Optimal Sensor Placement for Field-Wide Estimation of Soil Moisture

Hosein Pourshamsaei and Amin Nobakhti

Electrical Engineering Department, Sharif University of Technology, Tehran, Iran.

A paper from the Proceedings of the
14th International Conference on Precision Agriculture
June 24 – June 27, 2018
Montreal, Quebec, Canada

Abstract. *Soil moisture is one of the most important parameters in precision agriculture. While techniques such as remote sensing seems appropriate for moisture monitoring over large areas, they generally do not offer sufficiently fine resolution for precision work, and there are time restrictions on when the data is available. Moreover, while it is possible to get high resolution-on demand data, but the costs are often prohibitive for most developing countries.*

Direct ground level measurement can be a viable and economical alternative if one is able to accurately estimate the value of soil moisture over the entire field by using measurement from only a few points. If the number of measurement points, their location, and data are available, then Compressive Sensing (CS) theory may be used to give an estimate of the moisture. This is because although moisture values in a field do not constitute a sparse signal, they are spatially correlated and can be expressed as sparse signals in other domains such as DCT or DFT.

The difficulty in using the CS theory for estimation of moisture values is that the number and location of the sensors must be known a priori. In reality, this means the optimization problem has to be solved several times for various different network configurations to determine the best layout. Straightforward augmentation of the CS reconstruction optimization problem to include the configuration selection leads to so called MINLP optimization type of problems which are combinatorial and non-polynomial time. Such problems take exceptionally large times to solve for large scale problems (as encountered in PA type of applications).

In this paper, we propose a new heuristic algorithm to find a sub-optimal set for sensor locations. A data set for numerical experiments is extracted from the simulation of a simple field using the state-of-the-art TIN-based Real-time Integrate Basin Simulator (tRIBS). This data set is used for validation of the optimization results.

Keywords. *tRIBS simulator, Optimal sensor placement, Compressive Sensing, Network configuration optimization, Soil moisture estimation.*

Introduction

Precise closed-loop control of irrigation, and analysis of moisture on a field is not possible without gathering sufficient and accurate information. Traditional moisture measurements method as found in small controlled operations such as green houses (Hamouda & Elhabil, 2017) are not scalable to large plantations due to cost and maintenance issues. Remote sensing methods are scalable to such large fields (Gruhier et al., 2010), however in-demand data at the correct resolution are expensive or not available.

In other engineering applications with similar challenges, a successful approach has been to combine direct measurements with estimation theorems such as compressive sensing (CS). CS theory is based on a l_0-norm optimization problem which is NP-hard problem and requires an exhaustive search of all possible locations of the nonzero entries in response and in large scale and real-time problems, it is nearly impossible to solve them directly. The l_1 minimization yields similar result as the l_0 minimization in many cases of practical interest (Patel & Chellappa, 2013). To form a standard basis of comparison, the l_1 minimization is used as approximation of the main problem in this paper.

CS theory is an effective tool to reconstruct sparse signals using samples at rates below the Nyquist rate. Although moisture data is not sparse, it has a sparse representation in the frequency domain due to the spatial correlation between values. CS theory can be used to estimate an entire set of data based on a subset of known values referred to as the measurement vector. However, it does not place specific constraints on the number of measurements or the location of the sensors. In the other words, sensor numbers and locations are variables which need to be determined a priori application of the CS theory.

Methods such as coarse-grained monotonic ordering (X. Wu, Wu, Liu, & Zheng, 2011) can be used to improve efficiency of random sensor placement, but certainly random placement is not the best choice in the case that variation of the values is not uniform. Obviously for the points that variation of values is high, more sensors should be used for better performance, but in random sensor placement, this is not considered.

The main contribution of this paper is to propose a heuristic iterative algorithm for sensor placement. Although the algorithm does not guarantee achieving the optimal sensor locations, but will be shown that in some situations (especially very low sensor allocation), the proposed placement will significantly outperform random placement.

The remainder of this paper is organized as follows. Firstly, CS theory is explained briefly and formulated and applied to the moisture estimation problem. Then, a section is dedicated to the discussion about data set that is used for the numerical experiments. Subsequently, the proposed algorithm for sensor placement is described and the results are compared with random placement. Finally, a brief conclusion is proposed and some points are mentioned for future works.

Compressive Sensing (CS) Theory

CS theory (Donoho, 2006) is a concept in information theory and signal processing that is useful for reconstructing sparse signals from measurements at rates below the Nyquist rate (Patel & Chellappa, 2013).

