Remote Medical Monitoring Through Vehicular Ad Hoc Network

Hyduke Noshadi[*‡],Eugenio Giordano[*†§], Hagop Hagopian[*]
Giovanni Pau[*], Mario Gerla[*], Majid Sarrafzadeh[*]
[*]Computer Science Department - University of California, Los Angeles, CA 90095, US
e-mail: {hyduke|giordano|gpau|gerla|majid}@cs.ucla.edu
e-mail:{shaitani@gmail.com}
[†]DEIS, WiLab - Università di Bologna, Bologna, 40136, Italia
[‡]Google Inc.

Abstract

Several diseases and medical conditions require constant monitoring of physiological signals and vital signs on daily bases, such as diabetics, hypertension and etc. In order to make these patients capable of living their daily life it is necessary to provide a platform and infrastructure that allows the constant collection of physiological data even when the patient is not inside of the coverage area. The data must be rapidly "transported" to care givers or to the designated medical enterprise. The problem is particularly severe in case of emergencies (e.g. natural disasters or hostile attacks) when the communications infrastructure (e.g. cellular telephony, WiFi public access, etc) has failed or is totally congested. In this paper we present an evaluation of of the vehicular ad-hoc networks (VANET) as an alternate method of collecting patient pre-recorded physiological data and at the same time reconfiguring patient medical wearable body vests to select the data specifically requested by the physicians. Another important use of vehicular collection of medical data from body vests is prompted by the need to correlate pedestrian reaction to vehicular traffic hazards such as chemical and noise pollution and traffic congestion. The vehicles collect noise, chemical and traffic samples and can directly correlate with the "stress level" of volunteers.

I. Introduction

Ubiquitous computing is getting more popular due to design and fabrication of powerful small size processing units. The embedded architecture can be seen in a variety of applications such as invasive sensing, urban sensing, automotive, air and space, and health care industry. Different platforms have been introduced which can be used to monitor medical data while patient is out of hospital. These architectures can be used for two classes of applications, some are life critical (ECG Monitoring [2]) and some are not (monitoring the pressure on the feet or the way people are walking after post knee surgery). Some of the applications that are life critical must have the capability of reporting medical data to physicians with fixed schedules periodically. In addition these classes of applications must be able to notify the paramedics in case of any emergency to get immediate response. In all the cases when an application intends to transfer data to its appropriate destination, the patient either must be under coverage of a WiFi wireless access point or its on-body terminal must have the capability of using the cellular network. However, in some cases (emergencies, attacks, etc), neither the WiFi nor the cellular network is available. This fact introduces a major reliability issue regarding the extension of these applications to mobile patients.

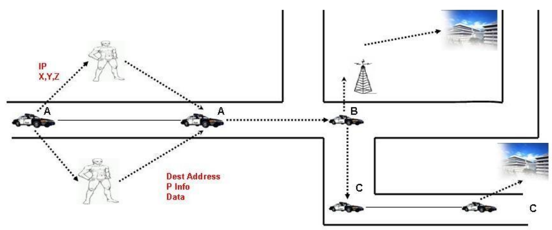

Fig. 1. Vehicles collect data from patients and transfer them to their final destination.

Vehicles collect data from patients and transfer them to their final destination using an ad hoc networks, where each mobile node is a vehicle. The vehicles communicate with each

§This work was supported by Microsoft Research through its PhD Scholarship Programme.

other and with nearby local roadside base stations. Vehicles move in an organized fashion and their range of motion is somehow restricted. For example, they mostly drive in the streets and highways. Moreover, a commuter drives every day almost just about the same time to work and returns home almost following same path.

VANETs have been used in many useful applications such as, forensic accident or crime site investigations, terrorist alerts, traffic congestion control, urban pollution monitoring, advertisement and content delivery [13][14] [11][3][12]. For example, a vehicular network can be utilized to alert drivers from possible traffic jams. Mobeyes [11] and CarTel [3], use sensors to collect real-time data, from the environment. The data can be used for possible crime investigation, traffic congestion estimation or air pollution analysis. VANET can also be used to propagate emergency messages.

In this paper we propose the use of VANETs as a medium to transfer patients data to its final destination and also as a reconfiguration and preliminary processing gateways. In section II we introduce some application scenarios in which our proposed platform can be used. Section III describes the medical monitoring platform and infrastructure, section III-C describes software components and architecture detail, section IV describes algorithms and communication protocols, section VI contains our experimental results and finally in section VII we describe the future work and conclusion.

