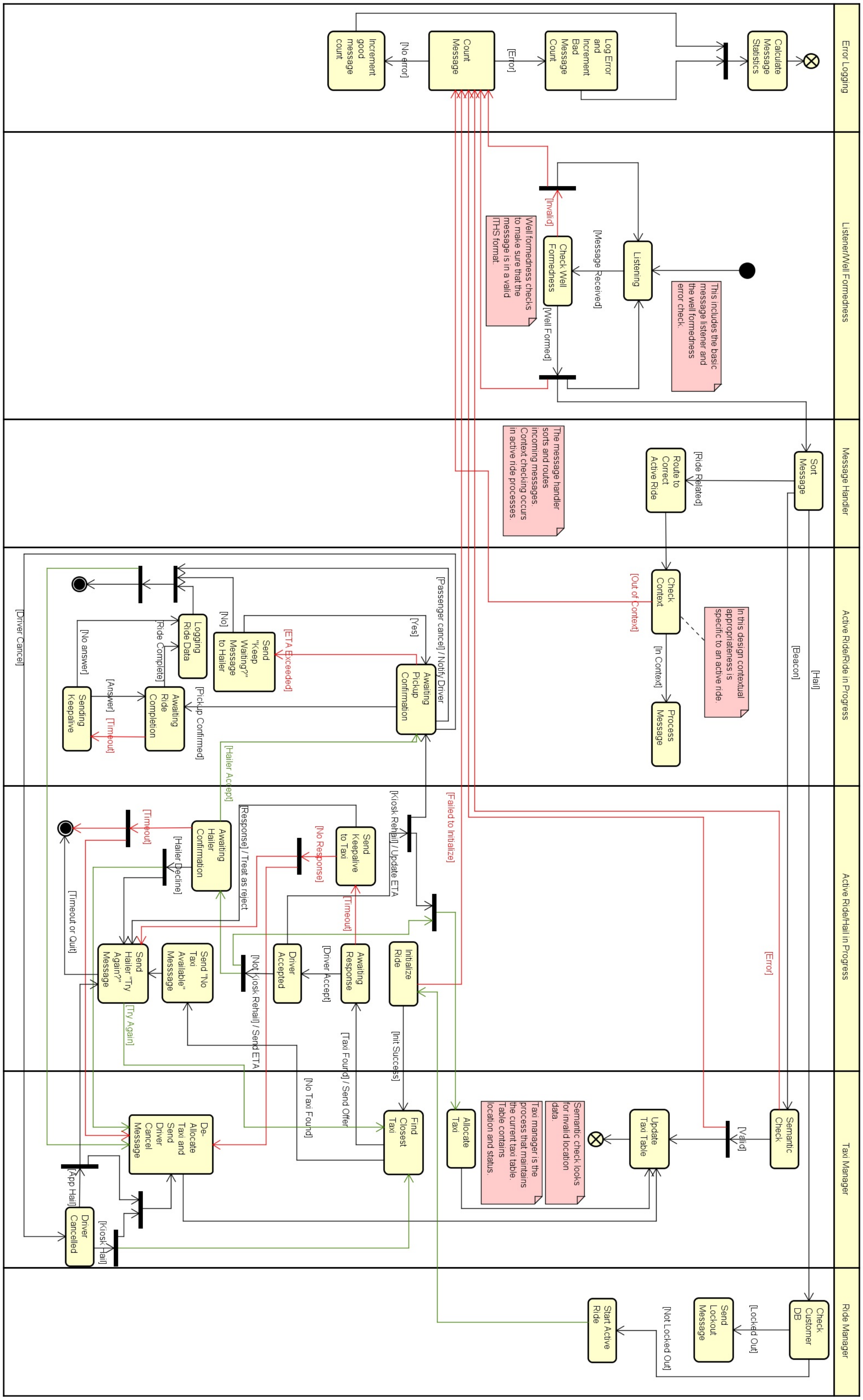


Fig. 3: Activity diagram for the initial version of the dispatch server, with centralized dispatch logic

| | |
|---|---|
| Hailing Protocol | Hailing Exception Management |
| Semantic Correctness | |
| Contextual Correctness | |
| Syntactic Correctness | Tabulation of Transmission Statistics |

Fig. 4: Four-layer SAFARI-Taxi message processing architecture

The *state transition diagram for kiosks* defines a partial map from 288 (state, event) pairs to one of 12 successor states.

