A Filtering-based Clustering Algorithm for Improving Spatio-temporal Kriging Interpolation Accuracy

Qiao Kang, Wei-keng Liao, Ankit Agrawal, and Alok Choudhary
{qiao.kang, wkliao, ankitag, choudhar}@eecs.northwestern.edu
Electrical Engineering and Computer Science Department
Northwestern University, Evanston, Illinois, United States

ABSTRACT

Geostatistical interpolation is the process that uses existing data and statistical models as inputs to predict data in unobserved spatio-temporal contexts as output. Kriging is a well-known geostatistical interpolation method that minimizes mean square error of prediction. The result interpolated by Kriging is accurate when consistency of statistical properties in data is assumed. However, without this assumption, Kriging interpolation has poor accuracy. To address this problem, this paper presents a new filtering-based clustering algorithm that partitions data into clusters such that the interpolation error within each cluster is significantly reduced, which in turn improves the overall accuracy. Comparisons to traditional Kriging are made with two real-world datasets using two error criteria: normalized mean square error(NMSE) and χ^2 test statistics for normalized deviation measurement. Our method has reduced NMSE by more than 50% for both datasets over traditional Kriging. Moreover, χ^2 tests have also shown significant improvements of our approach over traditional Kriging.

Keywords

Kriging; Spatio-temporal interpolation; Spatio-temporal clustering;

1. INTRODUCTION

Geostatistical interpolation is the process that uses existing data and statistical models as inputs to predict data in unobserved spatio-temporal contexts as output. Kriging[7] is a well-known geostatistical interpolation method that minimizes mean square prediction error. Everyday weather forecasting, environmental hazard prediction and mineral mining are all its application domains. For example, Holdaway[5] has modelled variogram for predicting monthly U.S temperature. Noel[11] has modelled Piezometric-Head data in the Wolfccamp Aquifer for predicting heavy metal pollution level.

Kriging interpolation algorithm requires statistical assumptions about data. Firstly, Kriging interpolation assumes intrinsic stationarity in data. It means that semivariogram, a function of covariance between two points, depends on the displacement vector between them rather than their absolute coordinates. If data is not intrinsic stationary, Kriging interpolation will have poor accuracy.[14] Another commonly used assumption for Kriging is isotropy. Isotropic data implies that covariance functions with respect to distance between data points in all directions are the same, which in turn simplifies computation of Kriging interpolation. Unfortunately, real-world datasets usually do not obey these two assumptions, so Kriging interpolation accuracy is poor if these two properties are blindly assumed. A strategy that improves Kriging interpolation accuracy is to apply non-parametric analysis of data using randomized subsampling[3] when a dataset does not hold these two properties. However, this approach requires human interaction. Moreover, because non-parametric analyses with subsampling use only a subset of data, there are still concerns for final Kriging interpolation accuracy.

In this paper, we present a filtering-based data clustering algorithm that is designed to improve Kriging interpolation accuracy when data does not obey essential statistical properties. Input data points are divided into clusters based on the similarity function of Kriging interpolation error with respect to semivariogram models specified by users. Because only data within the same cluster are used to interpolate results, Kriging interpolation accuracy does not suffer from invalid statistical assumptions for the overall dataset. Our clustering algorithm will find the number of clusters without user interaction. The algorithm has two phases: filtering and reinforcement. Clusters are formed by minimizing clustering-based Kriging interpolation accuracy, while maximizing number of points in individual cluster. Maximizing individual cluster size not only prevents the algorithm from producing trivial results, but also ensures that clustering-based Kriging uses as much data as possible for interpolation.

We use two real world datasets to evaluate our algorithm. Comparisons to traditional Kriging are made using two error criteria: normalized mean square error(NMSE) and χ^2 test statistics for normalized deviation measurement. Our clustering-based Kriging has reduced NMSE by more than 50% for both datasets compared to NMSE produced Kriging without clustering. Moreover, only our clustering-based

CIKM'16 , October 24-28, 2016, Indianapolis, IN, USA

DOI: http://dx.doi.org/10.1145/2983323.2983668

Kriging has χ^2 test statistics less than 1% significance level while Kriging without clustering has the χ^2 test statistics at least greater than 10% significance level.

