Slip Surface Localization in Wireless Sensor Networks for Landslide Prediction

Andreas Terzis
Computer Science Department
Johns Hopkins University
Baltimore, MD 21218
terzis@cs.jhu.edu

Annalingam Anandarajah
Civil Engineering Department
Johns Hopkins University
Baltimore, MD 21218
rajah@jhu.edu

Kevin Moore
Department of Engineering
Colorado School of Mines
Golden, CO 80401
kmoore@mines.edu

I-Jeng Wang
Applied Physics Lab
Johns Hopkins University
Laurel, MD 20723
I-Jeng.Wang@jhuapl.edu

ABSTRACT

A landslide occurs when the balance between a hill's weight and the countering resistance forces is tipped in favor of gravity. While the physics governing the interplay between these competing forces is fairly well understood, prediction of landslides has been hindered thus far by the lack of field measurements over large temporal and spatial scales necessary to capture the inherent heterogeneity in a landslide.

We propose a network of *sensor columns* deployed at hills with landslide potential with the purpose of detecting the early signals preceding a catastrophic event. Detection is performed through a three-state algorithm: First, sensors collectively detect small movements consistent with the formation of a *slip surface* separating the sliding part of hill from the static one. Once the sensors agree on the presence of such a surface, they conduct a *distributed voting* algorithm to separate the subset of sensors that moved from the static ones. In the second phase, moved sensors self-localize through a trilateration mechanism and their displacements are calculated. Finally, the direction of the displacements as well as the locations of the moved nodes are used to estimate the position of the slip surface. This information along with collected soil measurements (e.g. soil pore pressure) are subsequently passed to a Finite Element Model that predicts whether and when a landslide will occur.

Our initial results from simulated landslides indicate that we can achieve high degree of accuracy (in the order of cm) in the localization step as well as the slip surface estimation step of our algorithm. This accuracy persists as the density and the size of the sensor network decreases as well as when considerable noise is present in the ranging estimates. As for our next step, we want to evaluate the performance of our system in controlled environments under a variety of hill configurations.

IPSN'06, April 19–21, 2006, Nashville, Tennessee, USA.

Categories and Subject Descriptors

C.3 [**Special-Purpose and Application-Based Systems**]: Signal processing systems

General Terms

Algorithms, Design, Performance

Keywords

Landslide prediction, 3-D localization, sensor networks

1. INTRODUCTION

Landslides are geological phenomena causing significant loss of life and billions of dollars in damages each year. Although a basic understanding of the causes and behavior of landslides is available, systems that predict the occurrence of a landslide at a specific site do not exist.

There are two main reasons for this: First, much of the knowledge about landslides is qualitative and based on static measures. However, development of a landslide is a temporal process, which can take as long as a year to develop. Second, the phenomenology of landslides is fundamentally spatial in nature. Though single point-location measurements of soil properties can be helpful, to reliably infer the potential for a landslide in a given location, it is important to be able to characterize soil properties over a suitably-sized region. *Spatially-distributed sensing is essential before it will be possible to develop effective landslide prediction and warning systems.*

In our proposal, hills that have the potential for landslides are outfitted with a network of *sensor columns* placed in vertical holes drilled in a semi-regular grid pattern over the hill's surface. Each sensor column is equipped with a range of instruments such as strain gages, tensiometers, geophones, and pore pressure transducers that enable it to measure changes in its location over time as well as soil characteristics relevant to the landslide phenomenon. Sensor columns collaborate to detect the initial signs of a landslide and estimate the location of the *slip surface* – the surface separating the sliding part from the rest of the hill.

The proposed collaborative detection algorithm has three phases: **(i) Detection:** the formation of a slip surface is established during this initial phase based on sensor column deformations. We use a two-tier anomaly detection algorithm to avoid false positives caused by normal hill movements not related to a landslide. First, deformations at a single column have to be above a threshold to be relevant. Once a column has detected such a statistically significant deformation, it initiates the second tier detection algorithm that involves the collaboration of neighboring nodes. The insight behind this algorithm is that if a slip surface has actually formed then neighboring nodes must also experience consistent deformations. Otherwise the observed phenomenon is localized and no further action is taken. **(ii) Localization:** after the existence of a slip surface has been established, columns perform a collaborative localization task to determine the set $\mathcal{M}$ of nodes that moved and the new locations of the displaced nodes. We use a distributed voting algorithm to determine the nodes which moved. Next, the nodes in $\mathcal{M}$ are localized through trilateration. The updated node locations are calculated using collected range estimates between nodes in $\mathcal{M}$ and the set of stationary nodes ($\mathcal{M}^c$). These estimates are collected with the help of seismic sources and geophones placed at regular intervals on the sensor columns. **(iii) Estimation:** After the identities and locations of the moved nodes have been established, the location of the slip surface is approximated based on the direction of movement of the displaced nodes. Intuitively, the estimation algorithm positions the slip surface on the hypothetical boundary between the sets $\mathcal{M}$ and $\mathcal{M}^c$. Once the existence and approximate location of the slip surface are determined, nodes in the network start forwarding measured soil properties to an analysis station. The analysis station uses these measurements as input to a Finite Element Model (FEM) that predicts whether and when a catastrophic landslide will occur.

In this work we evaluate the distributed slip surface localization algorithm. Specifically, we investigate the impact of sensor density, sensor location, and ranging noise on prediction accuracy using simulated landslides generated by an detailed finite element model. Our initial results show that we can achieve high degree of accuracy (in the order of cm) in the localization step as well as the slip surface estimation step of our algorithm. This accuracy degrades gracefully as the density and the size of the sensor network decreases. Furthermore, we show that ranging noise does not propagate through successive layers of localization but instead the network is able to contain the ranging error by utilizing estimates by multiple neighbors at each sensor node.