- Kiosk states include seven for normal hailing (login requested, no hail yet requested, hail requested, hailer to decide on offer, hailer awaiting driver before/after ETA, driver at pickup), four for endpoint connection failures (hail not in progress, hailer awaiting driver before/after ETA, driver at pickup) and one for hailer lockout.
- Kiosk transition events include two for lockouts (lockout message received, time expired), two for card swipes (valid/invalid card), eight for hail negotiation (user requests hail, user quits session on timeout, no taxis available, hailer cancels request, hailer refuses rehail, hailer receives offer, hailer rejects/accepts offer), six for hails in progress (driver cancels pickup, hailer cancels pickup, taxi misses ETA, taxi arrives at rendezvous, driver signals pickup fails/succeeds), two for passenger transport (driver signals taxi at destination/drop-off completed), and four for lapses in communication (taxi went offline, taxi came back online, kiosk went offline, kiosk came back online).

The *mobile app state transition diagram* defines a partial map from 432 (state, event) pairs to one of 18 successor states.

- Mobile app states include eight for normal hailing (no hail yet requested, hail requested, hailer to decide on offer of rehail, hailer to decide on offer of ride, hailer awaiting driver before/fter ETA, driver at pickup, pickup successful), four for endpoint connection failures (as above), four for app out for range (hail not in progress, hailer awaiting driver before ETA, hailer awaiting driver after ETA, driver at pickup), one for hailer lockout, one for awaiting reply to resync request after lapse in communication, and one (successor-less) terminal state.
- Mobile app transition events include two for lockouts (as above), seven for hail negotiation (as above, less user quits on timeout) two for hails in progress (taxi misses ETA, taxi arrives at rendezvous), two for passenger transport (as above), four for lapses in communication (as above), and five for resyncs (awaiting driver before/after ETA, driver at pickup, pickup successful, hail cancelled).

A third, *driver state transition diagram* defines a partial map from 506 (state, event) pairs to one of 22 successor states.

- Driver app states include eight for taxi online (off duty, accepting hails, deciding on offer, awaiting hailer response to offer, en route to rendezvous, attempting pickup, en route to/arrival at destination), six for taxi offline (off duty, hail not in progress, en route to rendezvous, attempting pickup, en route to/arrival at destination), six for taxis in known dead zones (off duty, hail not in progress, en route to rendezvous, attempting pickup, en route to/arrival at destination), and two for dropping hails in progress (driver going off/ remaining on duty).
- Driver transition events include two for duty status (driver going off/coming on duty), six for hail negotiation (driver receives incoming hail request, driver rejects/accepts request, hailer cancels request, hailer reject/accepts offer), five for hails in progress (driver cancels confirmed hail, passenger cancels confirmed hail, driver signals taxi at rendezvous, driver signals pickup fails/succeeds), two for passenger transport (driver signals taxi at destination/ride completion), six for lapses in communication (taxi went offline, passenger went offline, taxi entered dead zone, passenger entered dead zone, taxi came back online, taxi emerged from dead zone), and two for dropping hails in progress (driver confirms/rejects handoff).

## VI. Threat management & Fault Resilience

SAFARI-Taxi's design is intended to mitigate threats due to failures of backbone components; communication failures, including dropped, missed, garbled, and inappropriate messages; and inappropriate use by hailers.

### *A. Backbone failures*

Server failures can be addressed with hot backups. Internet failures can be mitigated in one of three ways: by dividing the service area into zones, placing a server in each zone, and using a distributed algorithm to support cross-zone hailing; with a two-tier routing algorithm, wherein RSUs match users with taxis in their local coverage areas; or through the use of a mobile ad hoc network (MANET) for DSRC-based OBU-to-OBU communication, as described (e.g.) by Zhou et al. [30] and Mazilu [31]. Two potential obstacles to MANET use would be the limited memories of contemporary OBUs and the need to manage transitions between Internet- and DSRC-based dispatch on Internet failure and resumption of service.

### *B. Communication failures*

Potential communication failures include garbled messages, communication lapses, and messages with unexpected and implausible content. These issues will be addressed by an anti-corruption layer (ACL) [32] in OBU-, kiosk-, and smartphone-resident communication software (bottom three levels, Fig. 4).

The ACL's lowest, *syntactic correctness* layer will verify that messages conform to a recognized format. It will include a component that compiles statistics on error rates. These statistics will be used to check for possible degradations in quality of service, due (e.g.) to interference with wireless transmissions or high levels of network usage.

The ACL's middle, *contextual correctness* layer will use timeouts to detect and respond to losses of communication. It will retry failed transmissions and resynchronize with parties involved in hails following temporary lapses in communication. It will also verify messages' appropriateness in regards to the expected flow of a negotiation. This shall include attempting to identify out-of-order transmissions.

