Intelligent Mobile Data Mules for Cost-Efficient Sensor Data Collection

PREM PRAKASH JAYARAMAN
Caulfield School of Technology, Monash University, Melbourne, Australia
prem.jayaraman@infotech.monash.edu.au

ARKADY ZASLAVSKY[1], JERKER DELSING
Lulea University of Technology, Lulea, Sweden
{arkady.zaslavsky, jerker.delsing}@ltu.se

Sensor networks represent an important component of distributed pervasive infrastructure. A key challenge facing sensor networks is cost-efficient collection of data streaming from these distributed data sites. In this paper, we present a mobile data mule-based sensor data collection approach employing K-Nearest Neighbours queries. We propose a novel 3D-KNN algorithm that dynamically computes nearest sensors spread within a 3D environment around the data mule. The 3D-KNN algorithm incorporates a novel boundary estimation and neighbour selection algorithm to compute the nearest neighbour set. Further, we propose a neighbour prediction algorithm that computes sensor locations within the vicinity of the data mules' trajectory. We simulate the proposed 3D-KNN algorithm using GlomoSim validating its cost-efficiency by extensive evaluations. Results of our simulations conclude the paper.

Keywords: Sensor networks, query-based data collection, 3D-KNN algorithm,intelligent data mules

1. INTRODUCTION

Wireless sensor networks (WSN) have gained popularity in recent years with increased amount of research focus in this area [Culler et al. 2004; Shen et al. 2001]. Sensors are tiny battery-powered devices distributed within an area working together to achieve single or multiple goals [Shen, et al. 2001]. Sensor networks enable acquisition of data which previously was expensive, difficult or even impossible [Chu et al. 2006]. The inherent characteristics of sensor nodes represent them as a distributed source of data. This has led to wide acceptance of WSN across various application domains including military applications, environmental monitoring, habitat monitoring, logistic support, human-centric applications, smart homes etc [Arampatzis et al. 2005; Gharavi et al. 2003; Xu]. The exponential increase in WSN deployment has resulted in exponential increase in the amount of data generated by these smart sensing devices [Culler, et al. 2004; Shen, et al. 2001]. However, due to processing and battery limitations of sensor nodes, energy-efficient data collection from these distributed data sources has evolved into a key challenge. In this paper we propose a cost-efficient approach to collect sensor data using intelligent mobile devices namely "data mules". Advancement in technology and increased adaptation of mobility-based services/applications has led to an extremely large growth of mobile device population [Ahonen 2010; Alexander 2006]. These devices act as an excellent platform for processing and communication, thus creating a heterogeneous sensor network architecture- higher capacity mobile devices (data mules) interacting with low-powered sensor devices performing data dissemination, processing, collection and delivery. Our vision of employing day-to-day mobile devices as mobile data mules is presented in Figure 1. We propose a system framework that can be implemented on any mobile device platform enabling sensor discovery and data collection.

We extend our system framework to propose K-Nearest Neighbour (KNN) query-based solution

[1]Professor Arkady Zaslavsky also holds an Adjunct position at Monash University

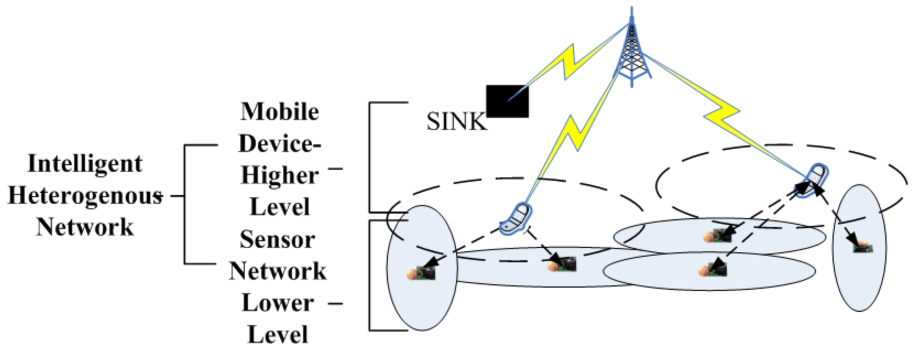

Figure. 1: Intelligent Heterogeneous Sensor Network

for cost-efficient sensor data collection. The mobile data mule employs KNN queries to dynamically compute a cost-efficient set of sensor nodes around it. Further, we propose a prediction algorithm to compute list of predicted neighbours along the data mules' path. The KNN computation is performed on-the-fly and does not require any global sensor topology information. Our proposed mobile data mule-based KNN algorithm is implemented over a three dimensional (3D) sensor space rather than most existing approaches that deal with two dimensional (2D) spaces. Our motivation to investigate a 3D sensor network originates from recent research that proves sensors are spatially distributed across 3D domains [Ganesan et al. 2004] as against 2D terrains assumptions in earlier works [Jea et al. 2005; Kansal et al. 2004; Liu et al. 2005]. We term our mobile data mules intelligent due to their ability to make data collection decisions on-the-run based on location, movement pattern and sensor information.

We use the term "cost-efficiency" to represent a function of cost parameters namely: (1) communication (energy); (2) query processing latency (performance); (3) overall network lifetime (total energy consumed). To the best of our knowledge, this work is a pioneering effort in exploring KNN query-based sensor data collection using mobile data mules in a three dimensional sensor network. The key contributions of this paper are:

➢ A system framework that enables mobile data mule-based data collection in sensor networks with no prior network topology knowledge and no special hardware requirements

➢ A 3D-KNN algorithm employed in a three dimensional space to compute cost-efficient set of nearest neighbours surrounding the data mule

➢ A neighbour prediction algorithm that further improves the cost-efficiency of the nearest neighbour set by computing future nearest sensor neighbours along the mobile data mules' path.

The rest of the paper is organized as follows. In section 2, we present related work. Section 3 presents a motivating scenario and the system framework for sensor data collection using intelligent mobile data mules. Section 4 presents the proposed 3D-KNN algorithm used to compute the k nearest neighbours surrounding the data mule and the neighbour prediction algorithm. Section 5 presents implementation details of the mobile data mule framework and evaluation results of 3D-KNN algorithm. Section 6 concludes the paper with remarks on future extensions.

2. RELATED WORK

Data collection approaches can be broadly classified into: (1) Data collection using static nodes (2) Data collection using mobile elements [Chakrabarti et al. 2003; Jain et al. 2006; Jea, et al. 2005; Kansal, et al. 2004; Ren et al. 2006; Shah et al. 2003; Somasundara et al. 2006]. Directed Diffusion is a popular static node-based data collection paradigm [Intanagonwiwat et al. 2003]. Directed Diffusion employs multi-hop data collection strategy, resulting in network flooding leading to broadcast storms [Tseng et al. 2002]. To overcome broadcast storms cluster-based approaches [Abbasi et al. 2007; Moussaoui et al. 2005; Younis et al. 2002] have been proposed. Though clustering reduces network flooding, the communication overheads involved in cluster maintenance and the resulting energy consumption is very high. Wireless transmission is the primary energy consuming operation in a sensor [Pottie et al. 2000]. Hence, reducing communication across the entire network is a key design consideration for data collection approaches. This design consideration has motivated the use of mobile data collectors for sensor data collection [Chakrabarti, et al. 2003; Jain, et al. 2006; Jea, et al. 2005; Kansal, et al. 2004; Ren, et al. 2006; Shah, et al. 2003; Somasundara, et al. 2006]. The literature further discusses mobility-based data collection approaches.

