A Two Round Reporting Approach to Energy Efficient Interpolation of Sensor Fields

Brian Harrington[1] and Yan Huang[2]

[1] Yahoo! Corporation *
{brh}@yahoo-inc.com
[2] University of North Texas
{huangyan}@cs.unt.edu **

Abstract. In-network aggregation has been proposed as one of the main mechanisms for reducing messaging cost (and thus energy) in prior sensor network database research. However, aggregated values of a sensor field are of limited use in natural science domains because many phenomena, e.g., temperature and soil moisture, are actually continuous and thus best represented as a continuous surface over the sensor fields. Energy efficient collection of readings from all sensors became a focus in recent research literature. In this paper, we address the problem of interpolating maps from sensor fields.

We propose a spatial autocorrelation aware, energy efficient, and error bounded framework for interpolating maps from sensor fields. Our work is inspired by spatial autocorrelation based interpolation models commonly used in natural science domains, e.g., kriging, and brings together several innovations. We propose a two round reporting framework that utilizes spatial interpolation models to reduce communication costs and enforce error control. The framework employs a simple and low overhead in-network coordination among sensors for selecting reporting sensors so that the coordination overhead does not eclipse the communication savings. We conducted extensive experiments using data from a real-world sensor network deployment and a large Asian temperature dataset to show that the proposed framework significantly reduces messaging costs.

1 Introduction

Sensor networks are expected to form a digital nervous system embedded in physical spaces to extend human beings' "tactile" sensations to every corner of the world. In recent years, coin-to-palm sized, programmable sensors have begun to be able to locate their positions, self-organize into a network, and communicate through multi-hop protocols with a gateway which incorporates long-haul communication capacity. This enables the deployments of robust distributed networks of hundreds to thousands of sensors to interact with the physical world.

[1] The work was completed while at the University of North Texas.

[2] This work was partially supported by the Texas Advanced Research Program under Grant No. 003594-0010-2006 and by the National Science Foundation under Grant No. OCI-0636421 and Grant No. CNS-0709285.

The feasibility of abstracting a sensor network as a database has been documented and prototyped in pioneer sensor database systems [4, 29, 22]. In an acquisitional sensor network database, a collection of sensors of the same type may be treated as a table, e.g. $lightSensors$, in a database. The rows of the table are distributed among sensors in a physical space. Each sensor generates records in the format of $< sensorID, reading, time >$. Users can interact with the network using declarative database query languages. The query is inserted into the network by either broadcasting or targeted routing. For example, users can issue queries such as "*SELECT avg(readings) FROM lightSensors WHERE location IN P EVERY 5 seconds*" where P is a given polygon.

On the one hand, full duty cycle operations on a sensor node, e.g. Berkeley Mica Motes, will deplete its energy supply in a few days. In-network aggregation [11, 25] has been proposed as one of the main mechanisms to reduce messaging cost due to the fact that communication is much more expensive than computation in sensors. For example, the power consumed by a sensor to transmit 1 bit of data is equivalent to 220 - 2,900 instructions on different architectures [29]. On the other hand, the average/sum/max readings of a spatial region is often of limited use to domain scientists. Many phenomena in natural science, e.g., temperature, hydraulic head, soil moisture, and ocean current velocity, are actually continuous, and thus best represented as a continuous surface over the sensor fields. In fact, a raster surface map, e.g. soil moisture map, is frequently created by domain scientists using interpolation upon receiving readings from sensors, and is fundamental for many subsequent field based operations. Energy efficient algorithms that allow domain scientists to interpolate the surface/map of the sensor fields for months to years are critical to future large scale deployments of sensor networks.

In this paper, we address the problem of interpolating maps from sensor fields. The naive way to interpolate the continuous spatial phenomena at the sink is to have all sensors report their readings and perform an interpolation at the sink. This approach depletes the energy of the sensor network very quickly. An alternative way is to utilize spatial autocorrelation to select a subset of sensors to report, and then use them to interpolate a map at the sink. However, for the sensors that do not report, the estimation should be under a user given error bound. Error bounded data collection is sufficient, especially considering that for mapping interpolation residual errors are generally considered common. This problem is also referred to as the "SELECT * " problem with error bound in recent research publications [5]. In this paper, we focus on utilizing spatial autocorrelation to perform energy efficient and error bounded map interpolation.

One way to utilize spatial autocorrelation is to divide the sensors into groups and let the group leader aggregate/select the readings to report in the group and represent the whole group. Unfortunately, the group leader selection process is non-trivial and usually incurs substantial messaging cost if done dynamically in the sensor field. In many cases, the benefit of grouping can not offset the overhead of dynamic group leader selection. When the group selection is static and once-

for-all, the grouping may not be able to adapt to the dynamic topological changes in the field and frequent sensor failures.

We propose a spatial autocorrelation aware, energy efficient, and yet error bounded framework for interpolating maps from sensor fields. The framework utilizes a simple probabilistic selection process to determine the sensors that need to report in the first round and relies on a second round to control reporting errors for all other sensors. The error bound is achieved in second round by pushing spatial interpolation models used by the sink to sensors in the field allowing the sensors to make decisions on the importance of their readings. The model utilizes qualitative measurements in spatial autocorrelation models, e.g. variograms, to allow a simple, localized, and energy efficient in-network coordination scheme among sensors so that the coordination overhead does not eclipse the communication savings.

We performed extensive experiments using two datasets. One is a real world sensor network deployment from the Intel Berkeley Research Lab [20]. This dataset is small with only 54 sensors. To further evaluate our framework, we evaluated various schemes using a large dataset consisting of thousands of points for 600 months. We compare the proposed model with 6 other simple models for approximate data collection from sensor networks. Our experimental results show that the proposed model provides significant savings.