The rest of this paper is arranged as the follows. Firstly, we present the ordinary Kriging interpolation algorithm in section 2. Secondly, we illustrate the design of our proposed filter-based clustering algorithm in section 3. Thirdly, we discuss the implementation of proposed algorithm using both caching and heuristics for improving computational performance in section 4. Finally, we demonstrate evaluations of the proposed algorithm with two real world datasets using two error measurements in section 5.

2. KRIGING ALGORITHM

There are three variants of Kriging: Simple Kriging, ordinary Kriging and universal Kriging.[12] The type of Kriging used in this paper is ordinary Kriging. In this section, we describe ordinary Kriging presented in [14] based on variogram function.

Given n input data points consist of k dimensional coordinates $S = \{s_1, ..., s_n\}$, their associated physical attributes $\{z_1, ..., z_n\}$, a variogram model γ for covariance between data points, Kriging algorithm is a mapping $Z : \mathbb{R}^k \to \mathbb{R}$ that predicts the physical attribute $Z(s_0)$ by $\sum_{i=1}^{n} w_i Z(s_i)$ at a new location s_0, whereas $Z(s_i) = z_i \forall s_i \in S$ by input definition. Weights $W = \{w_1, ..., w_n\}$ are assigned such that the mean square error (MSE) of $\hat{Z}(s_0)$ is minimized subject to constraint $\sum_{i=1}^{n} w_i = 1$. With intrinsic stationarity assumption, semivariogram function has the property $\gamma(h) = \frac{1}{2}E(Z(s+h) - Z(s)) \forall h \in \mathbb{R}^k$. As a result, we can formulate a dual optimization problem with the following Lagrange function.

$$\begin{aligned} L(S, W, \lambda) &= MSE(\hat{Z}(s_0)) - \lambda(\sum_{i=1}^{n} w_i - 1) \\ &= 2\sum_{i=1}^{n} w_i \gamma(s_0 - s_i) - \lambda(\sum_{i=1}^{n} w_i - 1) \\ &- \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j \gamma(s_i - s_j) \end{aligned}$$

Solving the Lagrange function with first order condition, we can have the following linear system.

$$\frac{dL}{dw_i} = 2w_i\gamma(s_0 - s_i) - 2\sum_{j=1}^{n} w_i \gamma(s_j - s_i) - \lambda$$

$$\frac{dL}{d\lambda} = 1 - \sum_{i=1}^{n} w_i$$

Setting the derivatives equal to zero, we can obtain a linear system of $n+1$ equations. Let Γ be an $(n+1) \times (n+1)$ dimension matrix, where $\Gamma[i, j] = \gamma(s_i - s_j)\ \forall i, j \in [n]$, $\Gamma[n+1, n+1] = 0$, and the rest of entries are 1. Let m be an $n \times 1$ dimensional column vector with $m[i] = \gamma(s_0 - s_i) \forall i \in [n]$ and $m[n+1] = 1$. Let $w = (w_1, ..., w_n, \frac{\lambda}{2})^T$. Sherman[14] formulates the linear system as $\Gamma w = m$, so assigning Kriging interpolation weights is reduced to solving the linear system and retrieving weights w. The variance of Kriging is $w^T m$.

2.1 Spatio-temporal Kriging

Although Kriging was originally formulated for the purpose of predicting physical attributes of spatial data, spatio-temporal data interpolation such as weather forecast can also apply Kriging. For example, in [5], a spatio-temporal variogram model is formulated to interpolate future temperatures in St. Paul metropolitan area using recent temperatures of all cities in Minnesota State by Kriging. The inputs are spatio-temporal coordinates of cities in Minnesota State $\{s_i = (x_i, y_i, t_i)\}$ for location (x_i, y_i) and time stamp t_i, next day's time stamp, and the spatial coordinates of St. Paul metropolitan area. Moreover, each s_i is associated with an attribute z_i, representing temperature at location (x_i, y_i) and time stamp t_i. The output is the estimated next day's temperature in St. Paul metropolitan area. An efficient implementation of real-time spatio-temporal Kriging for solving this type of problem is in [15].