The use of mobility for data collection can be classified into: (1) Random Mobility (2) Predicted Mobility (3) Controlled Mobility. Shah et al. [2003] and Jain et al. [2006] propose data collection using random mobility. They propose a three- layered architecture over a two dimensional sensor network. The middle layer comprises mobile elements that are equipped with specialized hardware to communicate with sensor nodes in the surrounding. The data collectors presented in the paper are animals or cars that move around the sensor network terrain. Chakrabarthi et al. [2003] propose predicted mobility-based approach. Predicted mobility is the term used to describe mobile elements whose movement pattern is fixed and does not change. An observer (e.g. a shuttle bus) collects data from the sensor nodes while moving around the sensor field. The observer is mounted with special designed equipments to communicate with the sensor nodes. Kansal et al.[2004], Jea et al.[2005] and Somasundara et al. [2006] propose the use of controlled mobility for data collection. They employ specially designed mobile robots that move along a pre-defined path controlled by the user collecting sensor data. The performance improvements of employing mobile elements for data collection over static node-based approaches are evident from the literature. The above proposed approaches require specialized hardware for data collection based on application requirement in one way or another. This is not practical in real-world pervasive environments. Further, our proposal builds on using controlled mobility, except in our case, we explore a K-Nearest Neighbour query based mobile data collection over a three dimensional sensor network.

K-Nearest Neighbour (KNN) queries are a class of queries employed to retrieve spatially distributed data [Mouratidis et al. 2005; Mouratidis et al. 2005]. A sensor network is one typical example of spatially distributed data. KNN queries have been traditionally used in databases that requires maintenance of complex index structures to identify nearest neighbours [Liu, et al. 2005]. KNN query processing in sensor networks can be broadly classified into infrastructure-based and infrastructure-less approaches. The work presented in [Lee et al. 2005; Liu, et al. 2005; Soheili et al. 2005] are infrastructure-based approaches that work over fixed network dynamics requiring maintenance of complex index structures (network topology structure). Adapting this approach in sensor networks is expensive due to the amount of communication required to maintain the index structures. Winter et al. [2005] propose a partial infrastructure-based two dimensional KNN algorithm namely KBT. KBT uses "TreeHeight" [Winter, et al. 2005] to estimate the KNN boundary. The TreeHeight is a maximum hop distance that the KNN search query propagates from the point-of-interest[2] . It uses restricted flooding and timers to achieve energy-efficient query processing. The primary factor for energy consumption in KBT is flooding caused by the use of fixed hop count (determined by network topology information). Hence nodes that are not nearest neighbours deplete energy broadcasting messages. Wu et al. [2007] present an infrastructure-free itinerary based KNN (DKINN). The DKINN propagates the KNN query Q from the sink to a node closest to the point-of-interest. While the query routes to the nearest node, it collects network information along the path. This information is used to estimates the KNN boundary. The DKINN approach depends on pre-computed trajectory for data forwarding [Niculescu et al.

[2]The point-of-interest is a location within the sensor network. The K-Nearest neighbours are computed around the point-of-interest.

2003]. The trajectory provides the path the query needs to take to reach the node nearest to the point of interest. Sensors require specialized hardware (parallel array antennas) [Niculescu, et al. 2003] to pre-compute the trajectory which may not be always feasible. Hence DKINN applies only to a specific class of sensors.

The two approaches KBT [Winter, et al. 2005] and DKINN [Wu, et al. 2007] use the base station (sink) as a central point for query origination which reaches a sensor close to the point-of-interest. In our proposed approach, the mobile data mule is both the centre point for query origination and the point-of-interest. The mobile data mule employs KNN to select a subset of sensor nodes around it. The cost of collecting data from this subset is minimum compared to any other subset of nodes around the data mule. The novelty of our proposal lies in dynamically computing cost-efficient nearest sensors around the mobile data mules using 3D-KNN.

3. SENSOR DATA COLLECTION USING MOBILE DATA MULE: SYSTEM FRAMEWORK

3.1 Motivation

There are about 4 billion mobile phones[3][ITU 2008] currently in use in the world. The profuse existence of mobile phones contributes to a ubiquitous mobile data-access network. Most mobile phones have a talk time of two and a half hours with only 50% of the talk time being utilized leaving enough energy to perform other operations (processing/ communication). A survey con-

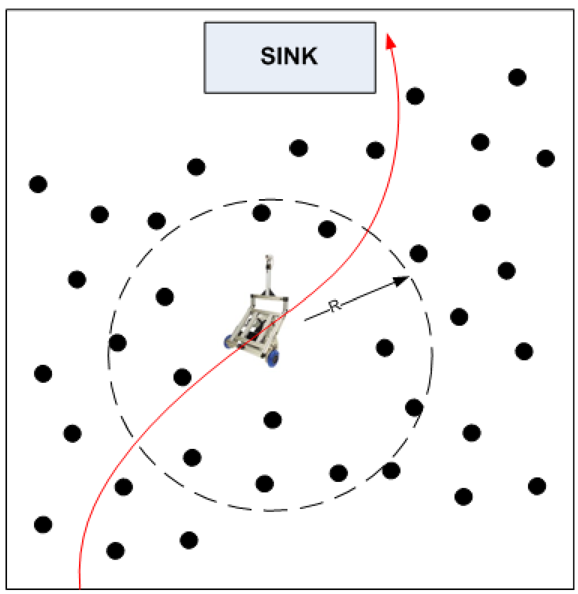

Figure. 2: A Scenario Description using Intelligent Mobile Data Mule

ducted over three days at info security 2006 at London [Alexander 2006], resulted in discovery of more than 2000 Bluetooth enabled devices in visible mode. This laid the foundation to our vision of using day-to-day computing devices as mobile data mules for sensor data collection. The use of such devices as mobile data collectors allow application developers to access sensor data which previously required specially designed data-sink infrastructure. Our approach is software-based allowing it to be implemented on any mobile devices with communication capability. This proposed approach can be extended to wider range of sensors with the feasibility of Zigbee based

[3]We use mobile phone as a classic example of widely available mobile device platform in day-to-day environments.

sensors [Zigbee August 2009] and Zigbee based mobile devices [Zigbee September, 2009]. A motivating scenario is depicted in figure 2. In the rest of the paper, we use the terms "data collector" and "data mule" synonymously to represent data collection using mobile devices.

3.2 System Architecture

Figure 3 illustrates the proposed system architecture. The system architecture has two components namely the Mobile-Device platform and the Data-Collection platform. The Mobile-Device platform comprises four main components namely: (1) mobile device-specific capability; (2) Location Manager; (3) Communication Manager; (4) Profile Manager.