Let **x** be a discrete time signal which can be considered as an N×1 column vector in $\mathbb{R}^N$. **x** is K-sparse if it has only K nonzero elements. A signal is considered as sparse signal if K<<N.

The l_p – norm of a vector is defined as,

$$\|x\|_p = (\sum_i |x_i|^p)^{\frac{1}{p}}, \tag{1}$$

and the l_0-norm is defined as the limit $p\rightarrow 0$ of the l_p-norm,

$$\|x\|_0 = \lim_{p\to 0}\|x\|_p^p = \lim_{p\to 0}\sum_i |x_i|^p. \tag{2}$$

The l_0-norm of a signal counts the number of nonzero elements in the signal. So, if **x** is *K*-sparse, $\|x\|_0 = K$.

In practice many real signals are not exactly sparse, instead they are compressible. A signal is compressible if the magnitude of the coefficients (when sorted in a decreasing order) decays according to the power law (Patel & Chellappa, 2013). CS theory can be applied to compressible signals as well. In this paper, the phrase "sparse signal" is used to refer to both exactly sparse, as well as compressible signals.

Let **x** be a sparse signal of size *N*. It is possible to reconstruct it from *M* samples where $y_{M\times 1} = Ax$. $\mathbf{A}$ is usually referred to as the measurement matrix in CS theory. The problem of finding the sparsest solution can be formulated as the following optimization problem (Patel & Chellappa, 2013),

$$\hat{x} = \arg\min_x \|x\|_0 \text{ subject to } \; y = Ax. \tag{3}$$

This is a NP-hard problem and solving it for large scale problems is impractical. Accordingly, in practice we seek to solve a reasonable approximation of this problem, where a small degree of inaccuracy is traded for large reductions in computational loads. In many applications, the l_1 norm optimization can be used instead of l_0 norm. The l_1 norm optimization produces sparse solutions due to the discontinuity in its differential (Donoho, 2006). Although other algorithms such as weighted l_1 norm (Candes, Wakin, & Boyd, 2008), FOCUSS algorithm (Gorodnitsky & Rao, 1997) and greedy algorithms (Blumensath & Davies, 2009; Mallat & Zhang, 1993; Needell & Vershynin, 2010; Tropp & Gilbert, 2007) offer various approximations for sparse signal recovery, in this paper, the simple l_1 norm optimization is used to provide a base reference for comparison of allocations. Accordingly, we seek to solve the following optimization problem instead of **Eq. 3**:

$$\hat{x} = \arg\min_x \|x\|_1 \text{ subject to } \; y = \mathrm{A}x. \tag{4}$$

Applcation of CS Theory to Moisture Estimation Problem

As discussed previously CS theory is only applicable for estimation of sparse/compressible signals. Obviously, moisture data over a field is not a sparse signal. Thus CS theory cannot be directly applied to the moisture estimation problem. It is however possible to apply CS theory to a modified form of the problem as shall be demonstrated next.

Important factors that mostly effect moisture content are precipitation, topography, soil properties, soil depth and vegetation (Gwak & Kim, 2016). Most of these factors do not change rapidly and can be considered almost constant over reasonable periods of time. This stationary feature means that although the absolute value of the soil moisture changes in time, but relative moisture between two points is predictable and changes much more slowly. In other words, soil moisture data is spatially correlated (X. Wu et al., 2011). Therefore, although moisture data is not a sparse signal itself, but it can be transformed into the frequency domain using linear transformation such as DCT (Discrete Cosine Transformation) or DFT (Discrete Fourier Transformation) and in that domain, it will represent a sparse signal.

Let **x** be the moisture data vector at *N* locations that should be estimated using only *M* measurements. Typically, *M* measurements are randomly selected from the *N* points. Suppose that $\boldsymbol{\Phi}$ is the measurement matrix with *M* rows and *N* columns such that entries of each row contains *N*-1 zeros and 1 one. Thus, the measurement vector **y** is defined by,

$$y = \Phi x. \tag{5}$$

Since **x** is not sparse, the DCT transformation is used to transform **x** to the frequency domain. Let $\boldsymbol{\Psi}$ be the IDCT (Inverse Discrete Cosine Transformation) matrix. Accordingly, the new vector $\boldsymbol{\alpha}$ can be defined such that,

$$x = \Psi\alpha. \quad (6)$$

Indeed, **α** is the transformed moisture data in the Fourier domain and therefore, it is a sparse signal. Thus **Eq. 4** can be reformed as,