2.2 Clustering-based Kriging

Abedini, Nasseri and Ansari exploit data preprocessing using K-means clustering algorithm for Kriging interpolation.[1] Firstly, the algorithm separates data points into clusters by using K-means clustering. Then, the prediction for physical attribute $\hat{Z}(s_i)$ of a data point s_i is interpolated via other members in the same cluster. Finally, the accuracy of the algorithm is evaluated by using normalized mean square error (NMSE), which is defined as the following equation.

$$\frac{1}{s^2 n}\sum_{i=1}^{n}(\hat{Z}(s_i) - Z(s_i))^2.$$

s^2 is the sample variance. NMSE not only measures the actual square error, but also takes the sample variance into account.

K-means clustering-based Kriging has two challenges to be addressed. Firstly, the algorithm requires the number of clusters to be determined in advance. For large spatio-temporal datasets, trying a large number of parameter k is expensive. Human effort is also required to compare different types of error measurements for choosing an optimal k. Secondly, the algorithm only considers distance between pair-wise data points without taking the covariance of physical attributes into account. Thus, the algorithm may not necessarily minimize the Kriging interpolation error.

3. DESIGN

We propose a top-down based clustering algorithm that improves Kriging interpolation accuracy within each output cluster. A cluster that does not violate an error constraint specified by users is called consistent cluster, which is formally defined in definition 3.1. The algorithm is an optimization problem that maximizes the size of individual clusters without violating consistent constraint for clusters. Maximizing size of clusters avoids trivial output, which means that one data point corresponds to one cluster. Moreover, it ensures that clustering-based Kriging uses as much consistent data as possible for interpolation, which in turn maintains precision for Kriging interpolation.

Kriging requires fitted variogram model to be specified in advance. In [2], it has been shown that weighted least square

Algorithm 1: Filtering-based clustering algorithm

Data: $s_1, ..., s_n, Z(s_1), ..., Z(s_n)$, a threshold, and a variogram model M
Result: A queue of clusters Q, each point s_i belongs to a unique cluster.

```
1  C ←{s_1, ..., s_n};
2  Q ←{C};
3  while (C ⟵ next_element(Q)) ≠ ∅ do
4      C′ ⟵new cluster;
5      converge ⟵ false;
6      while converge == False do
7          converge ⟵ true;
8          for s ∈ C do
9              converge ⟵
               converge ∧ Filter(s,C,C′,threshold,M);
10         end
11     end
12     converge ⟵ false;
13     while converge == False do
14         converge ⟵ true;
15         for s ∈ C do
16             converge ⟵ converge ∧
               Reinforce(s,C,C′,threshold,M);
17         end
18     end
19     Remove all elements in C′ from C;
20     if !is_empty(C′) then
21         Insert(Q, C′);
22     end
23 end
```

Algorithm 2: Filtering

Data: point s, cluster C, where $s \in C$, C', a threshold, and a variogram model M
Result: if the point is consistent to the cluster

```
Remove(s, C);
t ⟵ normalized_kriging_error(s,C,M);
if |t| > threshold then
    Add(s, C′);
    Return False;
else
    Add(s, C);
    Return True;
end
```

(WLS) is feasible for most common models such as exponential model and spherical model. The algorithm assumes that a predefined variogram model remains invariant.