II. Application Scenario

Vehicular ad hoc networks can be a very useful communication medium for medical monitoring applications in many of the following scenarios. 1) Patients carrying wearable medical monitoring devices are walking around in the streets outside of the wireless access point and there is a need to send the data to the physicians or medical centers, because of an emergency or according to a schedule. 2) In the case of a disaster, when the network infrastructure of an area is completely destroyed, there are many people with medical issues that need constant monitoring. 3) Physicians decide to reconfigure the software or the medical monitoring device, while the patient is outside of the wireless coverage range.

In such scenarios trusted and authorized vehicles, such as police cars, paramedics, drive down the streets sending advertisement messages. These messages indicate that a car is ready to upload the medical data from patient. The patient's on-body terminal receives the advertisement message from vehicles. If it needs to transfer data, it starts the handshaking process with the vehicle and transfers the message to that vehicle by indicating the destination address, where eventually the data will be delivered. In addition, supposed that, based on the collected data the physicians decide to monitor patients from a different perspective (For example if the received data has been filtered in the processing unit or on-body terminal, they might also want to check the unfiltered data). In order to achieve this, the wearable system might need reconfiguration at software level. In this case, vehicles can be used as mobile reconfiguration stations. Physicians can initiate appropriate reconfiguration initiation for their patient with particular ID, which is located in a particular geographical location. Vehicles reaching the patient initiate and perform reconfiguration.

III. Medical Monitoring Platform

A. Light-weight BodyNet

The medical monitoring platform should be a lightweight wearable system that people can carry. We selected the following components from the architecture described in [4]:

Sensors: Depending on the wearable platform's purpose, different types of sensors can be used to serve for variety of physiological measurements. Variety of the sensors, such as a single point pressure sensor, flex sensor, pressure sensor sheet, piezoelectric film sensors, accelerometers, galvanic skin response sensors and electrocardiogram can be used individually or in the group to collect desired physiological data.

Data Processing, Communication and Gateway Units: Processing units are mica2 [7] motes, which are low power processing units. These processing units have communication capabilities and can transfer sensor readings to the on-body terminal. We also have the same type of mote connected to on body terminal, which acts as gateway mote. It forwards advertisement messages received from the vehicles, which advertises the message from the same type of mote, and also collects the data broadcasted from processing units and sends it to on-body terminal.

On-body Terminal: On-body terminal, which in our scenario is a Pocket PC, is used to collect the sensed data from the sensors, which are broadcasted by processing units attached to the wearable system. It performs some more intensive computational tasks on the collected data such as filtering. In addition, it is a hub between the wearable devices and the outside world. On body terminal initiates the communication with vehicles, transfers the chucks of sensed data to the vehicles, guarantees the complete content delivery, and keeps track of patient's location by utilizing the embedded GPS system.

B. Vehicular Equipment

Every car is equipped with an onboard computer, smart GPS antenna and a 802.11g wireless card. There are two types of cars in the system. The first type is the leading cars that has EvDo Interface, which provides the entire Ad Hoc networks with connection to the internet. The second type is non leading car that does not have direct connection to the internet but uses the leading car as a gateway. The routing algorithm used in the cars is OLSR [10]. Each car has a low power mica2 [7] mote, which is advertising the cars IP address and its location consistently. The advertised IP addressed will be used by on-body terminal to initiate the communication if desired. The mote can also be used to send reconfiguration command to the wearable system.

C. Software architecture

In this section we describe the software architecture and capabilities of the programs running on wearable system, on-body terminal, vehicular network and care givers monitoring station.

1) Software running on lightweight components: We used TinyOS [5] as the operating system running on the low power processing units. TinyOS is a component-based operating system designed at University of California Berkeley for wireless sensor networks. The application running on the motes are written in NesC [6]. The program running on the processing elements attached to human body reads the physiological data from ADC channels, that are attached to passive sensors, perform some preprocessing, construct the packet with additional information and transmit the data to the on-body terminal over the wireless channel. The gateway stations attached to on-body terminal is designed such that it sends the data to the pocket pc over the UART(Universal Asynchronous Receiver Transmitter). In each vehicle we attached a lightweight mote (mica2). This mica2 mote is connected to the computer running on the vehicle. The main responsibility of this mote is to broadcast the IP address of the vehicle in the ad hoc network. In addition, this lightweight mote can be used to perform as a gateway to perform the initial steps needed for lightweight system reconfiguration.