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_1 \text{ subject to } \; y = \Phi\Psi\alpha, \quad (7)$$

where,

$$\hat{x} = \Psi\hat{\alpha}. \quad (8)$$

As mentioned previously, M and ***Φ*** must satisfy some constraints in CS theory. In approximation of the problem with l_1-norm, *M* should be,

$$M \geq CK\mu^2(\Phi, \psi) \log N, \quad (9)$$

where *C* is a small constant, *K* is the number of nonzero elements of **α** and $\mu(\Phi, \psi)$ is defined as,

$$\mu(\Phi, \Psi) = \sqrt{N} \max_{1 \leq i,j \leq N} |< \Phi_i, \Psi_j >|. \quad (10)$$

The number μ measures how much two vectors in $\mathbf{A} = \mathbf{\Phi\Psi}$ are correlated. The value of μ is between 1 and $\sqrt{N}$. The matrix **A** is called incoherent when μ is very small (Patel & Chellappa, 2013). The selection matrix ***Φ*** is actually the spike basis in the space domain. The incoherence condition holds for many pairs of bases and for the spike and Fourier bases, as well (Candes & Wakin, 2008), that means,

$$\mu(\Phi, \Psi) = 1. \quad (11)$$

Thus, the minimum number of sensors that is required for fine approximation depends on sparsity of the signal. This completes formulation of the moisture estimation problem in CS theory.

Data Sets for Numerical Experiments

Data that is used in this paper is generated by state-of-the-art *TIN-based Real-time Integrate Basin Simulator (tRIBS).* This simulator performs distributed hydrogeomorphic simulations over complex basins using Triangulated Irregular Networks (TIN) to form the basis for multiple-resolution representations (Vivoni, Teles, Ivanov, Bras, & Entekhabi, 2005).

Data that is used in this paper is affiliated with Peacheater Creek Watershed. Peacheater Creek watershed covers an area of 64 km^2 and is located in the northeastern corner of Oklahoma. A simple map of this location is shown in **Fig 1** (S. Wu, Li, & Huang, 2007). Data is related to simulation of Peacheater Creek Watershed based the conditions of summer 1991 and contains soil moisture values at the depth 100 mm at 6095 points of the field at 80 hours after start of the simulation.

In compressive sensing it is always desirable to have signals with higher degrees of sparsity. A simple way to increase sparsity in the frequency domain is to increase the correlation of data in the time domain by sorting them in an ascending order. The assumption of exactly sorting the moisture values is not practical, since if all moisture values are known a priori, estimation of moisture is no longer required. In practice, we derive a sorting index based on existing measured data, and use this index to sort all future values. This means data will not be exactly sorted; instead they will be approximately sorted in ascending order. The resulting fluctuations will show as high frequency low power harmonics which are negligible. A more detailed discussion on methods for robust sorting of the points of the field is out of scope of this paper. For instant, coarse-grained monotonic ordering is a good method (X. Wu et al., 2011).

Even if the best method for indexing the points is used, still when new data is sorted using this index, it shall not be sorted in increasing order exactly. For example, assume that the label 100 is dedicated to a point based on known information with sorting methods. This point may be 105^{th} or 90^{th} in the next moment. This fluctuation can be even larger in some situations.

The entire data set that contains sorted values at 6095 points is shown in **Fig 2**. Due to restrictions on using the simulator, the same data set that is used to solve the optimization problem is also used to calculate the residual error. Ideally this should be carried out over at least two separate data sets; something which we should address future extensions of this work. For now, we create two artificially distorted data sets based on the original one to provide some data richness. These are obtained by manual perturbation of the data sorting indices and shall be used to illustrate the sensor placement algorithm performance in instances in which the actual values are not sorted exactly. These two data sets are shown in **Fig 3**. **Fig 3 (a)** shows a data set with low sorting distortion and **Fig 3 (b)** shows a data set with high sorting distortion.