2. BACKGROUND

2.1 Landslides

A landslide is an event where a block of earthen mass (sliding mass) slides downhill. As per the physics, its weight pulls the sliding mass down the hill (driving force), and the shear resistance (resisting force) at the potential slip surface (interface between the sliding mass and the hill) resists the movement. When the driving force exceeds the resisting force, a landslide occurs. For a given hill geometry, the driving force comprises the weight of the soil and/or rock in the sliding mass, weight of man-made structures and weight of rain/snow. The resisting force depends on cohesion, friction, stresses, stress-strain relationship of the material in the slip surface, and pore water pressure along the slip surface. The magnitudes of the driving and resisting forces change with time due to changes in weather conditions (*e.g.*, rain water leads to increased pore pressure and decreases the resisting force) and other time-dependent activities. For example, physical and chemical weathering decreases the resisting force over time. Material creep may lead to increased pore pressures and/or instability. Construction of a structure uphill increases the driving force. In addition, excavation at the toe changes the geometry of the hill, potentially driving a slope from stable to unstable state [7, 19].

While the overall physics underlying a landslide is understood, prediction of the onset of landslide is complicated by the following spatial-temporal factors:

- Geologic materials are **highly heterogeneous** with **spatially distributed properties**. Satisfactory characterization is needed for successful landslide prediction, but is a formidable task.
- The **temporal variation** of the driving and resisting forces dictates the landslide potential, but is difficult to forecast.

The difficulty to collect sufficiently large measurement sets has hindered landslide prediction thus far. Sensor networks on the other hand, can provide a wealth of field data and thus remove this road block towards effective landslide prediction.

2.2 Landslide Prediction Algorithm

Parameters governing the occurrence of a typical landslide are schematically shown in Figure 1(a), where h is the slope height, θ is the slope angle, $\omega(t)$ is the rate of water inflow due to rain and/or runoff, and x^i_{slip} are coordinates of point i on the slip surface. Typically, the hill may be underlain by a number of material strata or pockets, each with distinct set of properties, λ^k_i (property i of layer/pocket k). W is the weight of the sliding mass, which includes known (*e.g.*, weight of buildings) and unknown (*e.g.*, weight of soil) components. The latter depends on the density of soil/rock in different strata.

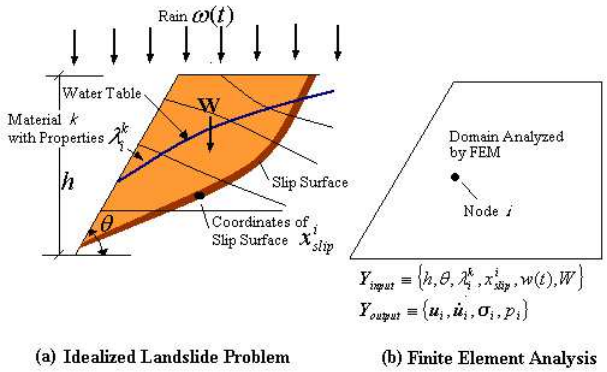

Figure 1: Schematic of landslide problem and the domain used for Finite Element Analysis.

Consider a point i in the vicinity of the potential sliding mass with Cartesian coordinates (x_i, y_i, z_i) at time $t = 0$.

The coordinates of this point at any time t is $(x_i + u_i^1, y_i + u_i^2, z_i + u_i^3)$, where the vector $u_i = (u_i^1, u_i^2, u_i^3)$ is known as the displacement vector. Our primary interest is to predict the variation of displacements with time, $u_i(t)$. In general $u_i(t)$ changes slowly, whereas a slide may develop over hundreds of days before it leads to a catastrophic failure (Figure 2). Development of an effective warning system requires the family of functions $u_i(t)$ to be predicted accurately before such a catastrophic event occurs.

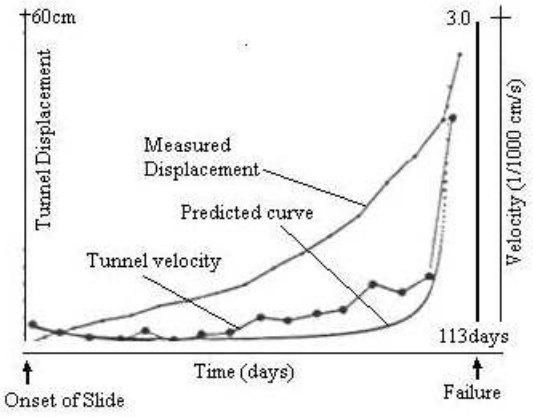

Figure 2: Progression of a Landslide (reprinted from [9]).

The finite element method ([20]) has the potential to predict $u_i(t)$, provided that the underlying physics is modeled accurately and the values for input parameters are estimated accurately. The parameters shown in Figure 1(a) are the input parameters, which are grouped in a vector as $Y_{input} = \{h, \theta, \lambda_i^k, x_{slip}^i, \omega(t), W\}$. Given Y_{input}, the FEM allows the time history of displacements $(u_i(t))$, velocities$(\dot{u_i}(t))$, stresses (s_i) and pore water pressures (p_i) at any point i in the slope to be calculated. We can group these output variables in a vector as $Y_{output} = (u_i, \dot{u_i}, s_i, p_i)$. The input and output parameters are shown in Figure 1(b).