2) Software Running on On-Body Terminal: The program is responsible for data collection, initiating the communication between medical wearable system and vehicle, and guarantees the complete content delivery. The program time stamps and stores the data collected from wearable system. Furthermore it performs some computationally intensive data processing on the recorded patient data and logs its outcome, which is meaningful for physicians. In some applications this might reduce the amount of transmitted patient data over the network. (For example sending heart beat interval instead of the whole heart beat [2]). The software running on the base station also initiates the connection with the vehicle and transfers the data to the vehicle. This program is implemented in C ♯ . The graphical user interface of the software is shown in 2. It lets the user to specify the final destination IP address and also shows the location of patient and cars which are in the range in the map.

3) Software running on vehicles: Two sets of programs are running in each car. On each vehicle, a server is running which is responsible to collect the data received from each patient through on-body terminal. The received packet will be saved with identifier indicating the patients ID, packet destination address and packet number. Another program will take the collected data and send it to the leading car either directly or indirectly. On each step the leading car will advertise the list containing the packets that it has received. Non leading vehicles upon receiving the list will check to see if they have some packets that are not in the list. If there exist such a packet, they will send the packet to the leading vehicle. The leading car is then in charge of forwarding the data to the physician using its internet connection.

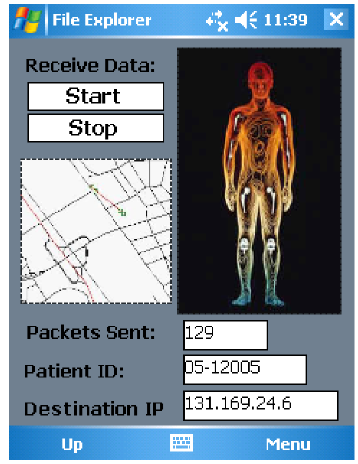

Fig. 2. The UI displays patient and cars' location in addition to number of transferred packets and destination IP address

4) Software running on Monitoring Station: In physician's office, monitoring station has the capability of playing collected data received from vehicles. The application shows graphics indicating the sensor readings over the time. Physicians can also use the same platform to initiate wearable systems' software reconfiguration to a desired new configuration using the same program. It is also important to note that the monitoring station can also be inside a vehicle, which is the part of the ad hoc network. This can be used to route the collected data in case of an emergency to the closest paramedic or doctor's car.

IV. Algorithms and Communication Protocol

The communication between vehicle and the patient is initiated while on-body terminal receives the advertisement message from vehicle. The advertisement message is composed of vehicle's IP address and GPS Information.

The mica2 mote, which is attached to the PC inside the vehicle will broadcast the advertisement message. Once advertisement message gets to the on-body terminal, it will decide to either initiate the connection or simply ignore the vehicles' existence. The gateway mote attached to the on-body terminal is responsible for receiving the advertised IP Address. The received IP address will be ignored if there exist no need for data transfer or the device is already connected to another vehicle.

If on-body terminal decides to connect to the vehicle, first it turns the radio on automatically, if it is disconnected. In order to decrease the power consumption of the on-body terminal, it is desired to have the wireless radio off. Since system needs only short amount of time to transfer the data to the cars, having wireless radio always on is unnecessary and will drain the battery of on-body terminal. Once the Radio is turned on, on-body terminal initiates a datagram connection [1] with vehicle using IP address received from advertisement message. Upon establishing the connection on-body terminal obtains the list of not delivered files, then it constructs the data packet by embedding the necessary information which will guarantee the accurate and complete content delivery.

We use UDP to deliver the message from on-body terminal to vehicle. In order to verify the success of message delivery, server running on the vehicle will send Ack message for each received meassage containing received message sequence. On-body terminal lunches a new process and waits for Ack packets. Once receiving the Ack packet, the appropriate file will be marked as delivered. Every time on-body terminal connects to vehicle to transfer data, it will first obtain the list of non delivered files and will start sending files with lower sequence number.

Once the on-body terminal decides to ignore an IP, it will queue the IP in the $ActiveIPList$. If the IP already exists in the $ActiveIPList$ the position of existing IP will be updated to the current position. When the on-body terminal get's disconnected from the car, system will choose the IP address of the vehicle that is located closer to the patients position. Every time system chooses an IP to connect to from $ActiveIPList$, if the time stamp of the last received is below some threshold value, it means that gateway mote has not received the IP broadcast of that vehicle for some time, therefore the IP will be dropped from $ActiveIPList$.