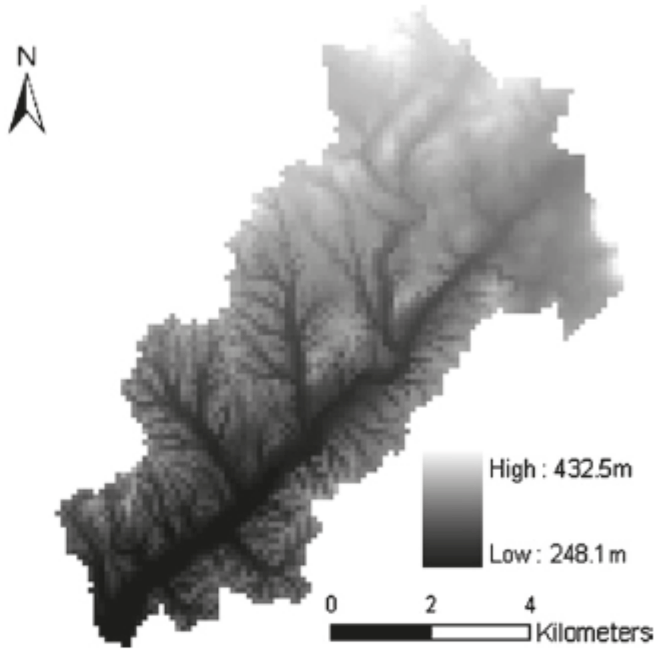


Fig 1. Peacheater Creek Watershed

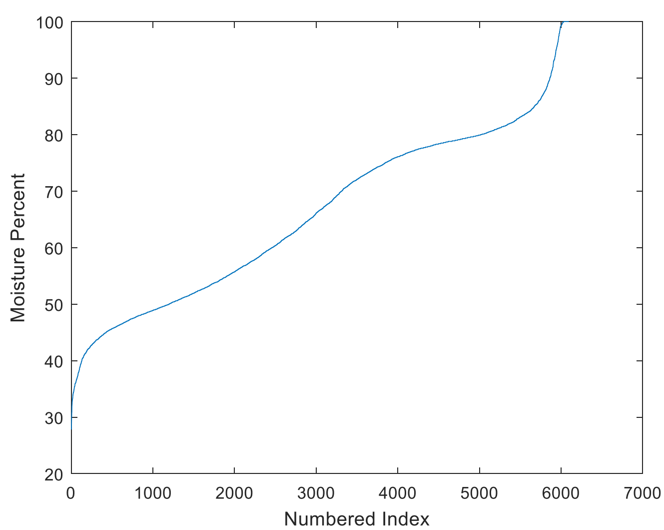
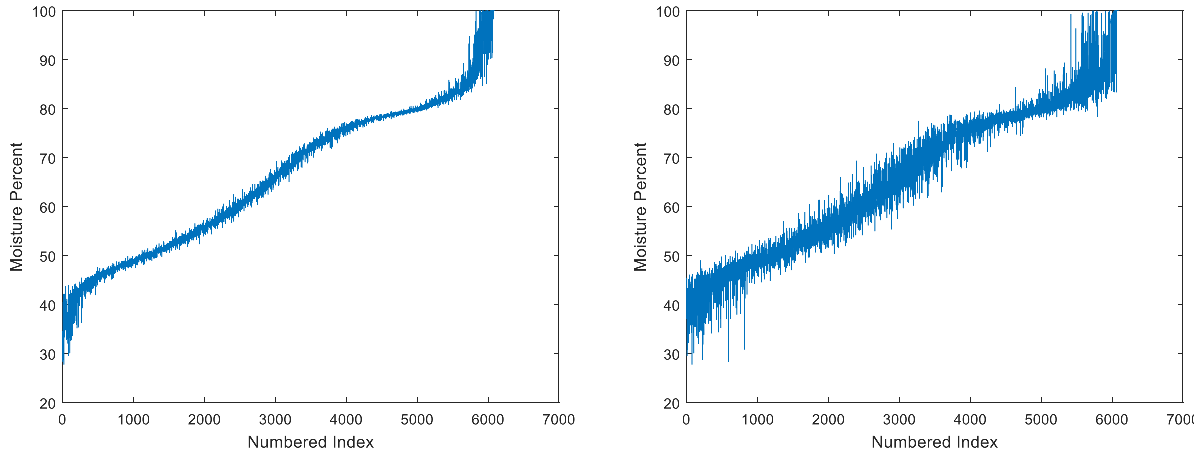


Fig 2. Whole data at 6095 points that is used as training data for sensor selection algorithm.



(a) (b)

Fig 3. Data sets that are used as testing data. (a) Data with low sorting distortion. (b) Data with high sorting distortion.