Initially all data points are inserted into a single cluster. Starting from line 3, the algorithm iteratively separates data points into clusters with two phases: filtering phase and reinforcement phase. Points that are inconsistent to the rest of points in the same cluster are filtered out at filtering phase. A new cluster will be initialized to hold the filtered points. Because the algorithm checks consistency by using all remaining data point in the original cluster, it may remove points that are consistent to the original cluster at the end of filtering phase. Therefore, it has to reinforce the cluster by adding falsely removed points back at reinforcement phase. Figure 1 illustrates the two phases.

Algorithm 3: Reinforcement

Data: point s, cluster C, where $s \in C$, C', a threshold, and a variogram model M
Result: if the cluster is consistent

```
Add(s, C);
if consistent(C,threshold,M) then
    Remove(s, C′);
    Return False;
else
    Remove(s, C);
    Return True;
end
```

3.1 Filtering Phase

The first inner while loop from line 6 to line 11 of algorithm 1 iteratively filters out any point that have large discrepancy between its Kriging interpolated physical attribute based on other points in the same cluster and its actual physical attribute. Points are removed from the cluster in a Gauss-Seidel style, which iterates through cluster and removes points without taking previously removed points into account.

Algorithm 2 (filter function) returns a Boolean. It calls the function *normalized_kriging_error*. This function uses other elements in C to predict the physical attribute of s by the Kriging interpolation described in section 2. Then the difference between the predicted value and the actual value of $Z(s)$ is normalized by dividing Kriging variance. The result is referred as normalized Kriging error. Consequently, the threshold can be chosen from a significance value in standard normal distribution. Selecting threshold from a standard distribution has the advantage of controlling the degree of confidence that a user requires for clustering-based Kriging interpolation accuracy. If the threshold is exceeded, the algorithm will return false and the convergence variable is set to be false. In this case, another round of filtering for cluster C will be triggered because the cluster has not converged yet.

3.2 Reinforcement Phase

In the second inner while loop from line 13 to line 18 of algorithm 1, the algorithm iteratively attempts to add removed points back to the original cluster if the consistency constraint is not violated. Hence the size of the cluster is maximized subject to the Kriging consistency constraint. Kriging cluster consistency, defined in definition 3.1, states that the normalized Kriging error should be less than the threshold for every point in all clusters explicitly.

Algorithm 3 (reinforce function) returns a Boolean. It inserts a point s back to a cluster C and test the Kriging consistency of C. This process computes the normalized kriging error for all points in C. If any of the points has normalized Kriging error exceeding the threshold, the consistency test fails and the point is rejected to be inserted back to the cluster C for current round of reinforcement.

3.3 Convergence Analysis

Clustering is defined to be a partition of objects into

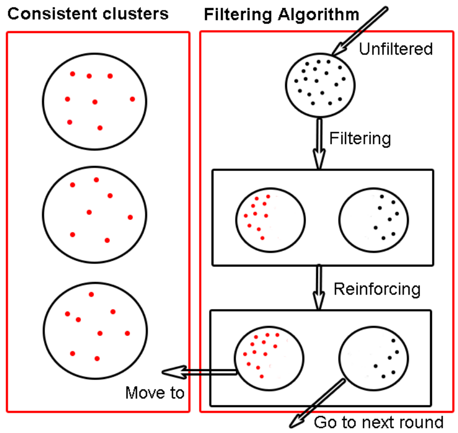


Figure 1: An illustration of filter algorithm. The left side contains clusters that are filtered and reinforced. The right side shows how a cluster is filtered and reinforced. The remaining points at the end of the process enter the next round.

groups such that objects within every group are as correlated as possible and objects between groups are as less correlated as possible. The term correlation defined for our clustering algorithm is based on Kriging interpolation accuracy: Any point within a cluster should be estimated by Kriging interpolation using other points in the same cluster with low normalized Kriging error.