The rest of the paper is organized as follows. We discuss related work in section 2. In section 3, we first formally define the problem and present an overview of our framework. Then we look into the details of using our framework with kriging as the spatial interpolation method along with a short discussion of other interpolation schemes. An extensive experimental study evaluating our proposed framework is presented in section 4. The paper is concluded with a discussion of possible future extensions of this work in section 5.

2 Related Work

We classify related work in the broad area of sensor network databases into four categories: in-network aggregation, correlation based sensor reporting, data compression, and interrogation.

TinyDB [22] is a sensor network database system with a traditional SQL like interface. Due to the resource and communication constrained nature of current sensors, query optimization schemes to reduce the energy consumption are the focus of much research effort [23, 6, 21, 26, 25]. In particular, in-network aggregation is considered an effective way to reduce the messaging cost for aggregation queries (e.g. sum and average) at the cost of simple in-network computations. The rationale is that communication is much more expensive than computation. In the TAG system [21], an aggregation tree is created when a query is broadcast to the sensors. The tree is used to aggregate the sensor readings from children to a parent all the way up to the root where the query originates. For distributive (e.g. sum, min, and max) and algebraic (e.g. average) aggregations, the TAG method significantly reduces the messaging cost by reducing message

hops. Recent work [25] pointed out that aggregation without considering the area that a sensor is representing may not be adequate for spatial aggregations such as average. With aggregation queries, detailed locational information is lost. Our work focuses on representing a field as faithful as possible while reducing communication cost.

Correlation based sensor reporting techniques utilize spatial and/or temporal correlation. Traditional temporal suppression schemes from stream processing that utilize approximate caching [24], time-series models [19], and Kalman Filters [14] have been adapted for mote size sensors. Approximate caching [24] relies on cached values to reduce the number of reports needed from the data stream to the sink. More sophisticated time-series models to capture the temporal trends of the sensor readings have been used in [19, 12]. Each sensor node calculates a function based on past readings to predict the readings of the node in the near future and sends it to the sink. In the case of high temporal autocorrelation, a time series is condensed into a single function, thus reducing communication cost. Our work focuses more on utilizing spatial autocorrelations and is orthogonal to models that use temporal autocorrelation. The approach to incorporate temporal models to the framework proposed in this paper will be discussed in the extended version of this paper.

Utilizing spatial autocorrelation has been suggested in prior research work. The clustering based approaches [12, 2, 27, 13] group sensor nodes according to the spectrum of sensing values or spatial proximity, and then select leaders to represent the group. Election and voting algorithms are important in selecting the representatives for a group of value correlated sensors [12, 31]. These algorithms must be distributed and localized in order to scale well for large sensor networks. Energy needs to be budgeted among representative election and communication of selected sensors. Spatial suppression was suggested through clustering sensors into groups and letting a group leader represent the whole group. The challenging problem of dynamic grouping and leader rotation were left for future research in [12].

Snapshot [16] investigated various heuristics for for electing a small set of representative nodes in the network in a localized manner to form a snapshot of the network and provide quick approximate answers to user queries. Unlike the scheme we propose in this paper, the representative selection process in Snapshot can only be performed very infrequently to achieve overall saving. In Ken [5], sensors are partitioned into disjoint cliques with one sensor in each clique selected as the leader of the clique. The leader assumes the duty of selecting a subset of data to be sent back to the sink according to a dynamic probabilistic predication model. The dynamic probabilistic predication model is obtained by a set of training data and is maintained by both the sink and the sensor field. Thus the sink can calculate the expected readings for sensors that do not report using the same predication model as the sensor field. Compared to Ken, our model has a simple probabilistic voting process to select sensors to report in the first round and relies on spatial interpolation models to control errors in the second

round. Our scheme can be extended to incorporate temporal compression and be compared with Ken in future work.

Data compression may be performed spatially or temporally in sensor networks. The information theoretical approaches [8] aim to find an optimal rate to compress redundant information in individual sensor readings. The joint routing and source coding approach [17] attempts to compress redundant information along the routing paths to reduce the number of bits transmitted. For these techniques, the number of transmitted messages are condensed but not reduced. Data compression is orthogonal to and may be applied on top of correlation based data suppression.

Selectively interrogating sensors is another way to avoid requiring all sensors to report. Bash *et. al.* proposed an energy efficient uniform sampling scheme for sensor networks [3]. A random sampler from the central station probes the sensor network to select the set of samples. A sensor uses its Voronoi cell to decide a probability to accept or reject the probe. Uniform sampling is useful for application domains such as querying the average sensor battery life. For other application domains such as finite element analysis uniform sampling may not be suitable. Other approaches include the Binocular system [10]. This approach divides sensors into working and sleeping sets and only collects data from the working set. A system model is used to estimate the values for the sleeping set. To avoid error accumulation the system model is updated by having all sensors report at some specified interval. Deshpande et. al. [9] proposed a model based probabilistic approach in the central site to answer queries which samples a few sensors when necessary to improve the estimation and achieve a confidence level guarantee. In general the interrogation based approach is "pull based" (compared with a "push based approach" such as Ken [5] and our model). A "pull based approach" does not provide an error guarantee and is insensitive to outliers which are more important for application domains such as environmental monitoring.

3 E2K Framework

In this section we will provide a formal definition of the problem and present our proposed E2K (Error Bounded Energy Efficient Kriging) framework. In addition, we discuss how to select an appropriate spatial interpolation model which is very important in our framework.