Novel Algorithm for Sensor Placement

Suppose that different moisture values have uniform distribution over the field. In the other words, if the moisture values are ordered in increasing order, it would be possible to fit a straight line to the data values. In this situation, random sensor placement or uniformly distributed sensor placement would deliver adequate results (obviously of the fit is a straight line exactly, only two sensors would be sufficient to estimate the entire data). The main drawback of the random placement is in the cases that variations of the values of sorted data is far from uniform.

For example, consider the data shown in **Fig 2**. The moisture values at the two extremes exhibit much more variation than those corresponding to the middle portions. In such instances, the typical random sensor placement, or even a mesh grid placement will perform poorly because high variation data is under sampled and low variation data will be over sampled. If the objective is to use as few number of sensors as possible, the algorithm for the placement cannot be oblivious to these changes in the variations.

From **Eq. 7**, finding the optimal sensor placement is equivalent to finding the best selection matrix $\boldsymbol{\Phi}$ such that solution of optimization problem of **Eq. 7** has minimum error with respect to the actual values. This is a Mixed Integer NonLinear Programming (MINLP) problem. Even if the number of sensors *M* is assumed to be known, in order to find the optimal sensor placement, it is required to solve the optimization problem *T* times, where $T = \binom{N}{M}$. By considering the fact that the number of sensors is not known a priori, the problem becomes even more complex and infeasible to solve for large scale problems. Instead we propose an iterative suboptimal algorithm which produces a good enough solution, but at a fraction of the computational cost. Although the proposed algorithm is not optimal, it will be shown that it outperforms random placement by a large margin.

The algorithm, starts by placing one sensor randomly. At each subsequent step, we solve the optimization problem **Eq. 7** and find the estimated values. The point whose estimation has the largest error is then determined and is used as the candidate for placement of the next sensor. This is continued until all desired number of sensors has been placed. This means at most we have to solve as many optimization problems as the number of sensors we wish to place.

Consider I_j as a 1 by *N* vector which has 1 one at *j*th entry and *N*-1 zeros. The iteration steps are as follows:

1. Set first sensor at a random place. Choose *i* as a random value between 1 and *N*. set k=1.
$$\Phi_1 = I_i. \tag{12}$$
2. Solve optimization problem and find estimated values $\hat{x}$:

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_1 \text{ subject to } y = \Phi_k \Psi \alpha, \quad (13)$$

$$\hat{x} = \Psi \hat{\alpha}. \quad (14)$$

3. Find the point that has the worst estimation:

$$i = \arg\max_{i=1,\dots,N} |x(i) - \hat{x}(i)|. \quad (15)$$

4. Place next sensor at location *i*.

$$\Phi_{k+1} = \begin{pmatrix} \Phi_k \\ I_i \end{pmatrix}. \quad (16)$$

5. Terminate if *k* reaches to specific number or estimation error becomes less than specific value. Otherwise, increment *k* and go to step 2
6. Φ_k states sensor locations.

Numerical Experiemnts

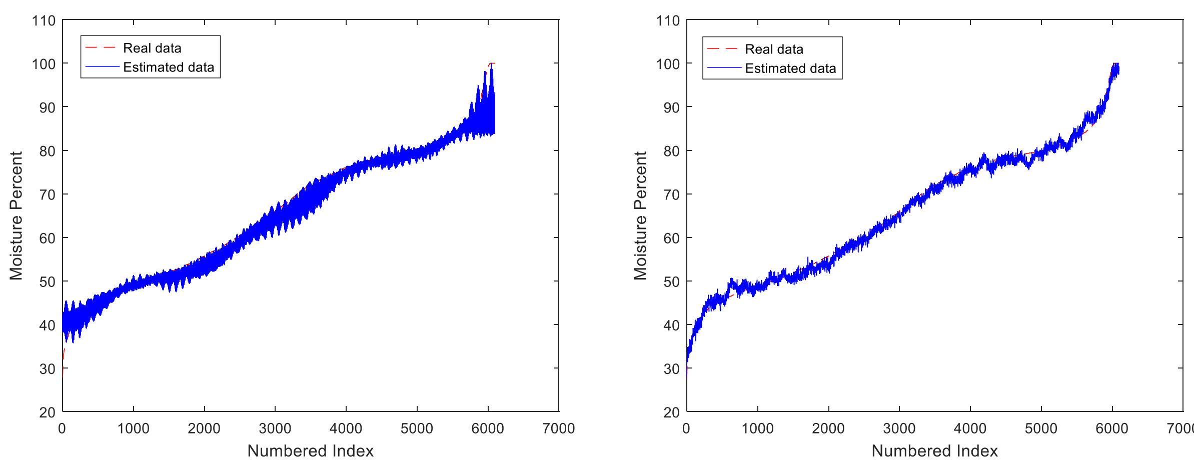
The data set of **Fig 2** is used as training data and the algorithm is applied to it. The estimation is carried using the selection matrix obtained from the proposed algorithm and random sensor placement using 70 sensors. The results are shown in **Fig 4**. **Fig 4 (a)** is related to random sensor placement and **Fig 4 (b)** is related to the proposed algorithm. Obviously, the better estimation is performed by the sensor placement with the proposed algorithm rather than random placement with the same number of sensors. The sensor locations are shown in **Fig 5**. In the proposed algorithm the number of sensors located in the first and last clusters with high variations of values is more compared to random sensor placement. Instead, the number of sensors in parts with lower variation is decreased.