Definition 3.1. *A cluster C is consistent if and only if the Kriging interpolation of any element in C using the rest of elements in C does not have a normalized Kriging error greater than the threshold.*

Definition 3.2. *A converged set of clusters $Q = \{C_1, ..., C_k\}$ satisfies: For any $2 \leq j \leq k$ and $C_i \in Q$ such that $j > i$, C_i is not consistent if any of points in C_j is inserted into C_i.*

Proposition 3.3. *The output of algorithm 1 is a converged set of clusters.*

Proof. It is possible to prove this statement by induction on the number of iterations for the loop at line 3.

Base case: At the end of first iteration, C has been filtered and reinforced. $\forall s \in C'$, because reinforcement phase is finished, it is impossible to put s back to C, which is the only element in Q.

Inductive assumption: Suppose that at the end of k^{th} step, $\forall s \in C'$, $\nexists A \in Q$ such that A is consistent after s is inserted into it.

Inductive step: Consider the $(k+1)^{th}$ step, C is the cluster created in the last iteration. Because we cannot filter more number of elements than the number of elements in C, C' at the end of this iteration strictly contained in the input C. By inductive assumption, $s \in C'$ is inconsistent with rest of the clusters in Q. However, because the way how the algorithm reinforces the cluster C, any element in C' must not be consistent with C in the end. Otherwise they would have been put back to C. As a result, the inductive assumption is true at $(k+1)^{th}$ step. □

Despite the fact that we have shown the final result is a stable solution for maximizing cluster size subject to threshold constraint, a question remains is that how the order of data affects rate of convergence. The strategy used in our implementation is to shuffle data in advance. However, in the future, it is interesting to find out how to align data so that filtering and reinforcement are faster.

3.4 Complexity Analysis

Kriging interpolation algorithm is the core of filtering and reinforcement phases. In section 2, we have shown that Kriging interpolation algorithm can be reduced to solving a linear system of equations. In our implementation, we use lower upper permutation (LUP) approach to solve the linear system. The solution is exact, but the complexity is bounded by $O(n^3)$. However, other solvers can speed up performance. For example, in [10] and [9], there are Kriging solvers that solve the linear system reduced from Kriging interpolation algorithm efficiently. In the following analysis, we treat this computation as Kernel function and focus on analysing the complexity of Algorithm 1 at high level, namely the number of calls to the kernel functions *normalized_kriging_error* and *consistent*.

A round of filtering and reinforcement at least reduces the size of data to be filtered by 1. Otherwise the outer loop at line 3 breaks, so the outer loop is executed at most by $n-1$ times. Consider filtering phase, the worst case is that we filter 1 point per round from C to C', which costs $O(n^2)$. For reinforcement phase, it is also possible that points are reinforced back to the C 1 point per round. which costs $O(n^2)$. Therefore, the overall complexity is $O(n^3)$.

4. IMPLEMENTATION

We designed two approaches for improving computational performance: Caching and Heuristics. These two approaches do not change the result of experiments. Hence there is no accuracy loss. The entire algorithm was implemented in C language.

4.1 Caching for Kriging Interpolation

When the filtering phase is converging, only a few points in the cluster are filtered out per round. Therefore, many points in the cluster still have the same neighbours as before. Thus, there is no need to compute normalized Kriging error again during the next round of filtering for those points if we can store the result of *normalized_kriging_error* for each data point. If no neighbours have been filtered out in the previous round by checking the number of neighbours, the algorithm can simply read out the result of *normalized_kriging_error* without computing it again.

Similarly for reinforcement, when a point is put back into C, which is the cluster it has been filtered out from, the algorithm can check consistency of the cluster using cached results for points that have the same neighbours as before. In addition, if reinforcement succeeds for a point, the cached result can be used for next round of reinforcement. Otherwise the caches of points that belongs to the neighbour of the reinforced point should be poisoned for correctness.

4.2 Heuristics for Reinforcement

During reinforcement, the algorithm checks if C is consistent after adding a point from C' to C. Checking consistency requires computing normalized Kriging error for each point in the cluster, which is equivalent to a Jacobi style filtering phase. A trick can be used to report inconsistency faster. We can check the consistency of points that are more likely to violate the threshold constraint first. If they violated the constraint, there is no need to check the rest of points.