The mobile-device-specific capabilities are features that are available on the mobile device

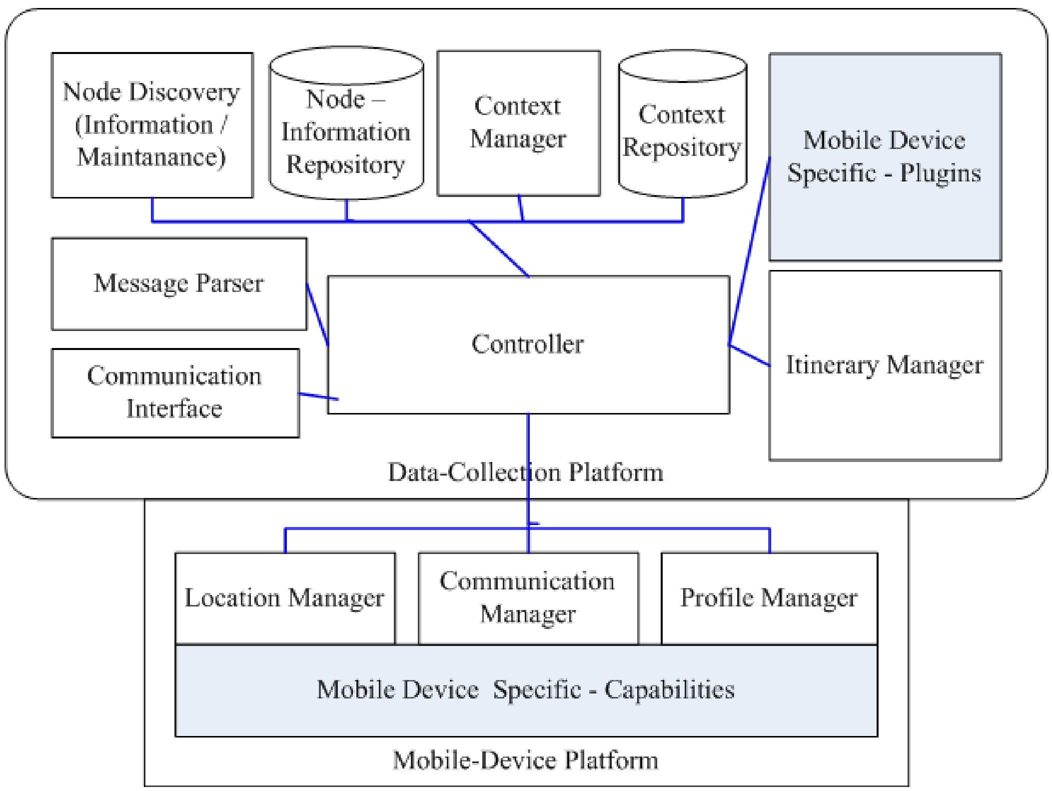

Figure. 3: Architecture of the system on the mobile node

platform. For example, the in-built GPS receiver of a mobile device is a mobile-device specific capability. The Location Manager performs the function of communicating with the device specific positioning hardware providing the Data-Collection platform with location information. The communication manager interfaces with the mobile device's communication hardware exposing them to the Data-Collection platform. E.g. if the mobile device is a mobile phone quipped with Bluetooth and Wireless LAN hardware, the communication manager exposes these two communication technologies to the Data-Collection platform. Finally the Profile Manager interfaces with mobile device's working states i.e. information that indicates if the mobile device is available to perform data collection operations.The Data-Collection platform comprises the Communication-Interface, Message-Parser, Node-Discovery, Context-Manager, Controller and Mobile Device-Specific plug-in. The communication-interface is responsible for deciding the communication mode required for node discovery, data collection and delivery. The message-parser is responsible for parsing messages that are exchanged between the data mule and the sensor node or other data mules in the surrounding. There are two types of messages, control and data

Algorithm 1 : Node Discovery/Management Module

Require: Node List L_N(Node Repository).

```
Obtain state from Profile-Manager
if state then
   scan for nodes into new list L_N(new)
   for each node n in L_N(new) do
      if n not exists in L_N then
         add n to L_N
      else
         Initiate connection with n
         if connection = false then
            inactive_count for n =+ 1
            if inactive_count = MAX_ALIVE then
               declare node inactive
            end if
         end if
      end if
   end for
end if
Update L_N with Sink
```

Figure. 4: Pseudo Code - Node Discovery/Management Module

messages. The control messages handle single bit messages used in collecting initial sensor information or instructing sensor to perform certain operation. For example, requesting the sensor to enter a sleep state to save energy. The data message is used to offload the data collected by the sensors. The node-discovery module handles node discovery and maintains a node repository. The node repository has the list of discovered nodes along with other node information required to make data collection decision. The node-discovery module synchronizes its information with the sink which is also updated by other data mules. The pseudo code of the node-discovery operation is presented in figure 4. The node management is capable of identifying inactive node. The value MAX_ALIVE in the node-discovery algorithm is an application defined constant that defines the number of attempts before which a mobile data mule declares a sensor node inactive. This creates node-inactivity problem i.e. a sleeping node might be reported inactive. To solve this, node information is synchronized with the centralised sink which solves the ambiguity between sleeping nodes and in-active nodes by comparing information obtained from other mobile data mules. The system framework is decentralised and hence has the capability to function without the availability of centralised sink. The context-manager handles context collection and storage that are used for computing data collection decisions. The context information includes location updates received from the Location Manager, mobile data mules' trajectory and sensor context information (Sensor sleep schedule, location, Received Signal Strength) obtained from node repository. The mobile-device-specific plug-in includes components that can be integrated into the data-collection platform based on mobile device specific capabilities. E.g. if the mobile device is a robot that moves within a building collecting sensor data, the mobile device-specific plug-in can incorporate an adaptive motion control algorithm that controls the speed of the mobile robot within the vicinity of sensor nodes. Finally the controller component which is the brain of the Data-Collector platform handles communication between each component and compute data collection strategies. It is responsible to discover, collect and deliver sensor data to the centralized sink.

4. 3D-KNN ALGORITHM

Section 3 presented our motivation and the system framework that can be implemented on any mobile device platform enabling intelligent sensor data collection. Our key contribution is

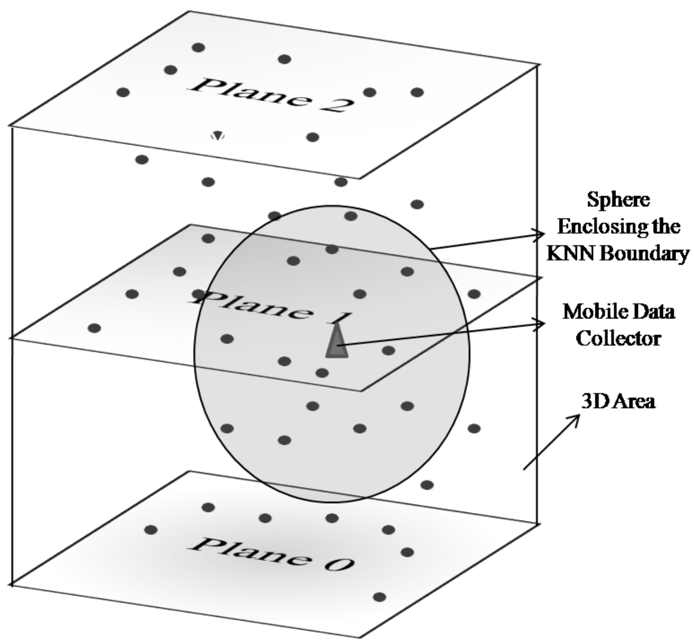

Figure. 5: A Three-Dimensional representation of the KNN Boundary Estimation Algorithm

employing KNN queries to compute cost-efficient set of sensors neighbouring the mobile data collector.

4.1 Definition and Network Model

An illustration of the 3D sensor network model with a mobile data collector is depicted in figure 5.

