GPS on Every Roof, GPS Sensor Network for Post-seismic Building-wise Damage Identification

Kenji Oguni
and Tomohiro Miyazaki
Department of System Design Engineering
Keio University
Hiyoshi, Yokohama, Japan
Email: oguni@sd.keio.ac.jp

Masayuki Saeki
and Naoki Yurimoto
Department of Civil Engineering
Tokyo University of Science
Noda, Chiba, Japan
Email: saeki@rs.noda.tus.ac.jp

ABSTRACT

Development of wireless sensor network equipped with GPS for post-seismic building-wise damage identification is presented in this paper. This system is called *GPS on Every Roof*. Sensor node equipped with GPS antenna and receiver is installed on the top of the roof of each and every building. The position of this sensor node is measured before and after earthquake. The final goal of this system is to i) identify the displacement of the roof of each house and ii) collect the information of displacement of the roof of the houses through wireless communication. Superposing this information on GIS, building-wise damage distribution due to earthquake can be obtained. The system overview, hardware and some of the key components of the system such as on-board GPS relative positioning algorithm to achieve the accuracy in the order of several centimeters are described in detail. Also, the results from a field experiment using a wireless sensor network with 39 sensor nodes are presented.

Keywords: GPS, Wireless Sensor Network, Earthquake

1. INTRODUCTION

High risk of earthquake beneath large cities and/or disaster affecting wide area caused by a huge earthquake should be anticipated in the earthquake-prone regions. Once a huge earthquake hits and causes significant damage to the cities, information about the damage on the buildings in that area should be gathered quickly. However, as the scale and the severity of the disaster increases, the information about the damage caused by the earthquake will be deteriorated in quality, quantity and in its speed. Besides, significant damage on backbone communication network could be caused by an strong earthquake.

Under this condition, a wireless sensor network system covering the whole city could be a very prominent candidate for generating and gathering precise information about the damages caused by a huge earthquake. Especially, a brute force application to install a sensor node equipped with a GPS receiver on the top of the roof of each and every building in the whole city is the most straightforward and the easiest idea to come up with. Just by measuring the position of the GPS receiver before and after a huge earthquake, the displacement of the roof of each and every building can be obtained and this information can be gathered by wireless communication between sensor nodes. However, this straightforward application has been automatically dismissed because of the poor positioning accuracy of GPS with affordable cost. The typical positioning accuracy of affordable GPS used in the vehicle navigation is in the order of several meters[1]. Obviously, this is not enough for identifying the structural damages caused by an earthquake. At least, the positioning accuracy in the order of several centimeters is required for this application.

Motivated by the discussion above, a system called *GPS on Every Roof* has been developed by the authors. *GPS on Every Roof* is a wireless sensor network system consisting of sensor nodes equipped with an affordable L1-GPS receiver, a CPU with high computation power and a wireless communication module with relatively long communication distance. This sensor node is installed on the top of the roof of each and every building in the target area. The position of this sensor node is measured before and after earthquake. Then, the displacement of the roof caused by the earthquake can be obtained.

2. SCHEMATIC VIEW OF GPS ON EVERY ROOF

GPS on Every Roof is used for identification of damages on each and every building in the area affected by an earthquake. Sensor node shown in Figure 1 equipped with a GPS patch antenna, a GPS receiver, a CPU (indicated as SH2) and a wireless communication device is installed on the top of the roof of each and every building. The position of this sensor node (more precisely, the position of the GPS patch antenna) is measured before and after earthquake. Then, the displacement of the roof caused by the earthquake can be obtained. This information is sent to a main server through an autonomous wireless sensor network. Superposing this information on GIS, information about building-wise damage can be obtained together with the distribution of the damage. Also, possible