Firstly, a point that violates the threshold constraint is likely to be the point that is reinforced because it has been filtered out in the filtering phase at current iteration. We can treat this point as a special case and revise its consistency to the cluster before all other points. Secondly, we can wisely use the caching information in the previous round. For those points that already have high errors, their errors are likely to exceed threshold than points with low errors when a new point is inserted to the cluster that they belong to. This estimation is a heuristic because the cached result may not be correct due to cache poisoning. However, it provides a raw approximation for normalized Kriging errors cheaply given the cluster size is large. We can sort the order of reinforcement in descending order based on cached normalized Kriging error and check consistency of points in this order. The extra cost is $O(nlogn)$ on average for quick sort. However, it may save many $O(n^3)$ operations for solving linear systems.

5. EXPERIMENTAL RESULTS

We used two datasets in different applications to evaluate our algorithm. The first SOCR dataset is a classical dataset for testing Kriging interpolation accuracy. We will compare our filtering-based Kriging to K-means clustering-based Kriging and Kriging without clustering in terms of accuracy. The second IGRA dataset is a spatio-temporal dataset with regular time interval. We will fit a variogram model step by step and justify that our clustering-based Kriging produces lower errors.

5.1 Quality Measurements

There are two measurements for evaluating the quality of spatio-temporal clusters in the following experiments. The first measurement is NMSE value, which is defined in section 2.2. It is a ratio between sum of squared leave-one-out cross-validation error and sample variance. The second measurement is χ^2 test statistics. We compute the normalized Kriging error $\frac{(\hat{s_i}-s_i)}{\sigma}$ using leave-one out cross-validation for all points in every cluster. The sum of squares of these errors is the χ^2 test statistics. For both measurements, the smaller the values, the better the quality of clusters.

5.2 SOCR Data

SOCR dataset [1] contains 85 data points, distributed near the south border of United States. Each data point has a 2 dimensional coordinate and a physical attribute that describes the water pollution level at the location. Because the dataset contains strong anisotropy[11], it is a challenging spatial dataset for Kriging. The dataset also contains nugget effect, which indicates that the covariance of two points that are very close to each other is a constant greater than 0.

Noel[11] fitted the parameters for NE-SW direction and NW-SE direction with geometrical anisotropy model derived by Journel and Huijbregts[6]as the following.

$$\gamma(h,\phi) = 14000 + 38h^2\cos^2(\frac{\pi}{4}-\phi) + 15h^2\cos^2(\frac{\pi}{4}+\phi)$$

The output of algorithm 1 successfully divided the data points into 5 clusters. The last two clusters contain only three points, so they are treated as noise. The experimental results are shown in Table 1. The t-test p-value is a paired sample t-test for absolute Kriging errors between each of Kriging interpolation methods and Kriging interpolation without clustering at every data point. We reproduced K-means clustering-based Kriging with k=6 in [1], which is the best result they claimed, using K-means implementation in [8]. Our filter algorithm reduced NMSE more than the K-means algorithm presented in [1]. Furthermore, although K-means clustering-based Kriging reduced NMSE, there is no significant improvement for both absolute errors and normalized errors by observing t-test p-value column and χ^2 column respectively. In addition, the 1% lower tile of χ^2 distribution with degree of freedom 85 is 57.63. As a result, only our filtering-based clustering algorithm has reduced errors that pass both χ^2 test and paired sample t-test with high confidence level.

Table 1: SOCR Dataset Results

Method	cluster size	NMSE	χ^2	t-test p-value
Unclustered	1	0.24	197.75	0.5
K-means	6	0.11	161.88	0.3252
Filtering	5	0.048	22.53	$9.541E^{-9}$

5.3 IGRA Data

IGRA dataset [2] contains sensor data collected by 67 NOAA stations across United States in real time. Every data point contains two dimensional spatial coordinates for longitude and latitude, a physical attribute for atmosphere temperature, and a time stamp. The update interval for atmosphere temperature between two time stamps is 12 hours for all stations. This dataset is challenging because of anisotropy. An illustration of anisotropy with maximum Kriging range equals 30 is shown in Figure 2a. The variogram in East direction appears to be different from other directions.