Definition 1: (K nearest neighbour): Given a set S of n sensor nodes, location L of mobile data collector, find a subset S' of k' nodes where S' < S and k' < n such that ∀ k'∈ S' and ∀ n ∈ S - S', DIST (k', L) < DIST (n,L) and DIST (k',L) ≤ KNN_BOUNDARY. The set S is the set of all nodes in the network. From the set S, we compute a subset S' of k' nearest neighbour that fall within the KNN boundary. For the rest of the paper, we use the notations S to represent the set of sensor nodes within the entire area and S' to represent the subset of nodes within the estimated KNN Boundary.

4.2 3D-K Nearest Neighbour

The 3D-KNN algorithm comprises three phases namely: (1) estimating KNN boundary (2) pre-routing or initial information collection and (3) plane rotation and cost-efficient nearest neighbours selection.

4.2.1 *3D-K Nearest Neighbour: Boundary Estimation Phase.* The KNN boundary estimation is one of the challenging steps in an infrastructure-less sensor network. In our model, the initial query Q is propagated from the mobile data mule which is located at the point-of-interest. The boundary estimation computes a sensor boundary around the data mules' location. The subset S' presented in the definition is depicted as a dotted circle in Figure 5. With the assumption that nodes are uniformly distributed within the 3D area, we use (1) to determine the density of nodes within the sensor network. We further use (1) to determine our KNN boundary (A) with known node-density $\mathrm{N_D}$ and sample size *k'*. Given a 3D building terrain, the computed KNN boundary is represented as a cube whose volume is less than or equal to the volume of the entire 3D deployment area. Since radio range is represented as a sphere with radius R, we use equation (2) to compute the radius of the sphere covering the *k'* nearest neighbours.

$$\text{Node Density } N_D(\text{per } m^3) = \frac{\text{No of nodes } n}{\text{Volume of the Terrain}} \tag{1}$$

$$\text{Radius } R = \sqrt{\frac{\text{Area A * 3}}{4 * \Pi}} \tag{2}$$

4.2.2 *3D-K Nearest Neighbour: Pre-Routing Phase.* Once the KNN search boundary is estimated, the pre-routing phase is employed to collect the following sensor information

- ➢ Node Location
- ➢ Signal-to-Noise Ratio (a cumulative function for each hop from mobile data collector to destination sensor node)

Given our assumption that no infrastructure information is required, the pre-routing phase is used to collect initial information required to compute the set K of k energy-efficient nearest neighbours such that K $\subseteq$ S'. Each node that receives the initial messages forwards it to its neighbours and sets a timer which is computed as a function of the KNN-BOUNDARY and the nodes distance from the mobile data collector. The timer is employed to achieve data aggregation saving additional communication. We use the notation K to represent the cost-efficient set of selected nearest neighbours from S'.

4.2.3 *3D-K Nearest Neighbour: Cost-Efficient Neighbour Selection.* At the completion of the pre-routing phase, initial node information i.e. node location and signal-to-noise ratio are available. We propose a plane rotation (mapping) algorithm that maps sensors in different planes onto a single reference plane. The mapping algorithm employs a metric based on sensor parameters namely signal-to-noise ratio (SNR) and distance to compute a single-valued output. We term this metric "KNN-METRIC". The KNN-METRIC provides a novel way of mapping sensors in different planes based on their characteristics including channel quality, interference, obstacles and distance. The technique of mapping sensors in different planes onto a single reference plane based on KNN-METRIC is called plane rotation. The reference plane is the plane on which the mobile data collector moves.

The KNN-METRIC is presented in (4). The distance D is computed using (3) while SNR is collected during pre-routing phase. The SNR and the distance parameters are inversely proportional i.e. higher SNR represents better channel quality and lesser energy consumption while greater distance estimates to higher energy consumption. c is a constant and α, β are pre-assigned weights. To compute a value for the constant c, we assume an ideal case (from simulation outcomes) where KNN-METRIC = 1, with $\alpha = 0.4$, $\beta = 0.6$ and ideal case values for R and SNR being 100 and 60 respectively. We determine c from (5).

$$\text{Distance } D = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2 + (Z_2 - Z_1)^2} \tag{3}$$

$$KNN - METRIC = c * \frac{\alpha * SNR}{\beta * \text{Distance } D} \tag{4}$$

$$c = 1 * \frac{0.6 * 100}{0.4 * 60} = 2.5 \tag{5}$$

The KNN-METRIC provides a value that can be used to map the sensors around the data mule based on sensor characteristics. This mapping is used to compute the set K that comprises k sensors that are close to the data mule and are cost-efficient such that $k \leq k'$ and K $\subseteq$ S'. Current related work focus on 2D planes relying only on Euclidian distance. Though the extension of distance-based approaches is straight-forward in 3D planes, it necessarily is not energy-efficient. For example, consider a sensor A at a distance d in a plane below the mobile data mule and sensor B at distance $d1$ in the same plane. Given the channel quality (computed using SNR) for B is better than A and distance $d1 > d$, the distance-based approaches would choose A over B while the KNN-METRIC would choose node B as B is the cost-efficient neighbour. Moreover, current approaches consider error-free communication channel which does not hold true in real-world scenarios. Our approach can be easily extended to incorporate more sensor metrics which can improve cost-efficiency of data collection. For example, including sensor residual energy parameters into the KNN-METRIC results in a set of sensors, whose distance from the data collector is minimum, have good communication channel quality and enough energy to successfully complete the data collection run. Further, the selection parameters can be correlated to improve sensor selection accuracy.

4.2.4 *3D-K Nearest Neighbour: Neighbour Prediction.* At the completion of the selection phase, the mobile data collector computes a set K of k nodes ordered by the KNN-METRIC. Based on our initial assumption that the mobile data collector is intelligent, the neighbour prediction algorithm iterates through the selected nodes computing the distance between each node k and mobile data collector's future locations (based on trajectory) i.e. locations during time T_1, T_2 ... Tx.

Definition 2: (Predicted Set of Nearest Neighbours P): The predicated set of nearest neighbours P is a set of p nodes where $P \times K$ such that $\forall p \in P$ and $\forall k \in K$, DIST (p, L_T) < DIST (p, L_{Ti}) where i = 1 to x

L_T represents the data-collectors current location at time T and L_{Ti} represents future locations at time T_1, T_2...T_X. A node p is said to belong to the set P if and only if its distance D from the data collector's current location L_T at time T is the less than its distances D_1,D_2, D_3... Dx at times T_1, T_2, T_3...T_X respectively. The rest of the nodes in the set K whose distances are not minimal at time T are added to mobile data collector's predicted next hop collection set K_P. The set P is an optimized energy-efficient set of sensor nodes from which data is collected by spending minimal overall energy. The prediction algorithm computes distance between sensor locations (obtained from KNN query) and mobile data mules' future locations. Hence actual SNR values may not be available. Hence the prediction algorithm ignores the SNR values while computing the predicted set P. Once the mobile collector reaches the next location Lx at time Tx, it piggy-backs the list of nodes whose distances were pre-computed with the new KNN pre-routing broadcast message. Each node receiving the messages checks its nodeID. If the nodeID is not found, it employs the KNN pre-routing phase to propagate the message to its neighbours. If the nodeID exists in the list, it checks for any neighbours that are part of the list. If a neighbour exists, it forwards the message to the neighbours and sets a timer. On timer expiry, the aggregated data is returned to the mobile data collector. If none of its neighbours exist in the list, the sensor responds with the actual data that needs to be delivered. The 3D-KNN neighbour prediction algorithm is presented in Figure 6.