Data sent to each vehicle contains patient's ID, packet destination, packet number in addition to the sensed data. Patient ID is a number which is assigned to the patient using physicians. This number is unique for each person. Packet destination indicates the final destination where a given packet must be delivered. This destination can be either in the form of IP addresses, physical address or even an email Address. Packet number is a unique number, which shows the content of each packet belongs to which part of the original content.

Upon receiving each packet, the vehicle sends an acknowledgement message back to the on-body terminal as an indicator of successful delivery. The vehicle initiates a UDP session with on-body terminal and will send an acknowledgement message by including the received packet number. The on-body terminal will save all the acknowledgements that it receives from the vehicles and will retransmit lost packets when connection gets available.

The vehicles will use the destination address to send the uploaded content from patient to its final destination, either directly when they pass a wireless access point by pushing the data out, or by sending data to the other vehicles in case if they are going to an other direction.

V. Reconfigurable Platform

Deluge [9] is a network dissemination protocol, which provides the TinyOS full image upgrades over the air. Deluge requires tinyOS bootloader called TOSBoot to program / reprogram a node. Deluge allows a node in the network to install multiple images in the node simultaneously. It also provides the functionality to change the running image on the node [8]. Our proposed system has all the essential pieces necessary to use deluge. We install TOSBoot on each processing node, which have sensor attached. The medical monitoring application will be programmed by desired K number of images with different functionalities. If reconfiguration is necessary, the vehicles can remove an old image, load a new one and reboot the wearable processing node to run a new desired image.

Configuration	Min	Max	Avg	Std
1 vehicle 15 MPH	12	19	15.31	2.4
1 vehicle 25 MPH	4	9	5.78	1.78
3 vehicles 15 MPH	10	22	18.18	2.83
3 vehicles 25 MPH	4	10	7.62	1.54

TABLE I. Statistics on number of packets delivered on different speed with single and three ad hoc nodes

VI. Experiments

To test the functionality of the whole system we performed the following experiments. We attached a lightweight wearable system to the patient walking in the street. One flex sensor was attached to patient's knee and two accelerometer to patient's foot. The flex sensor is used to measure the angle of the knee once the patient is walking and the accelerometer can be used to compute the speed in which patient moves its foot and the legs movement. The patient carries an on-body terminal and a gateway mote attached to it. The gateway mote is responsible for data collection from patient and also vehicles' advertised IP address. The recorded data from accelerometer and flex sensor gets recorded in the size of 40KB chunks. Each chunk contains the sequence number, recording time and actual data. We use our personal vehicles to form an ad-hoc network. Each vehicle was equipped with a Dell laptop running windows XP, a Belkin wireless antenna, a GPS receiver and a mica2 mote.

We did our experiments in two different locations. One of them was performed with three cars in Westwood, California during the daytime and the other one took place in the Northridge California using only one car, during evening time. We chose the aforementioned locations to make sure we cover the system, in both, crowded and non crowded areas. The Westwood area is usually crowded and there exist tall buildings in the area, instead Northridge is a residential area (both are located in Los Angels County). In both setups we

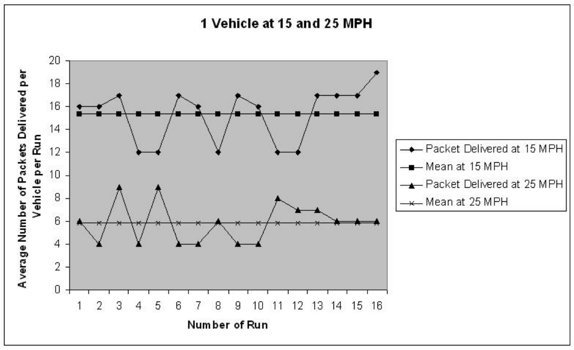

Fig. 3. 1 vehicle collecting data with 25 MPH and 15 MPH

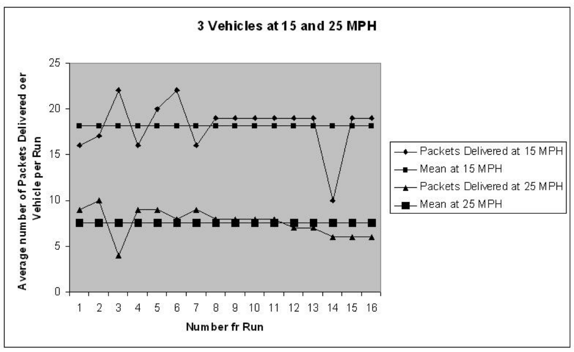

Fig. 4. 3 vehicles forming ad hoc network collecting data with 25 MPH and 15 MPH

passed the patient once with vehicle speed equal to 15 and 25 miles per hour. We passed patient in each of the configuration 15 times, 9 times in Westwood and 6 times in Northridge.