Similar to [5], we consider the monthly temperature of all stations. The semi-variance plot for average of all stations in January 2014 with 4620 data points is shown in Figure 2b.

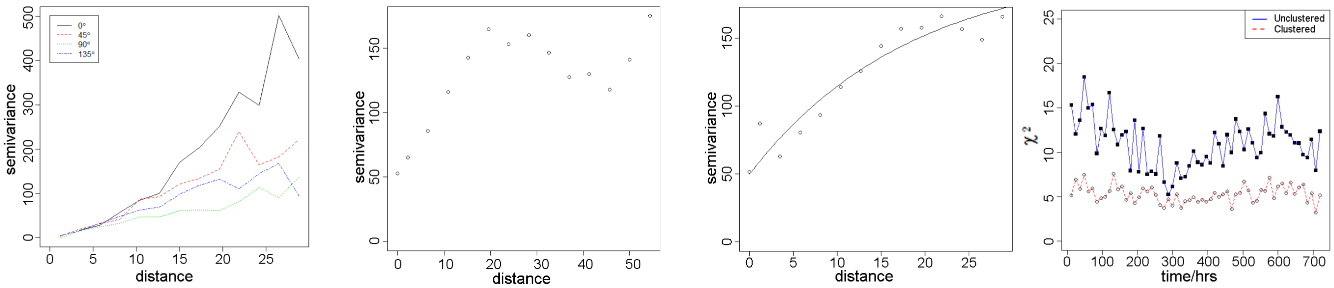
In Figure 2b, when $h > 30$, the variogram oscillates as h increases. This oscillation indicates that the atmosphere temperatures between two locations that are far from each other are weakly correlated. Similar to [4], which uses nearest neighbors for Kriging interpolation, we apply local Kriging based on spatial range. The limit for maximum distance is set to be 30. There is also a sign of nugget effect for semivariance, starting from approximately 50. Using geoR [13], we fitted an exponential variogram model illustrated in Figure 2c.

$$\gamma(h) = 54.84102 + 153.71639exp(-0.05462576\|h\|)$$

Setting threshold to be 0.6, which is the threshold that gives reasonable size and number of clusters by several trials, we apply our clustering algorithm to IGRA atmosphere data at 67 stations in January 2014. The result is compared with Kriging without clustering. At every time stamp, the number of clusters is at most 6, with the last one or two clusters containing noise filtered in the end. Therefore, the results are reasonable because the algorithm does not reduce errors by dividing stations into trivial partitions.

[1] http://wiki.stat.ucla.edu/socr/index.php/SOCR_061708_NC_Data_Aquifer

[2] http://www.ncdc.noaa.gov/data-access/weather-balloon-data



a b c d

Figure 2: (a) IGRA data variogram in four directions. The vertical axis shows the averaged covariance between points at a particular distance. Different lines show the variograms in different directions. (b) January IGRA data variogram. The vertical axis shows the averaged covariance between points at a particular distance. (c) Fitted model of the variogram curve bounded by distance 30. (d) Horizontal axis represents time in hours. Vertical axis shows the χ^2 IGRA Kriging error for Kriging with clustering and Kriging without clustering. The χ^2 statistics of clustering-based Kriging is lower than Kriging without clustering across all time stamps.

In Figure 2d, the χ^2 test statistics for both filtering-based clustering and Kriging without clustering are illustrated. For any given time stamp, Kriging with clustering has χ^2 test statistics much smaller than Kriging without clustering. Moreover, the NMSEs span from 0.08 to 3.8. However, the high NMSEs are resulted from noisy clusters filtered in the end. If we do not consider clusters with size less than 5, the NMSE is on average 0.27, with some outstanding time stamps at 1 or 2 for small clusters. Kriging without clustering has NMSE spanning from 6 to 18. These results indicate that Kriging interpolation accuracy based on clusters produced by our algorithm is better than Kriging interpolation accuracy without clustering.