Algorithm 2 : 3DKNN Algorithm to predict energy efficient set of neighbours

Require: K, set of k nearest nodes, mobile nodes future Location in a Vector V, sensor node Location loc_k.

for each node k in K **do**
  $current\_metric = current\_metric$
  for each L in V **do**
    Compute a max flow value between two nodes.
    Assign the computed max flow value as an epitome
  end for
  if $min\_metric = current\_metric$ **then**
    Add node to P
  else
    Add node to P_Ti
  end if
end for

Figure. 6: 3DKNN Algorithm to predict energy efficient set of neighbours

5. IMPLEMENTATION AND EVALUATION

5.1 Prototype System Implementation

The prototype of the mobile data mule was implemented on a robot controlled by a laptop equipped with 802.11 and Bluetooth communication capabilities. The robot used for implementation is a commercially available off-the-shelf robot (ER1) developed by Evolution Robotics [Robotics 2010]. ER1 has a set of programmable APIs that provide an interface to control the drive-motors, web-cam and the IR-sensors. The implemented system framework on the robot was tested in a building. The robot has the ability to navigate itself using localisation information obtained from Ekahau positioning engine (EPE) [Ekahau]. EPE is a software based location system that provides location information using triangulation technique. It triangulates location information by measuring signal strengths between access point and Wi-Fi card equipped on the robot. The data collection framework was implemented using Java on the laptop. We use the BlueCove [BlueCove], Bluetooth programmable interface to facilitate communication between data mule and sensors.

The sensor node used for the prototype implementation is the Mulle [Eliasson et al. 2008]. Mulle is a generic wireless sensor node using Bluetooth for communication, developed at EISLAB, Lulea University of Technology, Sweden. It is powered by a lithium ion battery ranging from 120 mAh to 2200 mAh. The Mulle can work with variety of sensing devices that can be connected to the expansion port. The sensors are named in the format <*sensor-id, location*> . This enables the mobile data mule to collect sensor information like name, location and received signal strength (RSSI) by performing a Bluetooth inquiry rather than establishing a connection. We use Bluetooth serial port profile for data exchange. The reason behind using serial port profile is the simplicity of setting up and using the RFCOMM between data mule and Mulle. The data received from the Mulle is time stamped and stored in the data mule which is later delivered to the base station. Our framework can be applied directly to any mobile devices with varying communication capabilities e.g. mobile phones can use GPRS to deliver data to the base station. Figure 7 presents screen shots of our prototype implementation.

The prototype presented is a proof-of-concept implementation employing mobile data mules as sensor data collectors within a Bluetooth based sensor infrastructure. To validate our proposed KNN query based cost-efficient data collection, we have implemented our data collection framework in GlomoSim [Gerla et al. 1999], a sensor network simulator. The simulation and evaluation details are presented in the following section.

Figure. 7: The Data Mule Framework implemented on a mobile robot platform and Screenshot of the Data Collection interface running on the Data Mule

5.2 Simulation Evaluations

To prove the cost-efficiency of the proposed KNN query-based data collection using mobile data mules, we validate the efficiency of our proposed 3D-KNN algorithm. The 3D-KNN algorithm has been implemented in a GlomoSim [Gerla, et al. 1999]. GlomoSim is a parallel discrete-event system simulator based on Parsec [Parsec]. GlomoSim has the features of simulating mobile wireless sensor nodes. It incorporates a number of mobility models used to implement the mobile data mules movement within the sensor network. For the simulation we use the parameters presented in table 1.

Table I: Simulation Parameters

Parameter	**Value**
Number of Nodes (N)	20 to 200.
Area Size (A_T)	1000 x 1000 x 1000
Radio T_x Power	15 dBm
k'	Number of Nearest nodes
c	2.5

Our sensor network is assumed to be distributed within a building (3D area). Neighbour discovery is done at runtime by the sensors, though it is a not a requirement to process the KNN queries. This approach allows the 3D-KNN algorithm to adapt to changing network infrastructure. The sensor nodes are assumed to be static with the mobile data collector moving along a known path. To validate the performance of the proposed 3D-KNN algorithm, we evaluate the algorithm over three criteria's namely: boundary estimation, query processing efficiency and energy consumption.

5.3 Boundary Estimation Algorithm Evaluation

The KNN query is disseminated into the sensor network by the mobile data collector. The boundary area is estimated by the mobile data collector based on the sample size *k'*. It is important to evaluate the boundary estimation algorithm as the boundary area computed needs to cover at least *k'* sensor nodes. The result of evaluation is presented in Figure 8. As the result show, for given sample size *k'*, the estimated boundary covers more than required *k'* sensor nodes. Moreover, the key performance outcome is the size of the set S' (subset of N) which is contained

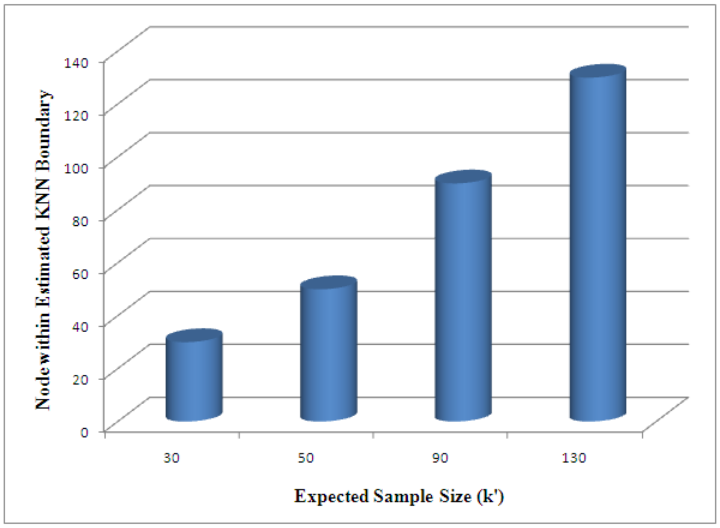

Figure. 8: Nodes Found within the KNN Boundary for request K using 3D-KNN algorithm

to cover at least *k'* sensors rather than covering the entire sensor network. This validates the efficiency of using node density-based sensor boundary estimation approach employed by the proposed 3D-KNN algorithm.

5.4 Query Latency

The query latency is the time taken to process a KNN query for varying sizes of *k'*. To validate the query performance of the proposed 3D-KNN algorithm, we compare the results of our simulations with KBT [Winter, et al. 2005]. As mentioned in the literature, KBT employs fixed tree height and hence does not adapt well for changing network topologies. The result of our simulation

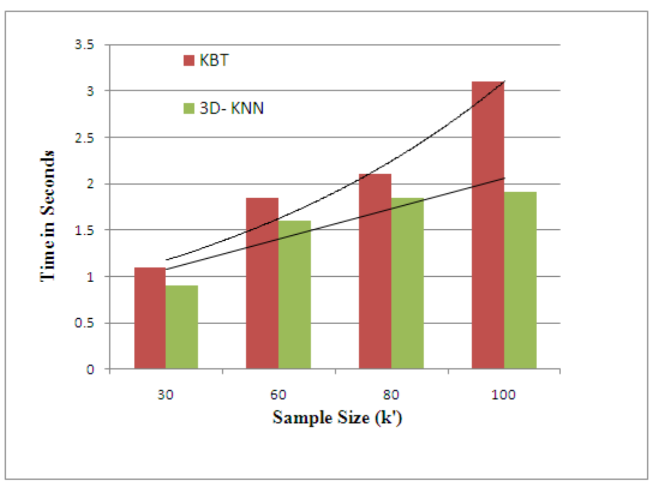

Figure. 9: 3D-KNN vs. KBT: Query Latency Comparison

is presented in Figure 9. A trend line projected on outcomes show exponential increase in the query processing time for KBT while 3D-KNN is more linear. This is primarily attributed to the efficient boundary estimation that adapts its outcomes based on sample size *k'*.