Y_{input} can be further divided into two parts as $Y_{input} = Y_{input}^{known} + Y_{input}^{unknown}$, where Y_{input}^{known} includes all easily measurable quantities such as h, θ, $\omega(t)$, and a portion of W (*e.g.*, weight of a building). Also, some of the soil properties (e.g, shear modulus) and the location of the slip surface x_{slip}^i can be calculated from sensor readings (without having to carry out the FEM analysis), and treated as known parameters. All remaining parameters are included in $Y_{input}^{unknown}$, which is to be back calculated from the sensor readings.

The FEM calculates Y_{output} at any point i, while the sensors are placed only at a subset of the i points. Let Y_{output}^k be the FEM output at the nodes where sensors are placed, S^k be the corresponding sensor readings. In an ideal case where $Y_{input}^{unknown}$ is known and the FEM represents the mechanical behavior perfectly, $Y_{output}^k = S^k$. Hence an optimal value for $Y_{input}^{unknown}$ can be obtained by minimizing a scalar objective function defined in terms of the difference between Y_{output}^k and S^k [1–3]. This inverse analysis of system identification begins with a trial set and computes an optimal set by a suitable nonlinear optimization method. As the sensor data is gathered continuously, the procedure is be repeated at suitable time intervals and $Y_{input}^{unknown}$ is updated.

3. SYSTEM ARCHITECTURE

The proposed wireless sensor network depicted in Figure 3, consists of a collection of *sensor columns* placed inside vertical holes drilled during the network deployment phase and arranged on a semi-regular grid over the monitored area. The size of the grid depends on the characteristics of the site where the network can be deployed with grids scaling to thousands of sensor columns.

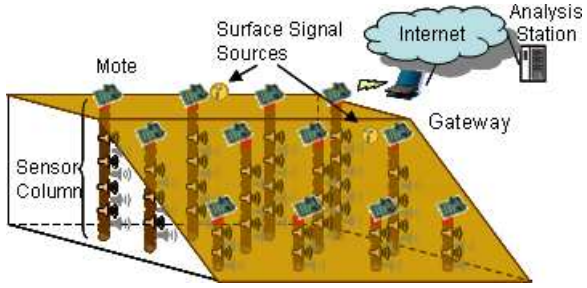

Figure 3: Landslide Warning System.

Each of the sensor columns has two components: The sensing component that is below ground and contains all the sensors and the computing component that stays above ground and contains the processor and radio module.

3.1 Sensor Column

Figure 4 shows one of the network's sensor columns. Each column includes four types of sensors: *geophones, strain gages, pore pressure transducers*, and *reflectometers* placed at regular intervals over the length of the column. Strain gages are placed on the surface of the tube along its vertical axis and indicate any bending of the tube caused by ground movements. Pore pressure transducers and reflectometers measure negative and positive pore pressure and moisture content at different depths of each column. Finally, geophones are placed on the outer hull of the tube at regular intervals. Their role is to measure the changes in distance between nearby columns at different depths. These sensors are controlled by a Stargate microserver [6] located at the end of the tube that is aboveground. Stargates are powered by rechargeable batteries that are periodically charged by solar panels.

An intuitive way to think about the system is as two cooperating networks: the above-ground network among Stargates and the below-ground network among sensors. The topside network (RF network) is used to pass information from the sensor columns to the analysis station and among Stargates to coordinate their measurements. The main task of the below-ground network is to collect displacement measurements using the geophones.

Geophone measurements are coordinated by the Stargates over the RF network. A column that sensed a movement (*e.g.* via its strain gages) notifies over the RF network columns that are within range, before sending an acoustic signal with its seismic source(s). Columns that receive the notification activate their geophones and associate the seismic signal with the corresponding column. Because RF signals travel much faster than the seismic signal, the time interval between the reception of the radio signal and the reception of the seismic signal can be used to calculate the distance between the seismic source and the local geophone.

Specifically, by multiplying the elapsed time with the speed of sound through soil, the distance between seismic source and geophone can be estimated. Since only local timestamps are used in distance calculations, precise time synchronization among the sensor network nodes is not necessary.

Unfortunately, the speed of sound is a function of the composition and water content of the soil and therefore a calibration mechanism is necessary. Our solution is based on the assumption that the initial column locations are known, allowing the *base speed of sound* to be calculated. This base speed is then adjusted to compensate for the measured changes in water content, using a set of calibration curves, to produce the *calibrated speed of sound*. The relative movements of geophones can then be calculated based on that calibrated speed.

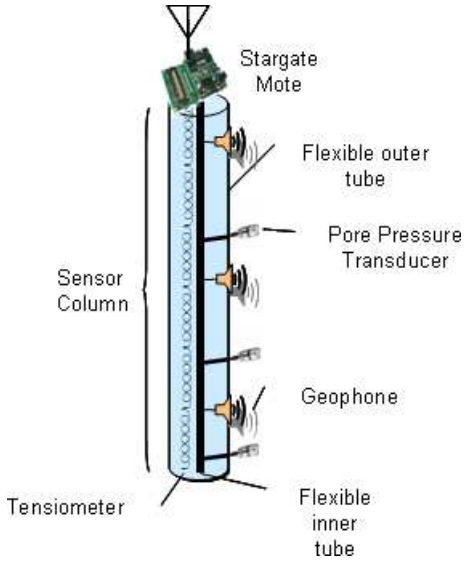

Figure 4: Sensor Column.

By powering off all the instruments (other than their strain gages), sensor columns can save considerable energy since for the majority of the time the hill will remain static. Furthermore, note that until the existence of a slip surface is confirmed, no measurements are sent to the analysis station, thus reducing the amount of data transmitted and further reducing energy consumption.