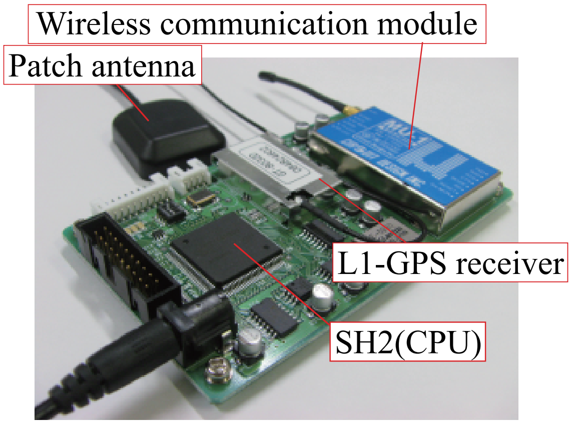


Fig. 1. Wireless GPS sensor node

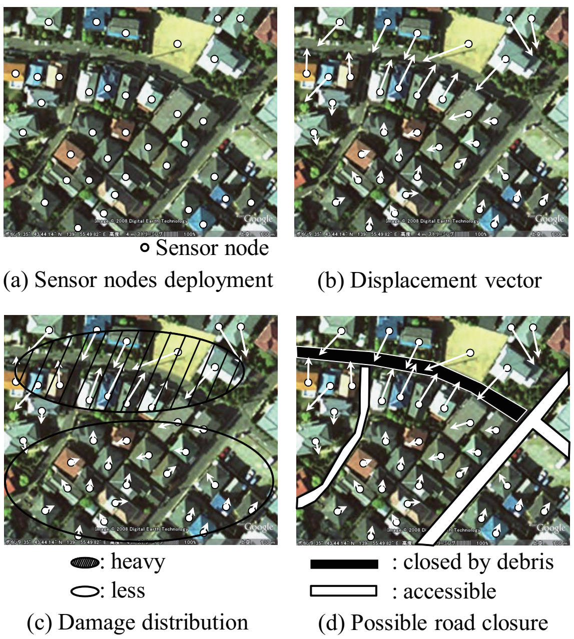


Fig. 2. Workflow of the system

road closure due to the debris from the collapsed houses can be estimated. This workflow is schematically shown in Figure 2.

3. IMPLEMENTATION

The accuracy of GPS positioning required for *GPS on Every Roof* is at least several centimeters. The positioning accuracy of GPS used in commercial car navigation system is at most several meters and this is not good enough for identifying the collapse of the buildings. On the other hand, GPS surveying with millimeter order positioning accuracy uses expensive hardware which cannot be spread on the top of the roof of each and every building in a city.

The algorithm for relative positioning with phase ranges has been implemented on the sensor node shown in Figure 1. This algorithm has been developed by the authors and will be discussed in detail in the next section. The positioning accuracy of several centimeters (several millimeters with high quality GPS observation data set) can be achieved by the on-board positioning analysis. This solves the dilemma between high positioning accuracy and low cost and this makes *GPS on Every Roof* possible.

In this section, the details of the design of the GPS observation data sharing, sensor node hardware and automatic allocation of sensor node ID are presented. The design philosophy behind these implementations is the suppression of the total amount of the wireless communication for reduction of time and for enhancement of the robustness of the system.

GPS obervation data sharing

To achieve the positioning accuracy of several centimeters using devices with affordable cost, the relative positioning with phase ranges [1] should be employed. The relative positioning with phase ranges uses the carrier phase of the signals from GPS satellites as the observation data. In this positioning algorithm, the relative position of a sensor node to the reference node is obtained. The relative position is computed based on the difference between the data observed on the sensor node under consideration and those of the reference node. This results in the requirement for the method to share the GPS observation data of the reference node.

Thus, we are left with the following two options. The one is the collection of all the GPS observation data from all the sensor nodes under local server and the positioning analysis is performed on the local server. The other is to spread the GPS observation data of the reference node to all the neighboring sensor nodes and on-board relative positioning analysis with phase ranges is performed on each and every node in the network.

Estimate of time for wireless communication for data sharing: GPS receiver on each sensor node receives the signals from GPS satellites and generates a data packet of 38 bytes in size every second. For 5 minutes GPS observation (typical duration of time for achieving the positioning accuracy of several centimeters using the GPS positioning method described in the next section), the total amount of GPS observation data on each sensor node results in 11.4 Kbytes. This data set should be sent to the CPU on which the GPS positioning analysis is performed.

The wireless communication device implemented on the sensor node requires T (msec) for transmitting a data packet as follows.

$$T = n \times 1.04 + 34 \qquad (1)$$

where, n is the size of data (bytes) in the packet. From Eqn. (1), the total amount of time for transmitting GPS observation data from one sensor node to the local server or to other sensor nodes is estimated as 22 seconds.