5.5 Energy Consumption

The energy consumption using the proposed 3D-KNN algorithm to facilitate mobile data collection is a key metric to validate the proposed approach of using KNN to collector sensor data. The mobile node in this simulation moves to a location, initiates a KNN query, computes the cost-efficient set of k sensor nodes and collects sensor data from them. The mobility-based approaches previously discussed in the literature do not explore KNN query-based cost-efficient data collection area computation. Hence, we compare the result of our simulation with static

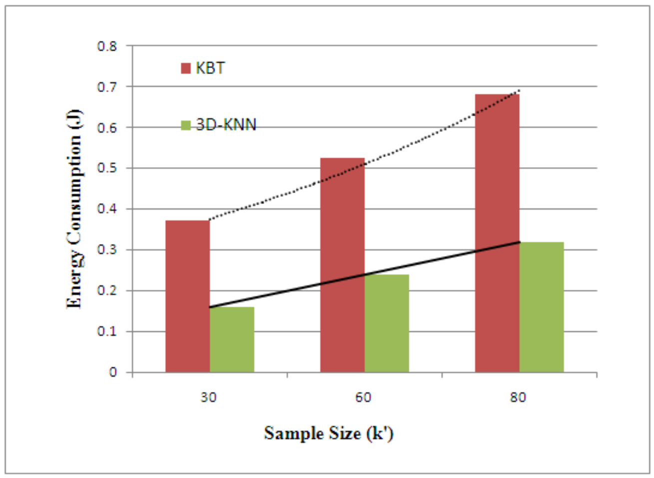

Figure. 10: -KNN vs. KBT: Energy Consumption Comparison

KNN query processing approach KBT [Winter, et al. 2005].The result of the simulation is presented in figure 10. Trend lines projected over the results indicate exponential increase in energy consumption using KBT while 3D-KNN based data collection has a linear increase. The energy efficiency of the proposed 3D-KNN algorithm is primarily attributed to: 1) Proposed boundary estimation based on network density that covers at least k' sensors hence reducing the amount of message broadcast in the network; 2) Plane rotation and k nearest neighbour selection based on the KNN-METRIC choosing sensor nodes that are both closer and energy-efficient (better communication channel). To further evaluate the plane rotation and neighbour selection based on the proposed KNN-METRIC, we evaluate the proposed neighbour selection algorithm (employed by the mobile collector to compute dynamic cost-efficient collection area comprising k sensors) against a basic implementation of KNN query processing algorithm.The basic implementation does not incorporate SNR heuristics while computing nearest neighbours and considers only Euclidean distance. We have implemented the fixed tree height approach employed by KBT in the basic KNN approach. The result of this simulation is presented in Figure 11. The highlighted part of the graph illustrated by dark circles indicates the metric values computed using the two approaches. Investigating the results, node 9 which is at a longer distance than 8 (deduced from basic KNN metric) would be much energy efficient nearest neighbour as its SNR and distance metric is higher than node 8. The similar outcome is observed for nodes 5 and 6. This evaluation proves and validates the energy efficient k nearest neighbour selection employed by the mobile data collector.

5.6 Energy Consumption of Individual Sensor Node

Figure 12 presents the result of the energy consumption of individual sensor nodes after a round of data collection using the mobile data collector. The mobile data collector employs the 3D-KNN algorithm to compute a dynamic collection area. The simulation was run over a 1000 x 1000 x 1000 area with 80 sensor nodes and the mobile data collector issuing KNN request from

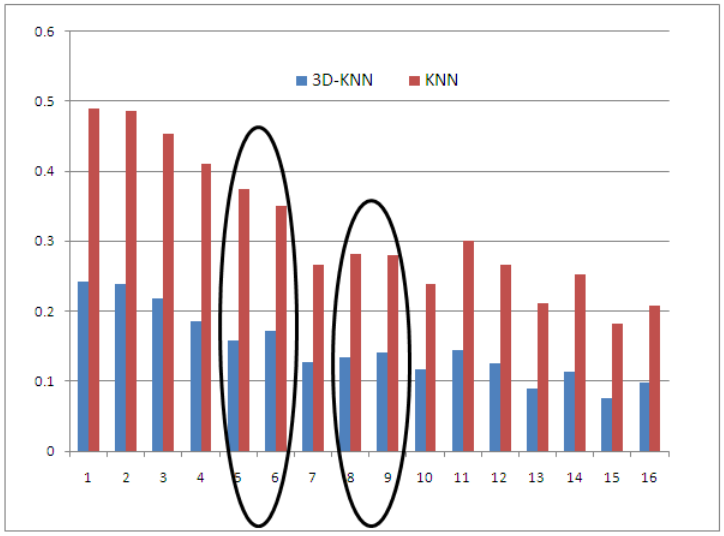

Figure. 11: 3D-KNN using proposed heuristics (SNR, Distance) vs. basic KNN Implementation

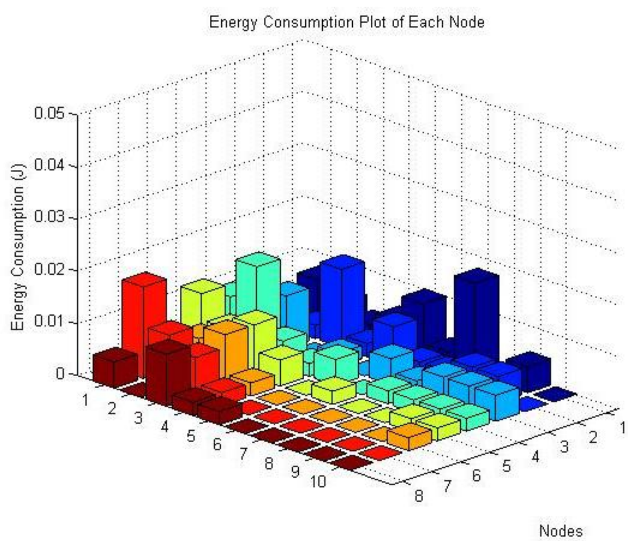

Figure. 12: Energy Consumption at each node employing the 3D-KNN algorithm

different locations within the sensor network with $k = 10$. The simulation did not use a grid based approach and hence the plots X, Y does not represent exact nodes locations. We used a 3D bar chart generated using MATLABS to present the uniform and efficient depletion of sensor energy across the entire network. The nodes with nil energy usage were the border nodes which did not fall within the path of the mobile data collector. The simulation results presented was averaged from 10 simulation runs. The result presented in figure 12 further validates the energy efficiency of the proposed 3D-KNN based mobile data collection approach.