The ACL's uppermost, *semantic correctness* layer will verify the plausibility of a message's content, checking (e.g.) for implausible changes in taxi velocity and location, as per the 2005 Draft IEEE WAVE security standard [33].

Logic for normal processing will reside above the ACL, with logic for managing ACL exceptions. While options for managing exceptions are limited—e.g., ignore them, reset the current operating mode, warn of possible service degradation, halt for maintenance—explicitly matching anomalies to options should help to assure system quality.

### C. Inappropriate use.

Bad-faith hailing by kiosk users without accounts will be deterred by requiring the use of a financial magnetic strip card to confirm a hail. This will allow the system to assess a token fee for hails that are cancelled or abandoned before an ETA expires. Bad-faith hailing by users with accounts will be addressed by temporarily locking these users' accounts.

Hailers could also attempt to displace a taxi's rightful claimant. To prevent such "taxi jumping", kiosks in high-use areas will issue tickets to users who place hails, stamped with a time and an identifying number. When a taxi arrives at a kiosk, the taxi will download a list of active tickets and their times of issue. This will allow a driver to service the next hailer in the queue, to confirm that hailer's identity, and to remove the request from the kiosk's "flight board".

Bad faith actions by drivers are more difficult to address. Hailers whose taxis fail to arrive on schedule will be allowed to cancel or request another taxi, with no penalty. Repeated failure by a driver to rendezvous with hailers can be detected from hailing records and managed outside the system.

## VII. Performance Evaluation

SAFARI-Taxi's potential effectiveness was assessed using minute-by-minute traces of locations and occupancy from San Francisco Yellow Cab Company's 536 taxis. Traces were taken between 2 pm and 3 pm on Monday, 2 June 2008, a random working day [10]. The site occupied by the taxi in Fig. 5 was chosen as a hypothetical pickup point for comparing line-of-sight and kiosk-based hailing. An RSU on this location's light post, GPS (37.788031,-122.406691), should conservatively detect beacon messages at a range of 300 meters (red circle, Fig. 6). A kiosk at this location could summon any taxi in this range, as opposed to street hailing, which limits hailers to taxis in the surrounding road segment (blue rectangle, Fig. 6).

Fig. 7 shows taxi availability for the RSU-enabled and line-of-sight hailing regions from 2 pm to 3 pm on 6/2/2008. Red and blue bars indicate the number of free taxis in the red circle and blue rectangle, respectively. During this time, 150 free taxis entered the RSU's zone of coverage, as opposed to 5 that passed in view of point A. A person looking for a taxi at 2 pm would have waited until 2:14 pm for a taxi, as opposed to 1-2 minutes with SAFARI-Taxi.

An analysis of the hit size for 2 June 2008 at location (37.788031,-122.406691) showed that 4.01 taxis on average could respond to an RHD call at any time between 6 AM and midnight. This means at least 240 passengers per hour could be served by this hypothetical downtown kiosk. This analysis applies to San Francisco, which at the time had 536 total yellow cabs. In a larger city like New York, where 13,000 yellow medallion cabs average of 470,000 daily trips, the average hit size would be much higher. The hit size for suburban traffic is comparatively low as people mostly travel with their own cars. Even so, an analysis of a second sample location (37.7573, -122.491363) near San Francisco Bay gives an average hit size of 0.18 taxis per minute and 194 per day.


Fig. 5: Street view of the pickup spot

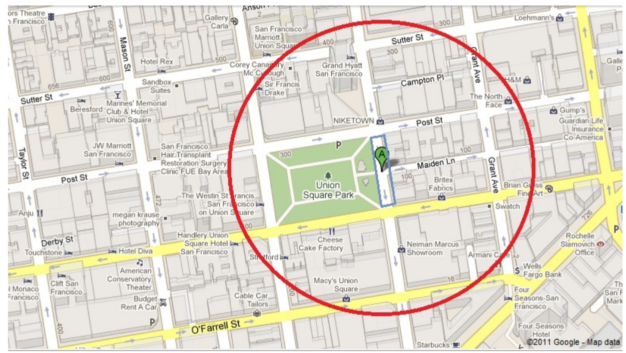


Fig. 6: Map of the pickup location (A), its DSRC broadcast area (red circle), and line-of-sight hailing area (blue rectangle)