4. PROBLEM FORMULATION

Let $X_1(t), X_2(t), \ldots, X_N(t)$, where $X_i(t) = (x_i(t), y_i(t), z_i(t)) \in \mathbb{R}^3$ be the location of sensors at time t. For each sensor, we assume that distance estimates to sensors within range r are available. We refer to the set of sensors that lie within range r of node i as its neighbors and denote the set as $\mathcal{N}_i(t)$. Let $D(t) = [d_{ij}(t)]$ be a $N \times N$ measurement matrix that contains all the available distance information and defined by

$$d_{ij}(t) \triangleq \begin{cases} \|X_i(t) - X_j(t)\| & \text{if } i \neq j \text{ and } X_j(t) \in \mathcal{N}_i(t) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Assume that the initial locations of sensors at time t_0 are known. The sensors operate periodically at discrete time. That is, measurements $D(t)$ are updated at time $t_0, t_1, t_2, \ldots$ with $t_{k+1} = t_k + \Delta$. For convenience, we denote the measurements at time t_k by D_k.

Consider the scenario where a slip surface is formed at unknown time t^*. To simplify the problem, we can assume that $t^* = t_{k^*}$ for some unknown k^*. Given that our algorithms are incremental, we can drop the dependency on t in the subsequent discussions. Let

- $X_1, X_2, \ldots, X_N$ denote the *known* locations of sensors at time t;
- D denote the measurement matrix at time t (note that all the distances can be derived if needed since locations are known); and
- D' denote the measurement matrix available at time $t + \Delta$.

Based on this information, we develop distributed algorithms to: **(1)** Determine whether a slip surface was formed (at time t); **(2)** If so, estimate the new sensor location $X'_1, X'_2, \ldots, X'_N$ (at time $t + \Delta$); and **(3)** Compute the location of the slip surface.

We claim the above problem can be solved by solving a set of subproblems discussed in the following sections.

4.1 The detection problem

Detection of the formation of a slip surface is based on changes in the length of sensor column strain gages. We assume that the strain gages are properly indexed and denote the location of the centroid of strain gage i at time t by $X_i^s = (x_i^s, y_i^s, z_i^s)$. Denote the length of the strain gage i at time t as $l_i(t)$. The distributed detection algorithm will make the detection decision based on $\Delta l_i(t)$ defined as $l_i(t+\Delta) - l_i(t)$. The detection problem is then to detect the formation of a slip surface with unknown location based on distributed measurements of length changes in strain gages $\Delta l_i(t)$. Let's assume that the processing nodes are synchronized. We will drop the dependency on t in the following when it is clear from the context.

To formulate the detection as a hypothesis testing problem, let H_1 denote the hypothesis that a slip surface was formed, and H_0 otherwise. Assume that the slip surface can be characterized by

$$\psi(X, \Theta_s) = 0, \quad (2)$$

where Θ_s is a vector of parameters and $\psi\colon \mathbb{R}^3 \to \mathbb{R}$ is a scalar-valued function given a specific Θ_s[1]. To simplify the presentation, we will use $\psi(\Theta_s)$ to refer to the slip surface specified by Eq.(2). Note that Θ_s is a random vector and we assume that it has a prior density denoted by $p_{\Theta_s}(\Theta)$. It is expected that a strain gage will experience a (statistically) large change in its length only if it "intersects" with the slip surface from Eq.(2). We use I_i to represent the event where the slip surface "intersects" with the strain gage i and I_i^c to denote its complement. To determine precisely whether I_i is true requires the information on the location, the length, and the orientation of the strain gage i. We will make an approximation and assume that the intersection occurs if the center of the strain gage X_i^s is within ε_l distance from the slip surface. That is, $I_i \equiv \{dist(X_i^s, \psi(\Theta_s)) \leq \varepsilon_l\}$, where $dist(X, \psi)$ denotes the distance between $X \in \mathbb{R}^3$ and the slip surface $\psi(X, \Theta_s) = 0$.

[1]For a linear slip surface in $\mathbb{R}^3$, $\psi(X, \Theta_s) = ax+by+cz+d = 0$ with $X = (x, y, z)^T$ and $\Theta_s = (a, b, c, d)^T$.

With the notation defined above, the detection problem can be characterized by the following models for Δl_i:

$$p_{\Delta l_i}(l|H_0) = P_0(l), \quad (3)$$
$$p_{\Delta l_i}(l|H_1, I_i) = P_1^I(l), \quad (4)$$
$$p_{\Delta l_i}(l|H_1, I_i^c) = P_1^c(l), \quad (5)$$

where $P_0(l)$, $P_1^I(l)$, and $P_1^c(l)$ are the density functions of Δl_i conditioned on their associated hypotheses. We make a simplifying assumption that the distribution of Δl_i given the hypotheses is independent of i and the location of the specific strain gage. For example, we assume that Δl_i and Δl_j have the same distribution if the slip surface intersects with the both strain gages i and j. Furthermore, we assume that $\{\Delta l_i\}$ are mutually independent given the hypothesis (both H_1 and I_i). We will also write

$$P_i(l) = p_{\Delta l_i}(l|H_1) = P_1^I(l)P\{I_i\} + P_1^c(l)P\{I_i^c\}. \quad (6)$$

Note that $P_i(l)$ depends on the location of the strain gage and the slip surface.