Once we have the readings from all sensors within some error threshold, the map interpolation problem becomes routine. To obtain a map, simply grid the space using a given spatial resolution, then interpolate using the readings from sensors to determine a value for each cell. So the problem is reduced to the following "SELECT * FROM sensorType FREQUENCY f WITHIN ϵ FOR t", or more formally:

Problem Statement: Let S be a set of spatially distributed sensors that monitor some attribute A at a time instance $t \in T$, and for $s \in S$ let $Z_t(s)$ be the value of A for sensor s at time t. Let C be a central collection sink that processes the information received from S, and let $Z_t^*(s)$ be the value C estimates for A

for sensor s at time t. Devise an algorithm for C and the sensors in S, such that for all s, t the estimated value is within a user specified error threshold $\epsilon > 0$ from the actual value, i.e. $|Z_t(s) - Z_t^*(s)| < \epsilon$, with the objective of reducing the total messaging cost.

Reducing total messaging cost is chosen as the objective because sending and receiving messages dominate the energy consumption in current sensor networks [5, 9, 21]. When a message is sent from a sensor, all the neighbors of that sensor will receive the message even though in many cases only a subset of the neighbors are intended destinations. Because sending and receiving have similar energy cost, it is fair to use the total message count sent or relayed from all sensors as the message cost (the actual total message cost that includes receiving and sending messages will be some multiple of the total message cost that we use).

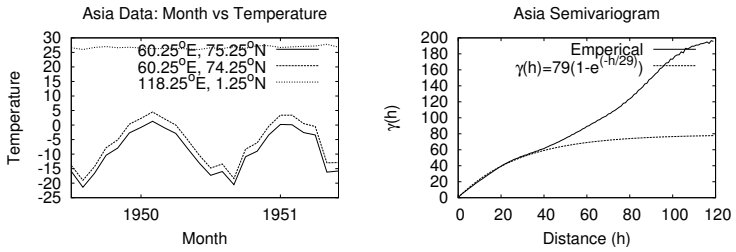

Fig. 1. Sample Spatial Autocorrelation and Empirical and Theoretical Variograms

Tobler's first law of geography [7] states that in space everything is related to everything else, but nearby things are more related than distant things. For example, the left chart in figure 1 shows temperatures of three locations from Asia for 2 years with a monthly sampling rate [1] (this dataset will be described further in section 4). The location (118.25°E, 1.25°N) is close to the equator and far away from the other two locations. The two close-by locations show very high correlation while the far-away location shows very little or no correlation with the other two.

Various models, e.g. variograms, Moran's I, and Geary's C [7], have been developed to quantify this phenomena (formally spatial autocorrelation). Spatial autocorrelation models have been incorporated into spatial interpolation models to create maps in many natural science domains. The spatial autocorrelation based interpolation models create "better maps" in the sense that the sum of the residual errors of the created map is closer to zero.

Our work is mainly inspired by spatial interpolation models utilizing spatial autocorrelation. The main thrust is a two round reporting framework featuring a probabilistic first round reporting and a spatial interpolation model based error control scheme in the second round.

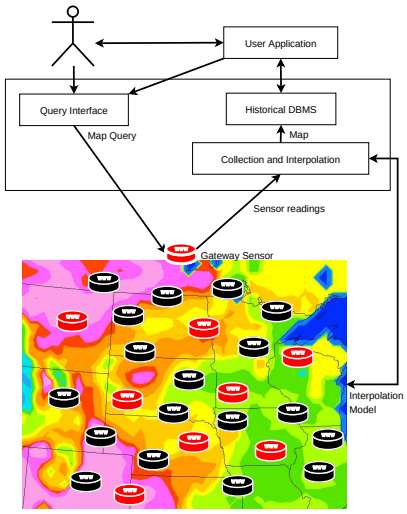

Fig. 2. System Overview

3.1 The Two Round Reporting Framework

The system level process is illustrated in Figure 2. Map queries are injected from a user interface into the sensor field. Each sensor makes a local decision about the importance of its reading and decides if it needs to report. As a result only a subset of the sensors (red or gray sensors in Figure 2) will actually report. The reporting sensors route their readings through any multi-hop protocol [30, 15], e.g. GPSR [15], to the gateway sensor or sink. The sink estimates the values for non-reporting sensors, and then interpolates a raster map from the sensor values using the same spatial interpolation model that was used in the field.

The key challenge here is the tradeoff between the complexity of coordination among sensors in dynamically deciding which ones should report, and the savings that result from having fewer sensors report while maintaining an error threshold from all sensors. We need to make sure that the coordination overhead does not eclipse the communication savings. Furthermore, it is desirable to have an error guarantee for each sensor. The three, often conflicting, goals are: (1) minimizing the number of sensors that report and thus save energy; (2) minimizing coordination costs among sensors; and (3) allowing the sink to interpolate readings for non-reporting sensors within an error threshold ϵ in the face of possible error propagation due to simple coordination schemes.

A naive probabilistic approach would let each sensor report with a probability p. This approach does not require coordination. However, there is no error bound for sensors that do not report. An alternative would be to have all sensors send their value to neighbors. Each sensor interpolates its own reading using the readings from its neighbors. If the interpolated value deviates from the real reading by more than ϵ, then the sensor reports. This approach again does not

have error guarantee for non-reporting sensors because of concurrent decision making and error propagation in sensor fields. We propose a two round reporting framework called E2K that is both energy efficient and has an error guarantee. E2K consists of algorithm 1 for individual sensors and algorithm 2 for the central site.