5.7 Energy Consumption with Neighbour Prediction

The energy consumption results presented in Figure 10 and Figure 12 does not employ the proposed neighbour prediction algorithm. To validate the proposed neighbour prediction algorithm, we compare the overall energy consumed by the entire sensor network over a single mobile data

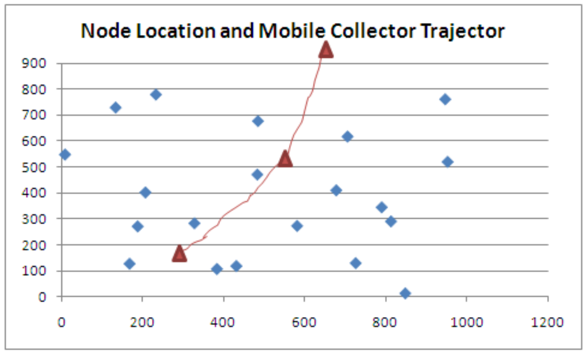

Figure. 13: Simulation Setup with Nodes and Mobile Collector Trajectory

collection cycle with and without neighbour prediction. The prediction algorithm is employed by the mobile data collector over the result-set obtained from the KNN query. Figure 13 gives an overview of the simulation setup with the mobile data collector moving in a known trajectory stopping at positions identified by the triangle.The prediction algorithm uses the mobile data collector's future locations at time T_1, T_2... T_N to compute nodes distances from the data collectors future locations. The proposed algorithm's energy efficiency is validated by the simulation results in figure 14. The outcome clearly exhibits great energy savings using neighbour prediction

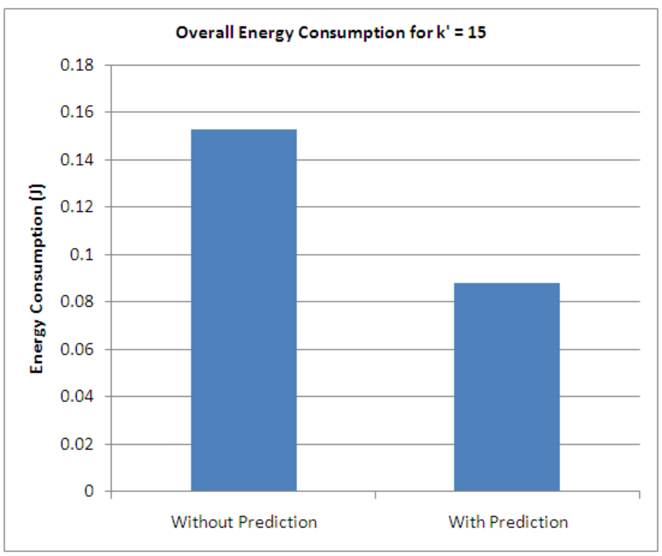

Figure. 14: Energy Consumption output with and without nearest neighbour prediction

attributed to the pre-computation of future sensor neighbours within the vicinity of the mobile data mules' path. Employing the prediction algorithm allows the mobile data collector to further optimize the energy consumed during data collection. This is validated by the simulation outcomes.

6. CONCLUSION

In this paper we have proposed a novel 3D-KNN query-based cost-efficient approach for sensor data collection using intelligent mobile data mules. The 3D-KNN algorithm accounts for sensors distributed across three dimensional spaces taking sensor characteristics into consideration while computing data collection decisions. The mobile data mule employs the 3D-KNN algorithm to compute a cost-efficient set of sensors surrounding it. The overall energy spent in collecting data from this cost-efficient set of sensors is minimal. Further, the data mule employs neighbour prediction algorithm that enables it to determine future sensors location information along its path. The proposed 3D-KNN algorithm requires no prior network topology knowledge. The proposed system framework has been implemented on a mobile robot based data mule as a proof-of-concept. The 3D-KNN based sensor data collection algorithm has been simulated and its cost-efficiency over large-scale sensor deployments has been validated. The 3D-KNN algorithm clearly has improved performance compared to KBT, hence validating our proposal of using KNN queries as a cost-efficient sensor data collection approach. We have also validated the proposed neighbour prediction algorithm. The result of the predicted approach shows clear savings in energy compared to the non-predicted approach. The results are promising and we look to extend our work by incorporating sensor errors (location error, SNR error) that can influence data collection decision. We would like to model and simulate the influence of these errors in real-world situations and explore the benefits of incorporating these errors for sensor data collection.

REFERENCES

Abbasi, A.A., and Younis, M. 2007. A survey on clustering algorithms for wireless sensor networks. *Computer Communications*, 30, 2826-2841.

Ahonen, T.T Explaining 4.6 billion mobile phone subscriptions on the planet *http://communities-dominate.blogs.com/brands/2009/11/explaining-46-billion-mobile-phone-subscriptions-on-the-planet.html* (Accessed: March 2010)

Alexander, G Bluetooth: London 2006. *http://www.viruslist.com/en/analysis?pubid=188833782* (Accessed: August 2006)

Arampatzis, T., Lygeros, J., and Manesis, S. 2005. A Survey of Applications of Wireless Sensors and Wireless Sensor Networks. In *Intelligent Control, 2005. Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation*, 719-724.

BlueCove *http://sourceforge.net/projects/bluecove* (Accessed: Jan 2007)

Chakrabarti, A., Sabharwal, A., and Aazhang, B. 2003. Using Predictable Observer Mobility for Power Efficient Design of Sensor Networks. In *Information Processing in Sensor Networks*, 552-552.

Chu, D., Deshpande, A.,Hellerstein, J.M., and Hong, W. 2006. Approximate Data Collection in Sensor Networks using Probabilistic Models. In *proceedings of the 22nd International Conference on Data Engineering, IEEE Computer Society.*

Culler, D., Estrin, D., and Srivastava, M.B. 2004. Guest Editors' Introduction: Overview of Sensor Networks. *Computer*, 37, 41-49.

Ekahau WiFi-based tracking *http://www.ekahau.com/* (Accessed: Feb 2010)

Eliasson, J., Lindgren, P., and Delsing, J. 2008. A Bluetooth-based Sensor Node for Low-Power Ad Hoc Networks. *Journal of Computers*, 3, 1-10.

Evolution-Robotics *http://www.evolution.com/* (Accessed: August 2009)

Ganesan, D., Ratnasamy, S., Wang, H., and Estrin, D. 2004. Coping with irregular spatio-temporal sampling in sensor networks. *SIGCOMM Comput. Commun. Rev.*, 34, 125-130.

Gerla, M., Bajaj, L., Takai, M., Ahuja, R., and Bagodia, R. 1999. GloMoSim: A Scalable Network Simulation Environment. In *Technical Report 990027*, University of California.

Gharavi, H., and Kumar, S.P. 2003. Special issue on sensor networks and applications. *proceedings of the IEEE*, 91, 1151-1153.

Intanagonwiwat, C., Govindan, R.,Estrin, D.,Heidemann, J. , and Silva, F. 2003. Directed diffusion for wireless sensor networking. *IEEE/ACM Trans. Netw.*, 11, 2-16.

ITU Corporate Annual Report *http://www.itu.int/dms_pub/itu-s/opb/conf/S-CONF-AREP-2008-E06-PDF-E.pdf* (Accessed: Jan 2010)

Jain, S., Shah, R.C., Brunette, W., Borriello, G., and Roy, S. 2006. Exploiting mobility for energy efficient data collection in wireless sensor networks. *Mob. Netw. Appl.*, 11, 327-339.

Jea, D., Somasundara, A., and Srivastava, M.B. 2005. Multiple controlled mobile elements (data mules) for data collection in sensor networks. In *proceedings of the IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS)*, Marina del Rey, California, USA 2005, 244–257.