Since comparing results of the different sensor placement on a special case is not enough for a general conclusion, the problem is solved with both methods by using a different number of sensors from 50 to 300. RMSE is selected as performance criterion for comparing the results.

The Root Mean Square Error (RMSE) is defined as follows,

$$RMSE = \frac{\|\hat{x} - x\|_2}{\sqrt{N}}. \quad (17)$$

The results of both methods with using different number of sensors are shown in **Fig 6**. When the permitted sensor number is high, random sensor placement performs as well as the placement algorithm. The main difference is when the number of sensors is restricted in which case random sensor placement compares rather poorly.



(a) **(b)**

Fig 4. Estimation of data by using *M*=70 sensors with different methods for sensor placement. (a) Random sensor placement. (b) Novel algorithm.

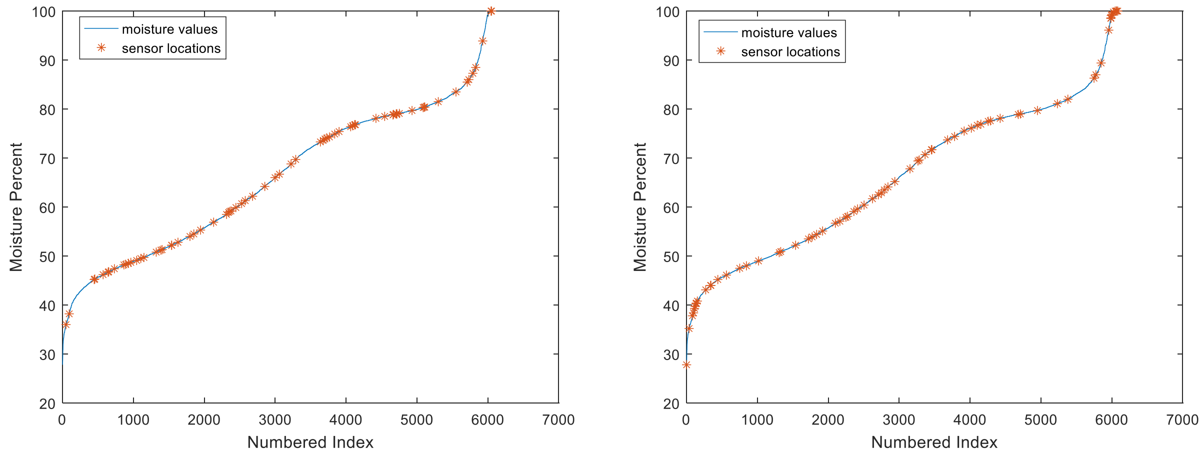


Fig 5. Sensor locations by using *M*=70 sensors with different methods for sensor placement. (a) Random sensor placement. (b) Novel algorithm.