Algorithm 1 Sensor (s_0) Algorithm

1: $Z_t(s_0) \leftarrow$ value of A for this sensor at time t.
2: $rand \leftarrow$ a random number $\in [0, 1]$
3: **if** $(rand < p)$ **then**
4: {Round 1}
5: Report $(Z_t(s_0), round_1)$ to the central site and neighbors within distance r.
6: **else**
7: {Round 2}
8: $R \leftarrow$ the set of readings from sensors within distance r that reported in first round.
9: $Z_t^*(s_0) \leftarrow interp(R)$
10: **if** $(|Z_t^*(s_0) - Z_t(s_0)| \geq \epsilon)$ **then**
11: Report $(Z_t(s_0), round_2)$ to the central site.
12: **end if**
13: **end if**

The algorithm for sensors is divided into two rounds. In the first round a sensor decides to report with a probability p. As a result, a set of representative sensors are selected to temporarily represent the field. Reporting sensors route their readings to the sink and also send their readings to all sensors within distance r for use in the second round. The second round is needed because the first round does not have error bounds for non-reporting sensors. In the second round, a non-reporting sensor in the first round will **interpolate** its reading assuming it does not report, using only the readings received from reporting sensors in the first round. If the estimated value deviates from the real value by more than ϵ, then it reports.

Any kind of interpolation method can be used in the second round, e.g., a simple average. However, a better interpolation method results in fewer sensors that need to report in the second round. Basically, the probabilistic first round provides a reasonable number of sensors to report so that the estimation methods will work well in the second round, and the better the interpolation method is the better the overall result will be. We will revisit spatial interpolation methods in section 3.2.

The algorithm for the central site is meant to mirror what is done by the individual sensors. If a reading is sent by a sensor, then that reading will be used. For sensors where no reading was received, then the reading will be estimated using the same method, and same set of neighbors, that the sensor used to determine whether to report in the second round which gives us an error bound for each sensor location of ϵ. Note that we assume a reliable (or at least best

Algorithm 2 Central Site Algorithm

```
1: Let S be the set of all sensors.
2: R_1 ← the set of values received from sensors for attribute A at time t in round 1.
3: R_2 ← the set of values received from sensors for attribute A at time t in round 2.
4: for all s ∈ S do
5:    if (s reported a value) then
6:       Z_t^*(s) ← value of s in R_1 ∪ R_2
7:    else
8:       R_n ← the subset of R_1 within distance r of s.
9:       Z_t^*(s) ← interp(R_n)
10:   end if
11: end for
```

effort messaging) sensor network in this paper. Reliability is an important issue in sensor network and will be addressed in a more extended version of this paper in the future due to space constraint. More formally:

Lemma 1 (Error Bounding). *For any sensor s, let $Z_t(s)$ be the actual value and $Z_t^*(s)$ be the estimated value of s for attribute A at time t. The $|Z_t^*(s) - Z_t(s)| < \epsilon$ using the proposed algorithms for each sensor and the central site.*

Proof. There are two cases to consider:

1. If s reported its value, then $Z_t^*(s) = Z_t(s)$ which implies $|Z_t^*(s) - Z_t(s)| = 0 < \epsilon$ since from the problem statement $\epsilon > 0$.
2. If s did not report its value, then $Z_t^*(s) = interp(R)$ where R is the set of values that reported in the first round and that are within distance r from s. This is the same as the estimated value used in the second round for sensor s. If $|Z_t^*(s) - Z_t(s)| \geq \epsilon$, them s would have reported. Therefore $|Z_t^*(s) - Z_t(s)| < \epsilon$.

Once all the sense readings are recovered, the space is gridded based on a given spatial resolution. A raster map is created by interpolating locations/cells without any sensor readings from all the sensors $(Z_t^*(s), s \in S)$.

3.2 E2K: Choosing Interpolation Method

Although our framework is general with respect to the interpolation method used and always guarantees an error bound, which interpolation method is used has a great impact on the performance of the E2K framework. Furthermore, we need to make sure the sensor field can handle sophisticated interpolation models without incurring too much storage and computation cost.

Let $Z(e)$ be a random function at a location e of a sensor field. The value of a location e_0 is estimated by a linear estimator using the neighbors, e_1, e_2, ..., e_n, within distance r:

$$Z^*(e_0) = \sum_{i=1}^{n} \lambda_i Z(e_i) \tag{1}$$

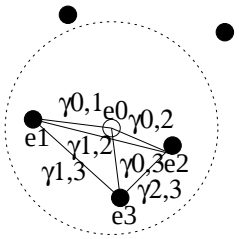

Fig. 3. Kriging Example

where $Z^*(e_0)$ represents the estimated value at location e_0, and λ_i refers to the weight given to the i^{th} neighbor's value. For example, in Figure 3, we wish to estimate a value at e_0, using the data values from the n neighboring sample points $e_i, i = 1, 2, 3$. So, we have $Z^*(e_0) = \lambda_1 Z(e_1) + \lambda_2 Z(e_2) + \lambda_3 Z(e_3)$.

There are a number of ways to assign the weights. It is desirable for the sum of the weights to be one so that if all neighbors have the same value, then this value will be the estimate. One way would be to assign equal weights to all neighbors, i.e., to take a simple average of the neighbors values (**Simple Average**). Another scheme would be to use the inverse of the distance as the weights, i.e., $\lambda_j = \frac{1/d_j}{\sum_{i=1}^{n} 1/d_i}$, where d_i is the distance from e_i to e_0 (**Inverse Distance**). However, these techniques do not examine the spatial structure of the data and may result in large estimation errors.

Kriging [7, 28] is widely used and has a long history of popularity in many natural science domains. It is a best fit linear unbiased estimator of a spatial variable at a particular site or geographic area. The goal of kriging is to create a raster map of a given resolution to represent a surface using a set of sample readings. It estimates a value at a location of a region for which a covariance/-variogram is known, or can be estimated using data in the neighborhood of the estimation location.