If the first option for data sharing (i.e., the collection of

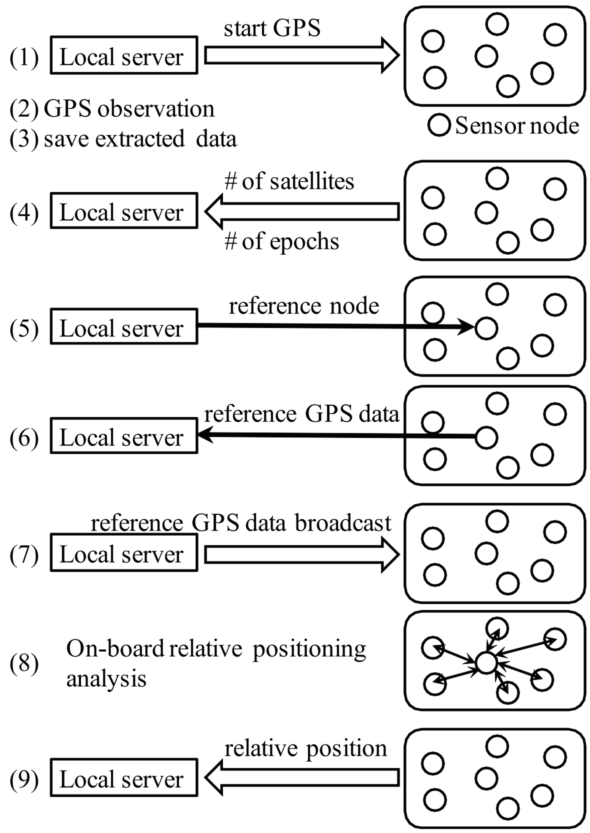


Fig. 3. Workflow of GPS positioning

all the GPS observation data from all the sensor nodes under local server and the positioning analysis is performed on the local server) were taken, the total amount of time for collecting the GPS observation data from (the maximum of 253) sensor nodes ends up with 93 minutes. This is too long and never satisfies the requirement for *GPS on Every Roof*.

Therefore, the second option (i.e., reference GPS data spread and on-board positioning analysis) has been employed in the proposed system. The only data sent to the local server from all the sensor nodes is the result of GPS positioning analysis (i.e., the x, y, z coordinates and an index for showing the reliability of the result). This results in the requirement for sending 1 data packet of 16 bytes in size from each sensor node and the total amount of time for 253 sensor nodes results in 13 seconds. Also, in the second option, the total number of wireless communication is suppressed. This makes the system scalable and robust.

Workflow of GPS positioning: Based on the discussion above, the reference GPS data is shared by the neighboring sensor nodes and on-board positioning analysis is performed in the proposed system. The workflow of GPS positioning is summarized in Figure 3.

(1) The local server broadcasts a command to the sensor nodes to start GPS observation.
(2) The sensor nodes respond the command and turn on the GPS receiver. Since the GPS receivers were turned off before the command, they have to search for the GPS satellites for the first 30 $\sim$ 60 seconds. This process is called a *cold start* and this makes the GPS observation time longer than that for a *hot start*.
(3) After the initialization process in the cold start, each sensor node observes the signals from GPS satellites and extracts the GPS data required for relative positioning with phase ranges. This extracted data set is saved on the on-board external RAM in the appropriate format.
(4) After GPS observation, each sensor node analyzes the observed data and evaluates the data quality. The number of observed GPS satellites and the number of observed epochs are the indicators of the data quality. These indicators are reported to the local server.
(5) The local server determines the reference sensor node based on the reported indicators of the data quality of each sensor node. Then the local server requests the reference sensor node to send back the whole GPS observation data to the local server.
(6) The reference sensor node responds to the command from the local server and reports its GPS observation data to the local server.
(7) The local server broadcasts the reference GPS data to all other sensor nodes.
(8) Each sensor node performs on-board relative positioning analysis using the reference GPS data and observed GPS data saved on its on-board external RAM.
(9) Each sensor node reports the analysis results (i.e., the x, y, z coordinates and an index showing the reliability of the result) to the local server.

This work flow requires the wireless communication of the whole GPS data only in the steps (5): from the reference node to the local server and (7): broadcast from the local server. Thus, with m reference nodes, the total number of the whole GPS data transmission is at most $2m$. This is a drastic reduction of the communication compared with the first option (to collect the GPS observation data from all the sensor nodes to the local server). This significant contribution for the reduction of time and for the enhancement of the robustness of the system is achieved by the implementation of on-board GPS relative positioning algorithm.

Sensor node hardware

A prototype sensor node shown in Figure 1 has been developed based on the discussion above. The major components of the current prototype sensor node to be described here are the CPU, the wireless communication module and the GPS receiver.

Since the on-board computation for GPS positioning requires double-precision arithmetic, a 32-bit RISC microcomputer, SH7144F (Renesas Technology Corp.) is employed as the main CPU on the sensor node. SH7144F works under 3.0V $\sim$ 3.6V input voltage with the highest clock frequency of 50MHz. The on-chip multiplier which can execute multiplication operations (32bits $\times$ 32bits $\rightarrow$ 64bits) in two to four cycles is the key feature of SH7144F to be employed as the main CPU on the sensor node. It also has 256KB of the internal ROM, 8KB of internal RAM and 4 asynchronous or clocked synchronous serial communication interfaces. Together with these internal devices, external RAM of 2MB is directly connected to SH7144F on the sensor board to keep and process the GPS observation data.