Kansal, A., Somasundara, A., Jea, D., Srivastava, M.B., and Estrin, D. 2004. Intelligent fluid infrastructure for embedded networks. In *proceedings of the 2nd International Conference on Mobile systems, applications, and services*, Boston, MA, USA 2004 ACM.

LEE, W.C., AND ZHENG, B. 2005. DSI: A Fully Distributed Spatial Index for Location-Based Wireless Broadcast Services. In *proceedings of the 25th IEEE International Conference on Distributed Computing Systems, IEEE Computer Society.*

LIU, B., LEE, W.C., AND LEE, D.L. 2005. Distributed caching of multi-dimensional data in mobile environments. In *proceedings of the 6th International Conference on Mobile data management*, Ayia Napa, Cyprus 2005 ACM.

MOURATIDIS, K., PAPADIAS, D.,BAKIRAS, S.,AND TAO, Y. 2005. A Threshold-Based Algorithm for Continuous Monitoring of k Nearest Neighbors. *IEEE Trans. on Knowl. and Data Eng.*, 17, 1451-1464.

MOURATIDIS, K.,PAPADIAS, D., AND HADJIELEFTHERIOU, M. 2005. Conceptual partitioning: an efficient method for continuous nearest neighbor monitoring. In *proceedings of the 2005 ACM SIGMOD International Conference on Management of data*, Baltimore, Maryland 2005 ACM.

MOUSSAOUI, O. AND NAMI, M. 2005. A distributed energy aware routing protocol for wireless sensor networks. In *proceedings of the 2nd ACM International Workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks*, Montreal, Quebec, Canada 2005 ACM.

NICULESCU, D., AND NATH, B. 2003. Trajectory based forwarding and its applications. In *proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, San Diego, CA, USA 2003 ACM.

PARSEC UCLA *http://pcl.cs.ucla.edu/projects/parsec* (Accessed: August 2009)

POTTIE, G.J., AND KAISER, W.J. 2000. Wireless integrated network sensors. *Commun. ACM*, 43, 51-58.

REN, B., MA, J., AND CHEN, C. 2006. The hybrid mobile wireless sensor networks for data gathering. In *proceedings of the 2006 International Conference on Wireless Communications and Mobile Computing*, Vancouver, British Columbia, Canada 2006 ACM.

Evolution Robotics *http://www.evolution.com/* (Accessed: August 2009)

SHAH, R.C., ROY, S., JAIN, S., AND BRUNETTE, W. 2003. Data MULEs: Modeling a Three-tier Architecture for Sparse Sensor Networks. In *proceedings of the IEEE SNPA Workshop*, 30–41.

SHEN, C.C., SRISATHAPORNPHA, C., AND JAIKAEO, C. 2001. Sensor information networking architecture and applications. Personal Communications, *IEEE*, 8, 52-59.

SOHEILI, A.,KALOGERAKI, V., AND GUNOPULOS, D. 2005. Spatial queries in sensor networks. In *proceedings of the 13th annual ACM International Workshop on Geographic Information Systems*, Bremen, Germany 2005 ACM.

SOMASUNDARA, A., KANSAL, A., JEA, D.D., ESTRIN, D., AND SRIVASTAVA, M.B. 2006. Controllably Mobile Infrastructure for Low Energy Embedded Networks. *IEEE Transactions on Mobile Computing*, 5, 958-973.

TSENG, Y.C., NI, S.Y., CHEN, Y.S., AND SHEU, J.P. 2002. The broadcast storm problem in a mobile ad hoc network. *Wirel. Netw.*, 8, 153-167.

WINTER, J., XU, Y., AND LEE, W.C. 2005. Energy Efficient Processing of K Nearest Neighbor Queries in Location-aware Sensor Networks. In *proceedings of the The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, IEEE Computer Society.*

WU, S.H., CHUANG, K.T., CHEN, C.M., AND CHEN, M.S. 2007. DIKNN: An Itinerary-based KNN Query Processing Algorithm for Mobile Sensor Networks. In *Data Engineering. IEEE 23rd International Conference on*, 456-465.

A Survey of Sensor Network Applications *http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.131.9647&rep=rep1&type=pdf* (Accessed: March 2008)

YOUNIS, M., YOUSSEF, M., AND ARISHA, K. 2002. Energy-Aware Routing in Cluster-Based Sensor Networks. In *proceedings of the 10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, IEEE Computer Society.*

ZIGBEE connects to mobile phones *http://www.automation.com/content/zigbee-connects-to-mobile-phones* (Accessed: August 2009)

ZIGBEE Alliance, ZigBee Wireless Sensor Applications for Health, Wellness and Fitness *http://www.zigbee.org/imwp/download.asp?ContentID=15585* (Accessed: September 2009)

Prem Prakash Jayaraman is currently a PhD candidate in Caulfield School of Information Technology, Monash University, Australia. He received the Bachelors in Information Technology degree from Madras University, India in 2003. In 2006, he received the Masters degree in Network Computing from Monash University, Australia. During 2006, he was employed part-time as a research assistant at Monash University working on projects under CoolCampus initiative. His current research interests are wireless sensor networks, context aware computing and mobility based applications.

Arkady Zaslavsky is holding a Personal Chair (Chaired Professor) at Lule University of Technology, Sweden. He worked at Monash University, Australia before joining LTU. He received MSc in Applied Mathematics majoring in Computer Science from Tbilisi State University (Georgia, USSR) in 1976 and PhD in Computer Science from the Moscow Institute for Control Sciences (IPU-IAT), USSR Academy of Sciences in 1987. Arkady Zaslavsky has published more than 240 research publications throughout his professional career. He organized and chaired many workshops and conferences in mobile computing area, including "Mobility in Databases and Distributed Systems" and International Conference on Mobile Data Management, MDM2003. He is a "Distributed databases" area editor for IEEE Computing-Online. His research interests include mobile and pervasive computing; distributed and mobile agents and objects; wireless networks; distributed computing and database systems; distributed object technology and mobile commerce. Arkady Zaslavsky has been awarded and involved in many research grants and projects including DSTC's "M3: Enterprise Architecture for Mobile Computation", "Context-rich mobile agent technology to support information needs of financial institutions", "Adaptive Distributed Information Services", "Mobile City" and others. He is a member of ACM,IEEE Computer and Communications Societies.

Prof. Delsing received the M.Sc. in Engineering Physics at Lund Institute of Technology, Sweden 1982. In 1988 he received the PhD. degree in Electrical Measurement at the Lund University. During 1985 - 1988 he worked part time at Alfa-Laval - SattControl with development of sensors and measurement technology. In 1994 he got the docent degree (associate professor) in Heat and Power Engineering. Early 1995 he was appointed full professor in Industrial Electronics at Lulea University of Technology where he currently is working as the scientific head of EISLAB, http://www.ltu.se/eislab. For the period 2004-2006 he also served as Dean of the engineering faculty at Lulea University of Technology. He has a long standing in ultrasound sensor technology in particularly applied to flow sensing. His present research profile can be entitled "Embedded Internet Sensors", EIS, with applications to industry, medicine and sport. The general idea is that most sensors and actuator will have communication capability using the Internet and the "TCP/IP" protocol suite and be capable of ah-hoc integration into a communication network and an application framework. Since 1999 he is chairman of ITF (Instrument Tekniska Foreningen/Instrument Society of Sweden).