We assume $Z(e)$ is *second-order stationary*. This means the expected value $E[Z(e)] = m$, where m is the mean, for any point of the domain; and the covariance between any pair of locations depends only on the vector h that separates them, i.e., $C(h) = C(e, e + h) = E[Z(e) \times Z(e + h)] - m^2$. We chose ordinary kriging due to its adaptivity to local conditions, i.e., it only requires local second-order stationarity as opposed to global. In ordinary kriging, the estimation of a location e_0 can be expressed by equation 1.

Kriging assigns weights according to a known or estimated covariance/-variogram function which captures the spatial autocorrelation. The variogram $2\gamma_Z(h)$ is defined as $Var[Z(e+h) - Z(e)]$. Using sampled data the semivariogram is estimated as:

$$\hat{\gamma}_z(h) = \frac{1}{2N_h} \sum_{i=1}^{N_h} (z(e_i) - z(e_i + h))^2$$

where N_h is the number of pairs of samples whose distance from each other is h.

The empirical semivariogram is then fit with a theoretical model. For example, right figure of 1 shows the empirical semivariogram for an Asia temperature dataset [1] fit with an exponential model. (the two datasets will be described further in section 4).

There are two important parameters of the variogram model: the range and the sill. The range is the distance it takes for the variogram to reach the sill. The sill is an asymptotic bound the variogram reaches indicating that those values no longer have a meaningful correlation. We use range as a guidance in choosing the neighborhood parameter r for our algorithms in our E2K framework. The covariance can be expressed in terms of the variogram as $C(h) = C(0) - \gamma(h)$.

Once the variogram is known, the weights are chosen to minimize the error variance of the estimated values. The error variance can be calculated as:

$$\begin{aligned}
\sigma_E^2 &= E[(Z^*(e) - Z(e))^2] \\
&= E[(Z^*(e))^2] - 2E[Z^*(e) \times Z(e)] + E[(Z(e))^2] \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i \lambda_j E[Z(e_i) \times Z(e_j)] \\
&\quad - 2\sum_{i=1}^{n} \lambda_i E[Z(e_i) \times Z(e)] + E[(Z(e))^2] \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i \lambda_j (C(e_i, e_j) + m^2) \\
&\quad - 2\sum_{i=1}^{n} \lambda_i (C(e_i, e) + m^2) + C(0)
\end{aligned}$$

Since the sum of the weights should be one, a Lagrange parameter μ is added to get $L = \sigma_E^2 + 2\mu\{1 - \sum_{i=1}^{n} \lambda_i\}$ along with the constraint $\sum_{i=1}^{n} \lambda_i = 1$. Optimal values for weights $\lambda_i, i = 1, 2, \ldots, n$, are then obtained by the standard method of taking first derivatives of L with respect to each weight λ_i and setting them to zero.

$$\begin{aligned}
\frac{\partial(L)}{\partial(\lambda_i)} &= 2\sum_{j=1}^{n} \lambda_j (C(e_i, e_j) + m^2) - 2(C(e_i, e) + m^2) - 2\mu \\
&= 2\sum_{j=1}^{n} \lambda_j C(e_i, e_j) - 2C(e_i, e) - 2\mu \\
&= 2\sum_{j=1}^{n} \lambda_j (C(0) - \gamma(e_i, e_j)) - 2(C(0) - \gamma(e_i, e)) - 2\mu \\
&= -2\sum_{j=1}^{n} \lambda_j \gamma(e_i, e_j) + 2\gamma(e_i, e) - 2\mu = 0 \\
\\
\implies &\sum_{j=1}^{n} \lambda_j \gamma(e_i, e_j) + \mu = \gamma(e_i, e), \text{ for } i = 1, 2, \ldots, n
\end{aligned}$$

If the variogram function $\gamma(e_i, e_j)$ is given or can be estimated, this system along with the constraint on the weights gives us $n + 1$ equations, n unknown weights $\lambda_i, i = 1, \ldots, n$, and the unknown Lagrange parameter μ that we wish to obtain through solving the linear system. In Figure 3, to obtain the weights for estimating the value of e_0, we have the following system:

$$\begin{pmatrix} \gamma(e_1,e_1) & \gamma(e_1,e_2) & \gamma(e_1,e_3) & 1 \\ \gamma(e_2,e_1) & \gamma(e_2,e_2) & \gamma(e_2,e_3) & 1 \\ \gamma(e_3,e_1) & \gamma(e_3,e_2) & \gamma(e_3,e_3) & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \mu \end{pmatrix} = \begin{pmatrix} \gamma(e_1,e_0) \\ \gamma(e_2,e_0) \\ \gamma(e_3,e_0) \\ 1 \end{pmatrix}$$

Because kriging is preceded by an analysis of the spatial structure of the data by using a variogram model to integrate the average variability into the estimation model, the interpolation is likely to be more accurate than simple models. For the ordinary kriging that we described, the interpolation is exact, meaning if a sample value is available at the location of estimation, the kriging solution is equal to that value. Furthermore, kriging as a statistical method provides an indication of the estimation error.

The two main issues in the E2K framework using kriging are training and processing. Unlike the simple average and inverse distance schemes, kriging uses a variogram that is determined using previous data. Fortunately, all the training could be performed at the sink where resources are not constrained. After the training, only a few values used to describe the theoretical variogram need to be disseminated to the sensor network. Specifically, we use a training surface to obtain an empirical variogram at the sink. Then a theoretical variogram is fit to the empirical variogram with the goal of fitting the data with distance less than r as close as possible, again at the sink. The theoretical variogram is described by three parameters, namely a type (exponential, spherical, etc), range, and sill is provided to each sensor. As such, a fairly small amount of information needs to be sent to the sensors to allow sensor field and the sink to have the same interpolation model.