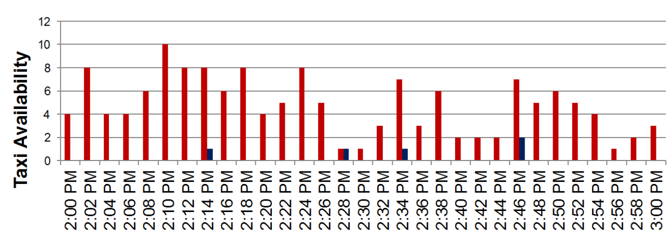


Fig. 7: Potential increase of cab availability with kiosk-based hailing. Red bars indicate taxis in range of the hypothetical RSU; blue bars, in line of sight from the pickup point

## CONCLUSION

SAFARI-Taxi is designed to be integrated with smart city infrastructures via DSRC technology, thereby providing data on transportation while improving taxi service and extending it to those without taxi apps. We have addressed issues related to threat management, dependability, and fault-tolerance. We plan to extend SAFARI-Taxi to integrate taxis with a multi-modal public transport service that allows urban commuters to book travel as a combination of taxi and public transit. For example, urban commuters could reduce travel times by combining bus rides with immediate taxi service. Since DSRC will connect buses and taxis to a common, citywide infrastructure, taxi drivers can determine when a bus will arrive and where to pickup a passenger. This would be particularly helpful when a final destination is outside of the transit bus system.

SAFARI-Taxi's prototype, currently in development, is expected to launch by 2018. We plan to incorporate SAFARI-Taxi with ride-sharing options and integrate with USDOT's dynamic mobility applications. When complete, the application's source codes and technical documentations will be released through USDOT's OSADP platform.

## ACKNOWLEDGEMENTS

The authors thank Rich Mauritz-Miller for his careful reviews of this paper and Jason Jones, Chris Maness, Nicolas McMahon, and Matt Moore for contributions to this work.