6. SUMMARY AND FUTURE WORK

In this paper, we propose a filtering-based clustering algorithm that improves Kriging interpolation accuracy. Theoretical proofs have shown that the final clusters have consistency and convergence characteristics. We have proposed two implementation techniques, caching and heuristics, that help to improve the performance of proposed algorithm. Furthermore, experimental results have shown that the filtering-based clustering algorithm has significantly improved Kriging interpolation accuracy compared to traditional Kriging approach and K-means clustering-based Kriging.

The future work consists of at least two directions. The first direction is to derive theorems that can bound the rate of convergence of algorithm 1 given ground truth about data. The second direction is to speed up the performance of filter algorithm by using parallel programming.

7. ACKNOWLEDGEMENTS

This work is supported in part by the following grants: NSF awards CCF-1029166, IIS-1343639, CCF-1409601; DOE awards DE-SC0007456, DE-SC0014330; AFOSR award FA9550-12-1-0458; NIST award 70NANB14H012; DARPA award N66001-15-C-4036.

8. REFERENCES

[1] Abedini, M., Nasseri, M., and Ansari, A. Cluster-based ordinary kriging of piezometric head in west texas/new mexico – testing of hypothesis. *Journal of Hydrology 351* (2008), 360–367.

[2] Brunell, R. M. An automatic procedure for fitting variograms by cressie's approximate weighted least squares criterion. http://www.smu.edu/-/media/Site/Dedman/Departments/Statistics/TechReports/TR259.ashx?la=en, 1992.

[3] Guan, Y., Sherman, M., and Calvin, J. A. Non-parametric test for spatial isotrophy using subsampling. *Journal of the American Statistical Association 99*, 467 (2004), 810–821.

[4] Guccione, P., Appice, A., Ciampi, A., and Malerba, D. Trend cluster based kriging interpolation in sensor data networks. In *Proceedings of the 2011 international conference on Modeling and Mining Ubiquitous Social Media* (2011), pp. 118–137.

[5] Holdaway, M. R. Spatial modeling and interpolation of monthly temperature using kriging. *Climate Research 6* (1996), 215–225.

[6] Journel, A., and Huijbregts, C. *Mining statistics*. American Press, New York, 1978.

[7] Krige, D. A statistical approach to some basic mine valuation problems on the witwatersrand. *Journal of the Chemical* (1952), 119–139.

[8] Liao, W. Parallel k-means data clustering. http://users.eecs.northwestern.edu~wkliao/Kmeans/, 2005.

[9] Memarsadeghi, N., Raykar, V. C., Duraiswami, R., and Mount, D. M. Efficient kriging via fast matrix-vector products. In *Aerospace Conference IEEE* (2008).

[10] Mou, S., Liu, J., and Morse, A. S. A distributed algorithm for solving a linear algebraic equation. In *transactions on automatic control IEEE* (2015).

[11] Noel, C. *Statistics for Spatial Data*. John Wiley & Sons Inc., 2015.

[12] Noel, C., and Wikle, C. K. *Statistics for Spatio-temporal Data*. John Wiley & Sons Inc., 2010.

[13] Ribeiro Jr., P. J., and Diggle, P. J. Analysis of geostatistical data. https://cran.r-project.org/web/packages/geoR/geoR.pdf, 2015.

[14] Sherman, M. *Spatial Statistics and Spatio-Temporal Data*. John Wiley & Sons Inc., 2010.

[15] Srinivasan, B. V., Duraiswami, R., and Murtugudde, R. Efficient kriging for real-time spatio-temporal interpolation. In *20th Conference on Probability and Statistics in the Atmospheric Sciences* (2010), p. 228.