In terms of processing the main requirement is that the sensor is capable of solving the linear system once it determines which neighbors report in the first round. Due to the computation constraint of sensors, a large linear system is not desirable. Fortunately for kriging not many neighbors are needed to interpolate a value. E2K achieves a desired number of neighbors for each sensor by setting the value of the probability to report in the first round in our sensor side algorithm to $\frac{n_{desired}}{n_{current}}$, where $n_{desired}$ is the number of neighboring sensors desired and $n_{current}$ is the current number of neighboring sensors within the distance r. With a sensor deployment following a Poisson distribution, we expect $\frac{n_{desired}}{n_{current}}$ percent of all the sensors will report in the first round. By setting a small value for the $n_{desired}$ parameter, e.g. 5, it also results in the beneficial effect of dynamically adapting to the density of the sensor network and provides more savings for dense sensor networks with a similar level of spatial autocorrelation.

The only messages that need to be sent for coordination are the messages sent by the sensors that decide to report in the first round to their neighbors.

However, since the sensors which decide to report in the first round need to send their readings to the sink, the other sensors in the neighborhood can ear-drop the reading. So when the interpolation neighborhood is less than the radio range of the sensors, the coordination cost in terms of messaging is 0. Formally, we have:

Lemma 2 (Conditional Zero Coordination Cost). *In E2K, the number of messages sent in order to coordinate sensors in deciding which ones need to report and maintain an error bound is 0 when the spatial interpolation neighborhood is less than or equal to the radio range.*

4 Evaluation

Performance of E2K was evaluated using two datasets: (1) *Lab:* an Intel lab dataset [20] consisting of 54 sensors running for a little over a month monitoring temperature, voltage, humidity, and light. This data contains traces from a real sensor network deployment including network failure information, so it is particularly useful for examining failures. (2) *Asia Temperature:* the Asian portion of a dataset created using station records from the Global Historical Climatology Network (GHCN) and Legates and Willmott's [18] datasets for monthly precipitation and air temperature. We used version 1.02 of this dataset, released in July of 2001, that contains data for each month from January 1950 to December 1999 [1]. The Asia subset was sampled randomly to get two densities with 25% and 75% of the points respectively. In our experiments, these sampled points were treated as sensors that formed a sensor network. This data set provides a large number of sensors in an outdoor environment where spatial correlations are stronger.

We compare our model with 6 other models on the two datasets:

- A base line model **TinyDB (T) [22]:** Every sensor will report in every epoch. The error is always 0, thus it is bounded in this scheme.
- Three models where each sensor only reports if its readings deviates from a reading agreed upon by both the sensor field and the sink by more than ϵ: (1) **Global Average (G)**: Both the sink and sensors keep a global average reading of all sensors obtained from the training data. If a sensor deviates from the global average by more than ϵ, then it reports; (2) **Approximate Caching (A) [24]:** Every sensor caches the last reported reading to the sink so the sink knows the same value as the sensor in case the sensor does not report. A sensor reports if the real reading deviates from the cached values by more than ϵ; (3) **Periodical Approximate Caching (P):** Same as approximate caching except that every sensor caches the last reported readings for a set of episodes and uses the previous episode to determine its value, e.g. for a 24 hour cycle the value is checked against the reading from the same hour in the previous day. The lab data uses a 24 hour period and the Asian temperature data uses a 12 month period.

Table 1. Abbreviations of Comparison Schemes

T :TinyDB	**G** :Global Avg.	**A** :Appr. Caching	**P** :Periodical Appr. Caching
S :Simple Avg.	**I** :Inverse Distance	**O** :Ordinary Kriging	

- **Ordinary Kriging (O):** E2K with *ordinary kriging* as the interpolation method. For the Asian temperature data, we removed the trend of the data for each sensor by subtracting periodic means obtained from the training data for each location. Each sensor will subtract that mean and use E2K for reporting its residual. The sink knows the same mean and will add it back upon receiving or interpolating the residual.
- Two schemes using the E2K framework but with simple interpolation methods without considering spatial structure: (1) **Simple Average (S):** E2K with *simple average* as the interpolation method; (2) **Inverse Distance (I):** E2K with *inverse distance* as the interpolation method.

The abbreviations for the schemes are summarized in Table 1 for the convenience of the readers. For all schemes, we implemented a program using Java to simulate the reporting behavior of each sensor node using the readings of each sensor and its neighboring sensors for a given time from Lab and Asia datasets (we are currently implementing the algorithms on Berkeley Motes for a project to monitor soil moisture at Ray Roberts Greenbelt of North Texas). Unless otherwise specified, the radio range is 30 meters and the error threshold is 0.5°C for the lab data, and the radio range is 5 degrees with an error threshold 1°C for the Asian temperature data.

For the lab dataset the total messaging cost was estimated using a fixed multiple of the number of reporting sensors, i.e, we assumed each sensor had to send a message to the central site using some fixed number of hops. This was done because the area and number of sensors are too small for there to be a large number of hops. For the Asia data set the central site was chosen to be the center of the map and the number of hops to send a message was estimated taking into account the distance of each reporting sensor to the sink. For the lab data the first 94 hours was used for training and the next 452 hours was used for testing. For the Asian temperature data the first 10 years was used for training and the next 40 years was used for testing. We performed extensive experiments using various values for the parameters and, due to the limitation of space, we present a representative set.

4.1 Performance on Asia Temperature Dataset

Messaging Savings in Percentage that Report In this section, we examine the percentage of sensors that actually reported for all the schemes. Note the communication cost needed to route the data to the sink are not factored in yet.