Given the power constraint of the system, we use a two-tier approach to the detection problem. The lower tier consists of local detection based on local measurement at each strain gage. Once a local detection decision is made, the network will activate the second tier detection that involves collaboration of multiple nodes and takes into account the correlation induced by the global phenomenon (the formation of the slip surface).

Local detection is based on an outlier detection algorithm to detect statistically large length changes based on an empirical characterization of the null hypothesis distribution $P_0(l)$[2]. Initially, we assume a simple Gaussian model characterized by the estimated mean $\bar{m}_l$ and standard deviation $\bar{\sigma}_l$. A local detection is made if $|\Delta l_i - \bar{m}_l|/\bar{\sigma}_l \geq Q_G(\alpha/2)$, where $Q_G(\alpha/2)$ is the $(1-\alpha/2)$-quantile of a standard Gaussian distribution with zero mean and unit variance and α is a design parameter that dictates the detection and false alarm trade-offs of the local detection algorithm. The empirical mean $\bar{m}_l$ and standard estimation $\bar{\sigma}_l$ can be estimated from prior measurements either only locally at the node or within the node's neighborhood.

Once local positive detections are made, the neighboring strain gages collaborate to evaluate whether the local decisions are consistent with the hypothesis that a slip surface as parameterized by Eq.(2) is formed. This collaborative signal processing can reduce the false alarms generated from random local movements. If appropriate distributions on Δl_i can be defined, we can apply the message passing algorithm, similar to the one proposed in [17], for the collaborative detection. Without this additional knowledge, we propose a simple heuristic based on the robust regression technique. Assume that the strain gage i made a positive detection decision locally. We will correlate this information with local detection decisions at the strain gages within a neighborhood of strain gage i, denoted by $\mathcal{N}_i^d$ (including the strain gage i itself). According to the local detection decision, we further divide the set $\mathcal{N}_i^d$ into two subsets: $\mathcal{N}_i^{d+}$ that contains nodes with positive local detection, and its complement $\mathcal{N}_i^{d-}$. First we apply the robust regression technique [11] to fit a linear surface, $\psi(X, \bar{\Theta}_s)$ to the locations of nodes in $\mathcal{N}_i^{d+}$. Next, we compute the distance between nodes in $\mathcal{N}_i^d$ and the fitted linear surface $\psi(X, \bar{\Theta}_s)$. The detection hypothesis is then rejected if the number of nodes lying within ε_l from the fitted linear surface $\psi(X, \bar{\Theta}_s)$ in $\mathcal{N}_i^{d-}$ exceeds a threshold.

[2]It is possible to use the likelihood ratio $\frac{P_i(l)}{P_0(l)}$ for detection if knowledge of $P_1^I(l)$, $P_1^c(l)$, and $p_{\Theta_s}(\Theta)$ are available.

4.2 The classification problem

Once we determine that a slip surface was formed, we decide first which sensors are above (and hence have moved) and which are below the slip surface (and hence have *not* moved). We will refer to this problem as the classification problem.

A simple distributed heuristic can be developed based on the following insights:

- The distance between two nodes below the slip surface should not change (at least not much statistically) since both of them have not moved;
- The distance between two points *across* the slip plane is likely to change;
- The distance between two nodes above the slip surface would see a small change since they moved somewhat together; and
- The nodes located closest to the known rigid part of the structure are unlikely to move. We will refer to these nodes as the *anchor* nodes.

If the sensor range is selected properly so that the neighborhood of any node is "local" (not empty and does not contain a large portion of the nodes), we claim that the following simple iterative heuristic solves the distributed classification problem:

- Let $S_i \in \{0, 1, U\}$ be the state of node i, where "0" represents the "NOT MOVED" decision, "1" the "MOVED" decision, and "U" the "UNDECIDED" decision.
- Initialize the states of the anchor nodes to 0 and states of the rest to U.
- For every node i that is not an anchor node, we update its state based on the majority vote resulting from a simple voting within its neighborhood (including itself). The voting rules are:
  1. Nodes with state U do not get to vote. All the others get one vote each.
  2. The node itself votes based on its state S_i.
  3. Neighbor j with zero or small Δd_{ij} votes based on its state S_j.
  4. Neighbor j with large Δd_{ij} votes based on the negation of its state $\bar{S}_j$.
- Do not change the state S_i if the voting resulted in a tie.
- Repeat the above voting-based update until all $S_i \in \{0, 1\}$ or for some number of iterations.

The convergence of the distributed algorithm given above can be established under appropriate assumptions on the neighborhood structure. The simple heuristics described above is similar to the the max-product algorithms [16] for solving the maximum a posteriori (MAP) problem.

4.3 The localization problem

After we classify the sensor nodes to those that moved (or lie above the slip surface), $\mathcal{M}$, and those that stayed stationary (or lie below the slip surface), $\mathcal{M}^c$, the network will localize the nodes in $\mathcal{M}$ via trilateration. Based on the new locations determined by this process, the displacements for nodes in $\mathcal{M}$ can be estimated. We denote the displacement vector of node i in $\mathcal{M}$ by $u_i = X_i' - X_i$. The displacement vectors will be used to estimate the location of the slip surface and transmitted back to the base station for the necessary FEM inverse mapping.