## REFERENCES

[1] NYC Taxi cab fact book; http://www.schallerconsult.com/taxi/taxifb.pdf, Mar. 2006; last accessed 25 May 2017
[2] Singapore Land Transport Authority, "Taxi info for LTA Website", https://www.lta.gov.sg/content/dam/ltaweb/corp/PublicationsResearch/files/FactsandFigures/taxi_info_2016.pdf; last accessed 24 May 2017
[3] Singapore Land Transport Authority, "Public Transport Utilisation - Average Daily Public Transport Ridership", https://data.gov.sg/dataset/public-transport-utilisation-average-public-transport-ridership; last accessed 24 May 2017
[4] Mohammad A Hoque, Xiaoyan Hong, Brandon Dixon, "Taxi Hailing System using Connected Vehicle Technology", ITS World Congress 2014
[5] Mohammad A Hoque, Xiaoyan Hong, Brandon Dixon "Innovative Taxi Hailing System Using DSRC Infrastructure", 22nd Annual ITS America Conf. and Exposition, 21-23 May 2012
[6] U.S. Department of Transportation, "DSRC: The Future of Safer Driving", https://www.its.dot.gov/factsheets/dsrc_factsheet.htm; last accessed 14 May 2017
[7] Amanda Ziadeh, "Smart transportation hits interoperability speed bump", 6 Jan. 2017. https://gcn.com/articles/2017/01/06/virginia-connected-vehicle-infrastructure.aspx; last accessed 26 May 2017
[8] Aaron Smith, "Record shares of Americans now own smartphones, have home broadband", 12 January 2017, Pew Research; http://www.pewresearch.org/fact-tank/2017/01/12/evolution-of-technology/, last accessed 30 May 2017
[9] https://itsforge.net, last accessed 30 May 2017
[10] Yellow Cab of San Francisco; http://www.yellowcabsf.com/, last accessed 30 May 2017
[11] Ziqi Liao, "Managing Wireless Communications in Taxi Services", Proceedings of the 5th Int'l Conf. on Electronic Business, Hong Kong, Dec. 5-9, 2005
[12] Ziqi Liao, "Taxi Dispatching via Global Positioning Systems", IEEE Transactions on Engineering Management, Vol. 48, No. 3, Aug. 2001
[13] Ziqi Liao, "Real-Time Taxi Dispatching Using Global Positioning Systems", Comm. ACM, Volume 46 Issue 5, May 2003
[14] Yen, John, "Intelligent taxi total service system", US Patent 6456207, 20 Feb. 2001; https://www.google.com/patents/US6456207, last accessed 25 May 2017
[15] Mohammad A Hoque, Xiaoyan Hong, and Brandon Dixon, "Analysis of mobility patterns for urban taxi cabs." 2012 Int'l Conf. on Computing, Networking and Communications, IEEE, 2012.
[16] Mohammad A Hoque, Xiaoyan Hong, and Brandon Dixon, "Efficient multi-hop connectivity analysis in urban vehicular networks." Vehicular Communications 1.2 (2014): 78-90.
[17] Santi Phithakkitnukoon, Marco Veloso, Carlos Bento, Assaf Biderman, and Carlo Ratti, "Taxi-Aware Map: Identifying and Predicting Vacant Taxis in the City", 2010; in LNCS 6439 (Ambient Intelligence 2010), pp. 8695, B. de Ruyter et al. (Eds.)
[18] Junghoon Lee, "Analysis on the waiting time of empty taxis for the Taxi telematics system", 2008, 3rd Int'l Conf. on Convergence and Hybrid Information Technology, pp. 66-69
[19] Junghoon Lee, Inhye Shin, Gyung-Leen Park, "Analysis of the passenger pick-up pattern for taxi location recommendation", 2008, 4th Int'l Conf. on Networked Computing and Advanced Information Management, pp. 199-204; DOI 10.1109/NCM.2008.24
[20] Shuo Ma, Yu Zheng, Ouri Wolfson, "T-Share: A Large-Scale Dynamic Taxi Ridesharing Service", 2013, Int'l Conf. on Data Engineering, pp. 410-421
[21] Daqing Zhang, Lin Sun, Bin Li, Chao Chen, Gang Pan, Shijian Li, Zhaohui Wu, "Understanding Taxi Service Strategies From Taxi GPS Traces", IEEE Transactions on Intelligent Transportation Systems, Vol. 16, No. 1, Feb. 2015; DOI 10.1109/TITS.2014.2328231
[22] T.W. Chim, S.M. Yiu, Lucas C.K. Hui, Victor O.K. Li, "VANET-based secure taxi service", Ad Hoc Networks 11 (2013), pp. 23812390, Elsevier
[23] Chyi-Ren Dow, Yu-Hong Lee, Shr-Chen Wang, and Shiow-Fen Hwang, "A Geo-aware Location Based Taxi Carrying System", 11th Int'l Conf. on Information Technology: New Generations, 7-9 Apr. 2014
[24] Chyi-Ren Dow, Duc-Binh Nguyen, Shr-ChenWang, Shiow-Fen Hwang, and Ming Fong Tsai, "A Geo-Aware Taxi Carrying Management System by Using Location Based Services and Zone Queuing Techniques on Internet of Things", Mobile Information Systems, Volume 2016, Article ID 9817374, Hindawi Publishing; http://dx.doi.org/10.1155/2016/9817374, last accessed 24 May 2016
[25] Fei Miao, Shuo Han, Shan Lin, John A. Stankovic, Desheng Zhang, Sirajum Munir, Hua Huang, Tian He, George J. Pappas, "Taxi Dispatch With Real-Time Sensing Data in Metropolitan Areas: A Receding Horizon Control Approach", IEEE Transactions on Automation Science and Engineering, Vol. 13, No. 2, Apr. 2016
[26] Yang Wang, Erkun Yang, Wei Zheng, Liusheng Huang, Hengchang Liu, Binxin Liang, "A Realistic and Optimized V2V Communication System for Taxicabs", 2016, IEEE 36th Int'l Conf. on Distributed Computing Systems, pp. 139-149; DOI 10.1109/ICDCS.2016.28
[27] Gen Li, Maode Ma, Chunfeng Liu and Yantai Shu, "Routing in Taxi and Public Transport Based Heterogeneous Vehicular Networks", 2016 IEEE Region 10 Conf., 22-25 Nov. 2016
[28] U.S. Dept. of Transportation, "National ITS Architecture", http://local.iteris.com/itsarch/index.htm; last accessed 31 May 2017
[29] https://www.standards.its.dot.gov/DevelopmentActivities/CVReference; last accessed 31 May 2017
[30] P. Zhou, T. Nadeem, P. Kang, C. Borcea and L. Iftode, "EZCab: A Cab Booking Application Using Short-Range Wireless Communication", In Proceedings of 3rd Int'l Conf. on Pervasive Computing and Communications 2005
[31] Sinziana-Petronela Mazilu, "EZCab: A VANET application for taxi booking", 2009, University Politehnica of Bucharest
[32] Eric Evans, "Domain-Driven Design: Tackling Complexity in the Heart of Software", Addison-Wesley, 2003, p. 364 ff.
[33] SCC32 Committee of the IEEE Intelligent Transportation Systems Society: Draft Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages, Oct. 2005, Draft IEEE P1609.2/D1.3; cited by Christine Laurendeau, Michel Barbeau, "Threats to Security in DSRC/WAVE", T. Kunz and S.S. Ravi (Eds.): ADHOC-NOW 2006, LNCS 4104, pp. 266279