Figure 4 shows the performance of the schemes with respect to the density of the sensors and the error threshold ϵ. The figures in Figure 4 show that E2K outperforms all the other schemes in terms of percentage of sensors that need

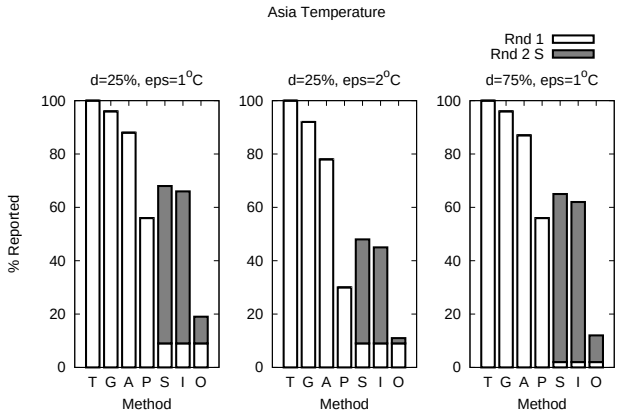

Fig. 4. Percentage Report for Asia Temperature Data

to report. Compared to the baseline method **T**, two round reporting helps E2K schemes, i.e., **O, S** and **I**, avoid unnecessary reporting from sensors that can be interpolated from the sink. E2K using ordinary kriging reduces the percentage of sensors that need to report to 19% with $\epsilon = 1$ and 12% with $\epsilon = 2$ for density of 25%. For density of 75%, the percentage of sensors that need to report are 13% with $\epsilon = 1$.

With increased density from 25% to 75%, the spatial related schemes including ordinary kriging (**O**), simple average (**S**), and inverse distance (**I**) show various levels of increased savings, ranging from 3% to 6%. There is no noticeable increase in savings for **G** when the density is increased. As expected, performance of temporal methods including approximate caching (**A**) and periodical approximate caching (**P**) do not change according to sensor density. In general approximate caching and global average do not do well resulting in more than 80% reporting sensors.

Total Messaging Savings In this section, we report the performance of the schemes with routing cost taken into consideration. Figure 5 shows the percent of total messaging cost for each method. It is important to note that the trend is the same as the percentage to report discussed previously. By keeping neighbors within radio range to eliminate coordination cost the most important factor is the number of sensors to report.

4.2 Performance on Lab Dataset

For lab data, schemes using E2K framework, i.e. **O, S,** and **I**, outperforms **T,G**, and **P** by more than 20%. With ϵ increasing, the savings increase. Approximate caching performs the best due to the strong temporal correlation in this dataset.

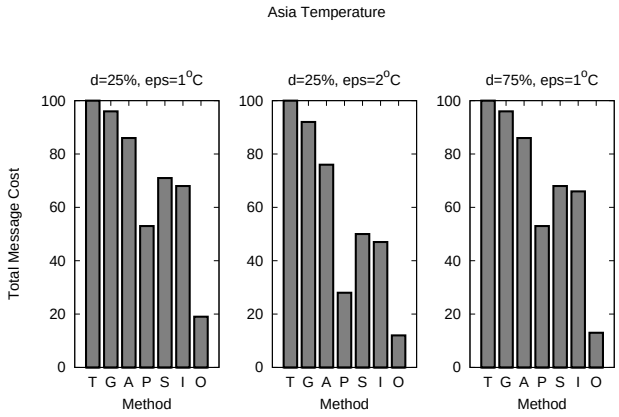

Fig. 5. Total Messaging Cost for Asia Temperature Data

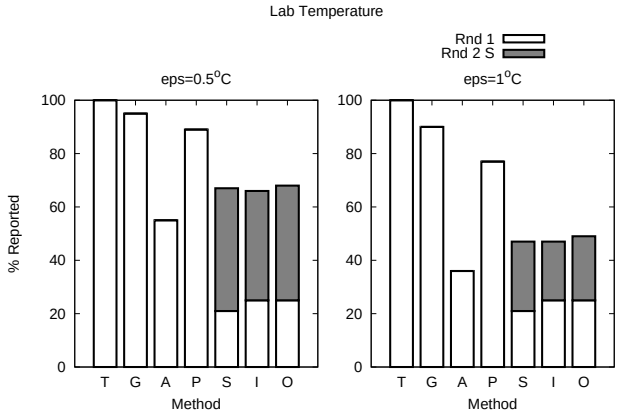

Fig. 6. Percentage Report for Lab Data

Incorporating temporal compression schemes into our E2K framework to allow adaptive utilization of spatial or temporal autocorrelation based on whichever is stronger will be an interesting topic for future work. An interesting observation is that the schemes using E2K framework achieve similar performance to each other with no method being clearly superior. The reason is that the lab data is not a typical outdoor environmental monitoring setup and the spatial autocorrelation is not be captured well by variograms. When using a nugget theoretical function (straight line parallel to x-axis) as shown in Figure 7 for the variogram, kriging interpolation degenerates into a simple average model. A promising topic for future research is a query optimizer for the sink that uses variograms to choose the best interpolation model. In case a simple average or reverse distance model is chosen, we will have the advantage of a simplified decision making process at sensor nodes.

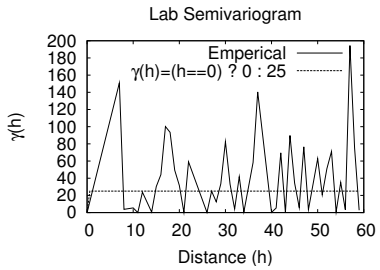

Fig. 7. Lab Data Empirical and Theoretical Variograms

5 Extensions and Conclusion

In this work, we proposed the E2K framework which can use any general spatial interpolation method. When the spatial interpolation method works well, i.e., the residual errors are small and it requires only localized information for spatial interpolation, our framework is likely to save more on messaging cost. Possible future extensions of our framework include incorporating temporal compression to and considering the use of regression or co-kriging to utilize auxiliary variables.