The result of our experiments indicates that the average amount of data collected using three vehicles that are in each others range is more than the number of packets collected using only one vehicle. The reason relies on the delay due to connection time. On average, it takes 4 seconds for a wireless to turn on and pocket PC connects itself to the vehicle. In case of three vehicles, the first vehicle receives less amount of data compared to the second and third vehicles. The average and standard deviation of the number of packets delivered is illustrated in table V. As shown in figure 3 and 4, the amount of transferred data sensibly decreases for a higher speed of the vehicles. This is mostly due to the delay between the triggering of the Wi-Fi and the actual initiation of the data transfer. In the worst case scenario (only one car at 25 mph) in one run we were still able to transfer on average 250KB, that could include several hours of sensed and preprocessed data. This could allow to perform the upload of data rarely providing a large time frame to let other patients upload their data.

VII. Future Work and Conclusion

In this paper we presented our feasibility study of using VANET as a data transfer medium among the physicians/care givers and patients, assessing what are the limits for data transfer from the wearable system used by patients to a vehicle that is passing by. As future work we intend to investigate the performances of dissemination protocols in order to assure fastest delivery of the patient data to the medical enterprise. Furthermore we intend to discover the best strategies in the choice of the vehicle to upload the data to (e.g. a vehicle that is moving towards an access point).

References

[1] http://www.faqs.org/rfcs/rfc768.html.

[2] Adaptive electrocardiogram feature extraction on distributed embedded systems. *IEEE Trans. Parallel Distrib. Syst.*, 17(8):797–807, 2006.

[3] V. Bychkovsky, K. Chen, M. Goraczko, H. Hu, B. Hull, A. Miu, E. Shih, Y. Zhang, H. Balakrishnan, and S. Madden. The cartel mobile sensor computing system. In *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 383–384, New York, NY, USA, 2006. ACM Press.

[4] F. Dabiri, H. Noshadi, H. Hagopian, T. Massey, and M. Sarrafzadeh. Light-weight medical bodynets. In *BodyNets 2007: Second International Conference on Body Area Networks*, 2007.

[5] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. E. Culler, and K. S. J. Pister. System architecture directions for networked sensors. In *Architectural Support for Programming Languages and Operating Systems*, pages 93–104, 2000.

[6] http://nescc.sourceforge.net/.

[7] http://www.xbow.com/.

[8] J. W. Hui and D. Culler. The dynamic behavior of a data dissemination protocol for network programming at scale. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 81–94. ACM Press, 2004.

[9] W. Hui and D. Culler. Deluge: Dissemination protocols for network reprogramming at scale.

[10] P. Jacquet, P. Mühlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized link state routing protocol for ad hoc networks. In *Proceedings of the 5th IEEE Multi Topic Conference (INMIC 2001)*, 2001.

[11] U. Lee, E. Magistretti, B. Zhou, M. Gerla, P. Bellavista, and A. Corradi. Mobeyes: Smart mobs for urban monitoring with a vehicular sensor network. In *IEEE Wireless Communications, Vol. 13, No. 5*, September 2006.

[12] A. Nandan, S. Tewari, S. Das, M. Gerla, and L. Kleinrock. Adtorrent: Delivering location cognizant advertisements to car networks.

[13] Q. Xu, T. Mak, J. Ko, and R. Sengupta. Vehicle-to-vehicle safety messaging in dsrc. In *VANET '04: Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, pages 19–28, New York, NY, USA, 2004. ACM Press.

[14] J. Yin, T. ElBatt, G. Yeung, B. Ryu, S. Habermas, H. Krishnan, and T. Talty. Performance evaluation of safety applications over dsrc vehicular ad hoc networks. In *VANET '04: Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, pages 1–9, New York, NY, USA, 2004. ACM Press.