Localization of any node in $\mathcal{M}$ with enough stationary neighbors[3] in $\mathcal{M}^c$, can be achieved by solving a weighted least squares problem using the known locations of stationary neighbors and the associated distance measurements. We apply a distributed approach presented in [5]. Specifically, consider the localization of node $i \in \mathcal{M}$. Assume that there are enough nodes in $\mathcal{N}_i \bigcap \mathcal{M}^c$. Then the location of node i, X_i' can be estimated by minimizing the following objective function

$$J = \sum_{j \in \mathcal{N}_i \bigcap \mathcal{M}^c} w_{ij} \left(\|X - X_j\| - d_{ij} \right)^2, \tag{7}$$

where X_j's are the known locations of stationary neighbors and w_j is a nonnegative scalar that reflects the quality of the distance measurements[4]. By the first-order necessary condition, a local minimum of J, denoted by X_i^*, should satisfy

$$X_i^* = \frac{\sum_{j \in \mathcal{N}_i \bigcap \mathcal{M}^c} \left[w_{ij} X_j + \frac{w_{ij} d_{ij} (X_i^* - X_j)}{\|X_i^* - X_j\|} \right]}{\sum_{j \in \mathcal{N}_i \bigcap \mathcal{M}^c} w_{ij}} \tag{8}$$

The new location of node i, X_i', can be obtained by applying Eq.(8) iteratively with an initial estimate $\hat{X}_i^0$ derived from trilateration:

$$\hat{X}_i^{k+1} = \frac{\sum_{j \in \mathcal{N}_i \bigcap \mathcal{M}^c} \left[w_{ij} X_j + \frac{w_{ij} d_{ij} (\hat{X}_i^k - X_j)}{\|\hat{X}_i^k - X_j\|} \right]}{\sum_{j \in \mathcal{N}_i \bigcap \mathcal{M}^c} w_{ij}} \tag{9}$$

Convergence of this recursive scheme is guaranteed if $\hat{X}_i^0$ is sufficiently close to the global minimum. However, identification of good initial estimates directly by trilateration from only range information can be a challenging task in three-dimensional space. On the other hand, since in our case the locations prior to the movements $\{X_i\}$ are known, the ambiguities for graph realization [15] can be addressed assuming an upper bound on the amount of displacement $\|u_i\|$.

For nodes in $\mathcal{M}$ that do not have enough stationary neighbors, the same technique can be applied as some or all of their neighbors are localized. This can be done sequentially (that is, one node at a time) or in a completely distributed manner by updating the location based on Eq.(8) at every node in parallel (as suggested in [5]).

4.4 Slip surface estimation

Once the new locations of displaced nodes are determined, the slip surface $\psi(X, \Theta_s)$ can be estimated by solving an optimization problem:

$$\max_{\Theta_s} \left\{ \min \left\{ dist(\mathcal{M}, \psi(\Theta_s)), dist(\mathcal{M}^c, \psi(\Theta_s)) \right\} \right\}, \tag{10}$$

subject to

- $\psi(X, \Theta_s)$ is "consistent" with $U_{\mathcal{M}} \triangleq \{u_i = X_i' - X_i : i \in \mathcal{M}\}$, and
- $\mathcal{M}$ and $\mathcal{M}^c$ are separated by $\psi(X, \Theta_s) = 0$.[5]

Where $dist(\mathcal{M}, \psi(\Theta_s))$ is the distance between the set $\mathcal{M}$ and the slip surface (2) and is defined by

$$dist(\mathcal{M}, \psi(\Theta_s)) \triangleq \min \left\{ dist(X_i, \psi(\Theta_s)) : i \in \mathcal{M} \right\}.$$

The objective function used in the optimization problem of Eq.(10) is referred to as the *margin* in statistical learning theory where models maximizing the margin have been shown to generalize well. The appropriate notion of consistency in the optimization constraint will depend on the specific parameterization used in $\psi(\Theta_s)$. For example, if a linear slip surface is assumed, then we can say $\psi(\Theta_s)$ is consistent with the displacements $U_{\mathcal{M}}$ if the slip surface is parallel to the median of $U_{\mathcal{M}}$. Here the median of $U_{\mathcal{M}} \subset \mathbb{R}^3$, denoted by $\bar{u} = (\bar{u}_x, \bar{u}_y, \bar{u}_z)^T$ is defined as a vector composed of the medians in each coordinate. That is, $\bar{u}_x$ is the median of $\{u_i^x : u_i = (u_i^x, u_i^y, u_i^z)^T, i \in \mathcal{M}\}$. An alternative definition based on the average of displacements is possible but could result in undesirable sensitivity to the presence of outliers.

5. EVALUATION

In order to judge the feasibility of the proposed wireless sensor network, we evaluate the performance of the slip surface localization algorithm presented in the previous paragraph. In this initial study, we chose to focus on a set of *structural* questions that need to be answered adequately before any further practical experimentation can be justified. Specifically, we are interested in the following three questions: **(1)**) What is the effect of *sensor node density* on prediction accuracy? **(2)** What is the effect of the *"geometry" of the sensor grid* on prediction accuracy, and **(3)** How does *ranging noise* affect prediction accuracy?

We use an abstract model of the sensor network that assumes all network communications are error-free and no sensor nodes fail. While such an environment is obviously unrealistic, we use this approach to determine the *upper bound* on the accuracy of the proposed algorithms. More realistic tests that evaluate the loss in accuracy due to realistic deployment conditions are the subject of future work.

5.1 Methodology

We simulated the performance of our algorithms using the virtual hill shown in Figure 5. We use a two-dimensional hill and we approximate the slip surface with a slip plane. The height h of the hill is 40m and we placed sensor columns at regular intervals (3m) and up to depth of d meters. The acoustic range of a sensor node is r meters.

[3]In principal, we need at least $p+1$ stationary neighbors to localize the node , where p is the dimension of the problem.
[4]For example, we can let $w_{ij} = \sigma_{ij}^{-2}$, where σ_{ij} is the standard deviation of d_{ij}

[5]If the two sets cannot be separated by any surface with the chosen parameterization, then we will select Θ_s such that the surface separates the largest subsets.