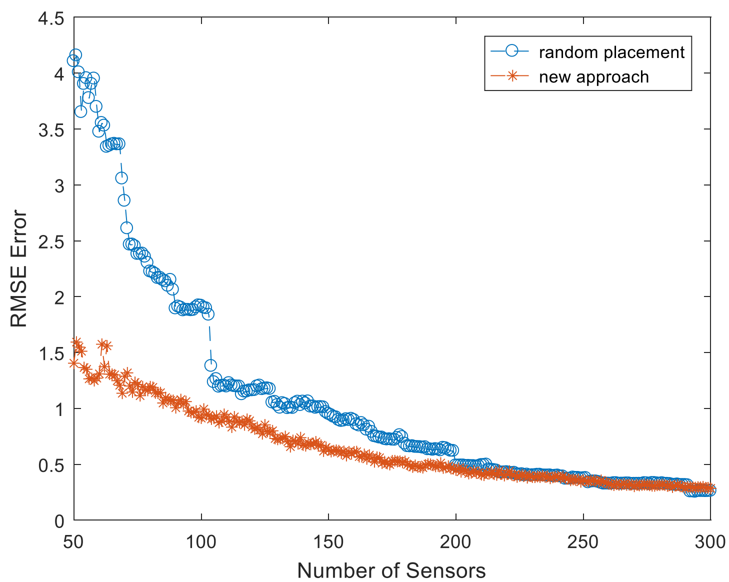
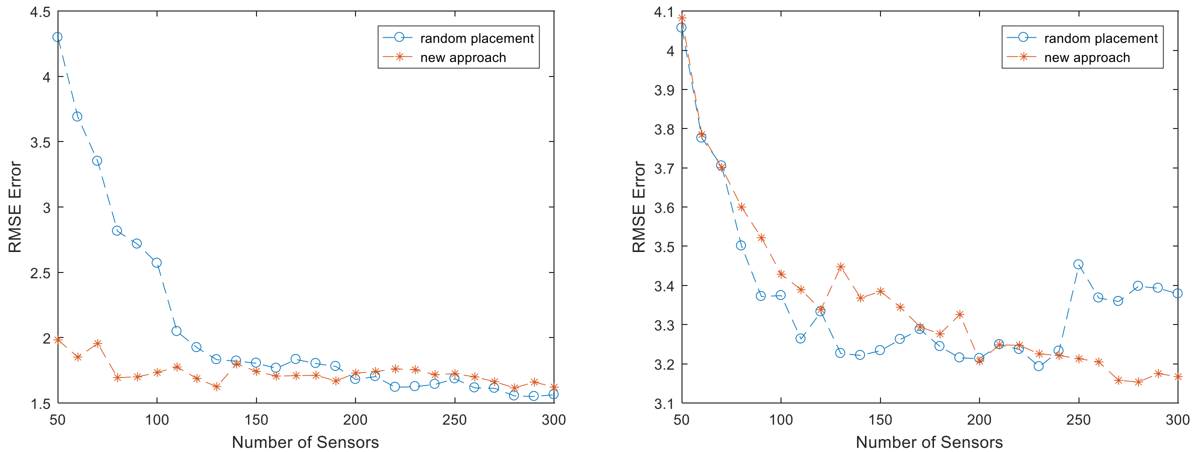


Fig 6. Comparison of random sensor placement and new algorithm using different number of sensors with criterion RMSE.

As discussed previously, in practice, the indexes for the points are dedicated using last measured data values and the same indexes are used to sort future data. Therefore, it is almost certain that future data would not be sorted exactly in ascending order.

To determine the sensitivity of the algorithm, data set of **Fig 2** is used to solve the sensor placement problem, but the distorted data sets in **Fig 3** are used to measure the error of estimation. For convenience, we refer to data set of **Fig 3 (a)** with low sorting distortion as data set **A** and that of **Fig 3 (b)** with high sorting distortion as data set **B**. The comparison of the results by using sensors number *M* from 50 to 300 in steps 10 is illustrated in **Fig 7**. As seen, if the values are not sorted exactly but the amount of sorting distortion is reasonable such as data set **A**, the placement algorithm maintains a performance advantage. However, if the indexes are distorted severely, the performances are in par.



(a) **(b)**

Fig 7. Comparison of the results of random sensor placement and new algorithm for sensor placement using different number of sensors. Exactly sorted data is used as training data set for finding sensor locations and the same locations are used for two other sets of data as test data to see performance of sensor placement methods in the case of not well sorted data. (a) Using data set with low sorting distortion as test file. (b) Using data set with high sorting distortion as test file.

In summary, while the placement algorithm delivers superior estimation performance, it has some drawback which should be addressed in future work.

The first drawback of the proposed algorithm is computational issues. For deploying *M* sensors over a field, the optimization problem **Eq. 7** should be solved *M* times. For very large scale problems and in cases with computational resources restriction, this can become an operational bottleneck.