References

1. University of delaware surface air temperature data. http://climate.geog.udel.edu/~climate.
2. M. H. Ali, W. G. Aref, and C. Nita-Rotaru. Spass: Scalable and energy-efficient data acquisition in sensor databases. In *MobiDE*, 2005.
3. Boulat A. Bash, John W. Byers, and Jeffrey Considine. Approximately uniform random sampling in sensor networks. In *DMSN*, 2004.
4. Philippe Bonnet, J. E. Gehrke, and Praveen Seshadri. Towards Sensor Database Systems. In *Proc. of Second International Conference on Mobile Data Management*, 2001.
5. David Chu, Amol Deshpande, Joseph Hellerstein, and Wei Hong. Approximate data collection in sensor networks using probabilistic models. In *ICDE*, 2006.
6. Jeffrey Considine, Feifei Li, George Kollios, and John Byers. Approximate aggregation techniques for sensor databases. In *ICDE*, 2004.
7. N.A.C. Cressie. *Statistics for Spatial Data*. Wiley and Sons, ISBN:0471843369, 1991.
8. Antonios Deligiannakis, Yannis Kotidis, and Nick Roussopoulos. Compressing historical information in sensor networks. In *ACM SIGMOD*, pages 527–538, 2004.
9. Amol Deshpande, Carlos Guestrin, Samuel R. Madden, Joseph M. Hellerstein, and W. Hong. Model-driven data acquisition in sensor networks. In *Proc. of VLDB*, pages 588–599, 2004.
10. F. Emekci, S.E. Tuna, D.Agrawal, and E.Abbadi. Binocular: A system monitoring framework. In *International Workshop on Data Management for Sensor Networks*, August 2004.

11. Q. Fang, F. Zhao, and L. Guibas. Counting targets: Building and managing aggregates in wireless sensor networks. *Tech. Report, Palo Alto Research Center*, 2002.
12. Samir Goel, Andrea Passarella, and Tomasz Imielinski. Using buddies to live longer in a boring world, 2004. Rutgers Depart. of Computer Science Tech. Report DCS-TR-558.
13. Brian Harrington and Yan Huang. In-network surface simplification for sensor fields. In *ACM-GIS*, 2005.
14. Ankur Jain, Edward Y. Chang, and Yuan-Fang Wang. Adaptive stream resource management using kalman filters. In *SIGMOD*, 2004.
15. Brad Karp and H. T. Kung. Gpsr: greedy perimeter stateless routing for wireless networks. In *MobiCom*, 2000.
16. Yannis Kotidis. Snapshot queries: Towards data-centric sensor networks. In *ICDE*, pages 131–142, 2005.
17. Bhaskar Krishnamachari, Deborah Estrin, and Stephen B. Wicker. The impact of data aggregation in wireless sensor networks. In *Proceedings of the 22nd International Conference on Distributed Computing Systems*, pages 575–578, 2002.
18. D. R. Legates and C. J. Willmott. Mean seasonal and spatial variability in global surface air temperature. *Theor. Appl. Climatol.*, pages 11–21, 1990.
19. Ming Li, Deepak Ganesan, and Prashant Shenoy. Presto: Feedback-driven data management in sensor networks. In *Proceedings of the Third ACM/USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, May 2006.
20. Samuel Madden. Intel lab data. http://berkeley.intel-research.net/labdata/.
21. Samuel R. Madden, Michael J. Franklin, Joseph M. Hellerstein, and Wei Hong. Tag: a tiny aggregation service for ad-hoc sensor networks, 2002. OSDI.
22. Samuel R. Madden, Michael J. Franklin, Joseph M. Hellerstein, and Wei Hong. Design of an acquisitional query processor for sensor networks. In *SIGMOD*, 2003.
23. Samuel R. Madden, Robert Szewczyk, Michael J. Franklin, and David Culler. Supporting aggregate queries over ad-hoc wireless sensor networks. In *Workshop on Mobile Computing and Systems Applications*, 2002.
24. Chris Olston, Boon Thau Loo, and Jennifer Widom. Adaptive precision setting for cached approximate values. In *SIGMOD Conference*, 2001.
25. Mehdi Sharifzadeh and Cyrus Shahabi. Supporting spatial aggregation in sensor network databases. In *GIS '04: Proceedings of the 12th annual ACM international workshop on Geographic information systems*, 2004.
26. Niki Trigoni, Yong Yao, Alan Demers, Johannes Gehrke, and Rajmohan Rajaraman. WaveScheduling: Energy-Efficient Data Dissemination for Sensor Networks. *Internet Draft*, 2004.
27. Mehmet C. Vuran, B. Akan, and Ian F. Akyildiz. Spatio-temporal correlation: theory and applications for wireless sensor networks. *Comput. Networks*, 45(3), 2004.
28. Hans Wackernagel. *Mulitvariate Geostatistics*. Springer, 1995.
29. Y. Yao and J. Gehrke. The cougar approach to in-network query processing in sensor networks. In *Proceedings of SIGMOD*, 2002.
30. Y. Yu, R. Govindan, and D. Estrin. Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks, 2001. UCLA Computer Science Department Technical Report UCLA/CSD-TR-01-0023.
31. Feng Zhao and Leonidas Guibas. *Wireless Sensor Networks : An Information Processing Approach*. Morgan Kaufmann Series in Networking,ISBN: 1558609148, 2004.