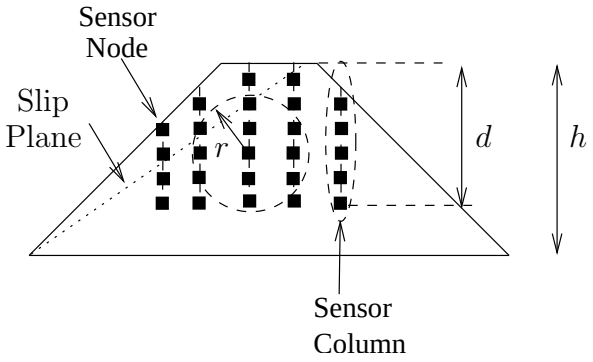

Figure 5: Simulated Hill.

We simulated hill movements using the Finite Element Model developed by one of the authors [1]. Through this model we were able to create a virtual landslide across the slip plane shown in Figure 5 and recorded the actual displacements of all the sensor nodes in the network. We treat the displacements calculated by the FEM model as the "ground truth" and compare them with the displacements calculated by our localization algorithm.

Specifically, we use two metrics to measure the accuracy of the proposed algorithms: **(1)** the average, as well as the standard deviation of the error introduced by the localization algorithm (§4.3) measured over all the sensor nodes that moved, and **(2)** the maximum distance between two points located on the actual and the estimated slip plane. This second metric captures the worst spatial deviation between the calculated and real slip planes and provides an estimate of how well we can predict its location in the presence of localization errors and node mis-classifications between sets $\mathcal{M}$ and $\mathcal{M}^c$.

5.2 Node Density

In order to evaluate the effect of node density on prediction accuracy, we vary sensor range r. Table 1 indicates that the localization algorithm is able to estimate the location of the moved nodes with high degree of accuracy over all the sensor ranges.

Node Density	Mean Error (m)	Std Dev
9.64	0.00269	0.000157
16.45	0.0023	0.000142
19.37	0.0026	0.00018
24.63	0.0022	0.00011
29.99	0.0019	0.00012
33.54	0.0018	0.00010

Table 1: Localization error as a function of node density ($d = 40$).

Because the locations of the moved sensor nodes can be estimated with high accuracy, the slip surface estimation also produces very accurate results, as Table 2 illustrates.

5.3 Grid Geometry

Next we investigate the effect of the maximum sensor column depth d necessary to achieve high degree of accuracy. This is an important consideration, since drilling the holes for the sensor columns is a labor-intensive and expensive process.

Node Density	Slip Plane Localization Error
9.64	0.001710
19.37	0.008774
24.63	0.008774
29.99	0.006744
33.54	0.003637

Table 2: Slip Plane Localization error as a function of node density.

Max Node Depth	Slip Plane Loc. Error
15	4.07542
20	3.38964
25	2.35554
30	1.68701
35	0.04812

Table 3: Slip Plane Localization error as a function of maximum node depth. ($r = 9.64$)

Table 3 presents the slip plane localization error as the maximum depth d increases. It is evident from this table that the maximum depth plays a profound role in accuracy. On the other hand, even at intermediate depths the localization error is moderate if we keep in mind that the calculated slip plane location is only used as initial input to the FEM model and will be subsequently refined by that model.

5.4 Noise

In our final set of experiments we investigate the effect of ranging noise in localization accuracy. Specifically we are interested to estimate how ranging noise *propagates* as nodes that have moved during the landslide are localized with the help of other displaced nodes that have been previously localized. Given the potentially large size of the deployed network, the slip surface localization algorithm will produce inaccurate results if noise propagation is severe.

Error Std. Dev.	Mean Error (m)	Std Dev
0.1	0.1121	0.0547
0.2	0.1466	0.0753
0.3	0.1871	0.0878
0.4	0.2135	0.1252
0.8	0.3864	0.2210

Table 4: Localization error as a function of noise in range measurements

We used a Gaussian variable with zero mean and variable standard deviation to model the noise in ranging estimates. Table 4 presents the results from our experiments for a network of $r = 33.54$ under increasing levels of noise. As is evident from this Table, the average localization error grows slower than the noise level while the deviation of the localization error grows with the level of noise. These very encouraging results are due to the high degree of network connectivity that enables node to "cancel" the noise in range estimates by utilizing the multitude of estimates from their neighbors.

As our final test, we decrease node density to $r = 24.99$ and $r = 16.45$. In those cases the average localization error was 0.48m and 0.65m respectively, showing that the hypothesized noise propagation does not occur even in networks with lower density. These results indicate that increasing the network's density is a promising strategy to cancel the negative effects of considerable ranging noise that we expect to find in geophone measurements.

6. RELATED WORK

Landslide Prediction. Several mechanistic analysis methods have been used in the past. Simplified momentum transfer models (*e.g.*, [12], [10]) and fluid models [14] have been used for predicting the spread of debris (i.e., distance traveled by debris) during a landslide. GIS-based, empirical models have been developed to predict landslide potentials (*e.g.*, [13], [4]). In some cases, such models have been enriched with hydrological models (e.g., [8]). Our work focuses on the behavior from the onset of movements until a complete slide, rather than the debris flow incurred after-wards.