Another drawback of the algorithm is related to the Simpson's paradox (Kock & Gaskins, 2016). The proposed placement algorithm is based on forward selection. Although in each step the best location for the next sensor is chosen, there is no guarantee that if two or more sensors are chosen simultaneously instead of one by one, the optimal sensor location would be the same as when sensors are placed individually. Similarly, if *M* sensors are selected one by one, it cannot be claimed that the optimal sensor placement is achieved.

Conclusion

In this paper, compressive sensing theory was formulated for the moisture estimation problem. Since finding the optimal sensor placement requires solving a large scale MINLP problem, it is very hard and impractical to find the true optimal solution. Instead, an iterative algorithm is proposed and applied to the problem. The results show that the algorithm has better performance than random sensor placement, especially in the case that number of used sensors is low. This performance advantage disappears if the sorting of the data is significantly distorted.

References

Blumensath, T., & Davies, M. E. (2009). Stagewise Weak Gradient Pursuits. *IEEE Transactions on Signal Processing, 57*(11), 4333-4346.

Candes, E. J., & Wakin, M. B. (2008). An Introduction to Compressive Sampling. *IEEE Signal Processing Magazine, 25*(2), 21-30.

Candes, E. J., Wakin, M. B., & Boyd, S. P. (2008). Enhancing Sparsity by Reweighted l1 Minimization. *Journal of fourier analysis and applications, 14*, 877-905.

Donoho, D. L. (2006). Compressed Sensing. *IEEE Transactions on Information Theory, 52*(4), 1289-1306.

Gorodnitsky, I. F., & Rao, B. D. (1997). Sparse Signal Reconstruction from Limited Data Using FOCUSS: a re-weighted minimum norm algorithm. *IEEE Transactions on Signal Processing, 45*(3), 600-616.

Gruhier, C., Rosnay, P. d., Hasenauer, S., Holmes, T., Jeu, R. d., Kerr, Y., . . . Zribi, M. (2010). Soil moisture active and passive microwave products: intercomparison and evaluation over a Sahelian site. *Hydrology and Earth System Sciences*.

Gwak, Y., & Kim, S. (2016). Factors Affecting Soil Moisture Spatial Variability for a Humid Forest Hillslope. *Hydrological Processes, 31*(2), 431-445. doi:10.1002/hyp,11039

Hamouda, Y. E. M., & Elhabil, B. H. Y. (2017). *Percision Agriculture for Greenhouses Using a Wireless Sensor Network*. Paper presented at the 2017 Palestinian International Conference on Information and Communication Technology, Gaza City, Palestinian Authority.

Kock, N., & Gaskins, L. (2016). Simpson's paradox, moderation, and the emergence of quadratic relationships in path models: An information systems illustration. *International journal of Applied Nonlinear Science, 2*(3), 200-234.

Mallat, S. G., & Zhang, Z. (1993). Matching Pursuits with Time -Frequency Dictionaries. *IEEE Transactions on Signal Processing, 41*(12), 3397-3415.

Needell, D., & Vershynin, R. (2010). Signal Recovery From Incomplete and Inaccurate Measurements Via Regularized Orthogonal Matching Pursuit. *IEEE Journal of Selected Topics in Signal Processing, 4*(2), 310-316.

Patel, V. M., & Chellappa, R. (2013). *Sparse Representations and Compressive Sensing for Imaging and Vision*. New York: Springer.

Rogers, P. P., Llamas, M. R., & Martinez-Cortina, L. (2006). *Water Crisis: Myth or Reality?* London: Taylor & Francis.

Tropp, J. A., & Gilbert, A. C. (2007). Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit. *IEEE Transactions on Information Theory, 53*(12), 4655-4666.

Vivoni, E. R., Teles, V., Ivanov, V. Y., Bras, R. L., & Entekhabi, D. (2005). Embedding Landscape Processes into Triangulated Terrain Models. *International Journal of Geographical Information Science, 19*(4), 429-457.

Wu, S., Li, J., & Huang, G. H. (2007). Modeling the effects of elevation data resolution on the performance of topography-based watershed runoff simulation. *Environmental Modelling & Software, 22*(9), 1250-1260.

Wu, X., Wu, Y., Liu, M., & Zheng, L. (2011). *In-Situ Soil Moisture Sensing: Efficient Random Sensor Placement and Field Estimation using Compressive Sensing*. Paper presented at the 7th International Conference on Wireless Communications, Networking and Mobile Computing, Wuhan, China.

Zhang, Q. (2015). *Precision Agriculture Technology for Crop Farming*. Washington: CRC Press.