Very little work is available on analyses for predicting the failure time. For example, [9] used a simplified, one-dimensional creep model to simulate the observed failure time for the Takabayama (Japan) landslide, which occurred on 22 January, 1970. The hill consisted of a clayey soil. A field displacement transducer (installed in a tunnel) indicated that the landslide occurred 113 days from the time the displacement began increasing (shown in Figure 2). However, it is important to note that Fukuzono's work is an after-the-fact analysis while evidence of successful a-priori failure time prediction cannot be found in the literature.

Sensor Networks. Application of wireless sensor network technologies to the landslide prediction problem is only a recent concept. Besides the work presented here, a proposal and initial design based on a network of strain gages is presented in [18]. That proposal focuses on the detection of specific strain signatures in rocks near the surface (with strain gages operating at low depth around 25–30cm) that have been identified as a potential prelude to a landslide. Our approach relies on the more detailed FEM with an objective to provide a general framework for early and credible prediction that can be applied to areas with diverse geological characteristics. To provide spatially-distributed information necessary for proper instantiation of a FEM, our sensor network designs involve a heterogeneous set of sensors operating at a much greater range of depth. The relative efficacy between the two proposed approaches in terms of the accuracy and the lead-time for prediction cannot be determined without realistic experiments and actual field tests.

7. SUMMARY

In this work we proposed a wireless sensor network for the prediction of landslides. Such prediction is enabled by the fact that minute hill movements can be sensed days or even months before a catastrophic landslide occurs. Our proposed sensor network uses a collection of instruments to detect such movements and collectively estimate the displacements of sensor nodes embedded in the hill under observation. Through these estimates the location of a slip plane is estimated and passed to an analysis station along with pertinent soil measurements. These measurements are used to back-calculate the input parameters of a Finite Element model used to predict the possibility and the onset of a catastrophic landslide.

Our preliminary simulation results are encouraging as they show that the proposed localized algorithms estimate the displacements as well as the location of the slip plane with small errors over a wide range of sensor ranges, deployment strategies, as well as ranging noise. Emboldened by these results, we are currently building a small scale prototype of the system that we will test over medium-size artificial hills to evaluate the performance of our system under realistic conditions.

8. REFERENCES

[1] A. Anandarajah. HOPDYNE: A finite element computer program for the analysis of static, dynamic and earthquake soil and soil-structure systems. Technical report, Civil Engineering Report, Johns Hopkins University, 1990.

[2] A. Anandarajah. Numerical Methods for Seismic Analysis of Dams. In *International Workshop on Seismic Stability of Tailings Dams*, Nov. 2003.

[3] A. Anandarajah, J. Zhang, and C. Ealy. Calibration of dynamic analysis methods from field test data. *Journal of Soil Dyn. Earthquake Engineering*, 2005.

[4] A. Carrara and F. Guzzetti. Use of GIS technology in the prediction and monitoring of landslide hazard. *Natural Hazards*, pages 117–135, 1999.

[5] K. K. Chintalapudi, A. Dhariwal, R. Govindan, and G. Sukhatme. Ad-hoc localization using ranging and sectoring. In *Proceedings of the IEEE INFOCOM 2004*, pages 2662–2672, Nov. 2004.

[6] Crossbow Corporation. Stargate Gateway (SPB400). Available at: `http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/Stargate_Da%tasheet.pdf`, 2004.

[7] M. J. Crozier. *Landslides: causes, consequences & environment*. Croom Helm, 1986.

[8] E. Dietrich, R. Reisss, M. Hsu, and D. Montgomery. A process-based model for colluvial soil depth and shallow landsliding using digital elevation data. *Hydrological Processes*, pages 383–400, 1995.

[9] T. Fukuzono. Creep model of Kanto loam and its application to time prediction of landslide. *Landslides. (Eds: Chacon, J., Irigaray, C. and Fernandez, T.)*, pages 221–233, 1996.

[10] W. V. Gassen and D. Cruden. Momentum transfer and friction in the debris of rock avalanches. *Can. Geotech*, pages 623–628, 1989.

[11] P. Huber. Robust estimation of a location parameter. *Annals of Mathematical Statistics*, pages 73–101, 1964.

[12] J. N. Hutchinson. A sliding-consolidation model for flow slides. *Ca. Geotech*, pages 115–126, 1986.

[13] X. Lan, C. Zhou, L. Wang, H. Zhang, and R. H. Li. Landslide hazard spatial analysis and prediction using GIS in the Xiaojiang Watershed, Yunnan, China. *Eng. Geology*, pages 109–128, 2004.

[14] S. McDougall and O. Hungr. A model for the analysis of rapid landslide motion across three-dimensional terrain. *Can. Geotech*, pages 1084–1097, 2004.

[15] D. Moore, J. Leonard, D. Rus, and S. Teller. Robust distributed network localization with noisy range measurements. In *Proceedings of the ACM SenSys'04*, pages 50–61, 2004.

[16] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[17] K. Plarre, P. R. Kumar, and T. I. Seidman. Increasingly correct message passing algorithms for heat source detection in sensor networks. In *Proceedings of the First IEEE International Conference on Sensor ad Ad Hoc Networks (SECON 2004)*, 2004.

[18] A. N. Sheth, K. Tejaswi, P. Mehta, C. Parekh, R. Bansal, S. Merchant, T. N. Singh, U. B. Desai, C. A. Thekkath, and K. Toyama. Poster Abstract, A Sensor Network Based Landslide Prediction System. In *Proceedings of Sensys 2005*, Nov. 2005.

[19] C. Veder. *Landslides and Their Stabilization*. Springer-Verlag, 1996.

[20] O. Zienkiewicz and R. Taylor. *The finite element method*. MacGraw-Hill, 1989.