The wireless communication module on the sensor node is MU-1-1252 (CIRCUIT DESIGN, INC.). MU-1-1252 is a designated low power wireless communication module with the frequency band of 1252MHz. Sensor nodes with this module can be spread anywhere without any license or permission. In spite of its limited RF output power (10mW), the maximum distance of the wireless communication by MU-1-1252 is 600m with a line of sight. This distance will be reduced to 100m $\sim$ 200m when the line of sight is lost. This distance is long enough for the proposed application and this is the biggest reason for employing MU-1-1252 despite of its low communication speed (effective rate is 6800bps). The energy consumption is 60mA and 35mA for transmission and receiving, respectively with input voltage of 3.3V.

GPS receiver used in the prototype is a L1-GPS receiver GT8032 (FURUNO ELECTRIC CO., LTD.). Originally developed for GPS clock based time synchronization, GT8032 outputs the carrier phase in the format of FURUNO Binary in addition to the ordinary NMEA format GPS data. This data set of carrier phase is used in the GPS positioning proposed in this paper. In the proposed system, this GPS receiver is connected to a patch antenna used for car navigation for reduction of the cost.

This prototype sensor node works on a Li-ion battery with 3.6V output. This voltage is reduced to 3.3V on the sensor node. Although this prototype works on a battery (1900mAh@3.6V) for several hours without additional energy supply, reduction of energy consumption is preferable. For this purpose, energy hungry SH7144F should be used only for on-board computation of GPS positioning. Other tasks such as controlling the wireless communication unit and GPS observation should be taken care of by other CPU with lower energy consumption. In the next version of the sensor node, this energy management with dual CPU should be implemented and this is left for the future work.

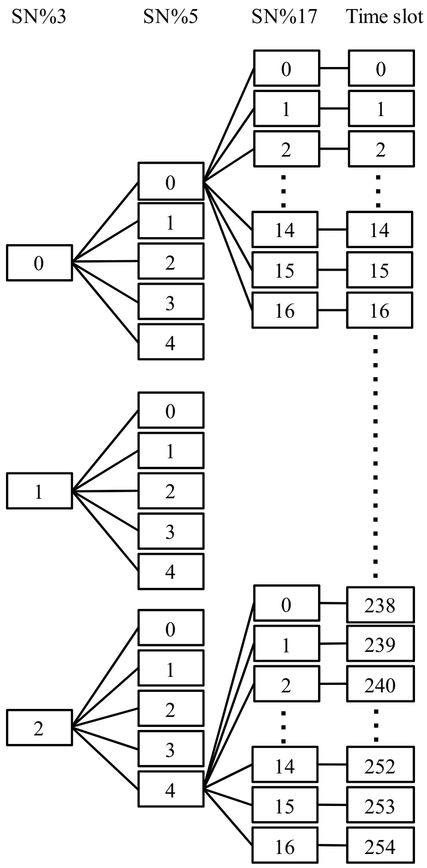
Automatic allocation of sensor node ID

Automatic allocation of sensor node ID is inevitable when large number of sensor nodes exist in the network. Sensor nodes are frequently replaced, removed or added. Especially in *GPS on Every Roof*, many sensor nodes could be physically damaged. Under this condition, management of the sensor node list in the network is a tedious task and this may be a potential source of a bug. Large scale sensor network should be a maintenance-free and deploy-and-work system. In this sense, automatic allocation of sensor node ID is one of the most important functionality for a sensor network with scalability. Particularly in *GPS on Every Roof*, the amount of the wireless communication and the total time for allocation of sensor node ID should be minimized for generating displacement data of each and every building in the whole area within 30 minutes.

Although there are many existing algorithms for automatic allocation of sensor node ID in the ad-hoc sensor network, a new algorithm customized for the wireless communication module (MU-1-1252) on the sensor node for *GPS on Every Roof* has been developed. The major features of the customized algorithm are its robustness and quickness.

For the assignment of ID, serial number of MU-1-1252 on each sensor node is used in *GPS on Every Roof*. Each sensor node reports its own serial number to the local server and the local server allocates unique ID to each sensor node. In this simple algorithm of ID allocation, the total time becomes a problem. Since MU-1-1252 avoids interfering the communication between other stations by checking RSSI (Received Signal Strength Indicator) before transmitting data packet, each sensor node should be given its own time slot with clean environment. The serial number of MU-1-1252 has 8 digits and one epoch of the wireless communication requires 70ms. If 70ms time slot is simply assigned for reporting its own serial number, time slots for the serial number 1 to 99,999,999 should be prepared. Then, the total time for completing ID allocation results in 81 days. The minimum number of time slots to be prepared is 253 (the maximum number of sensor nodes under a local server). Unnecessary idle time of the network in the simple assignment of the time slot is the major problem. Therefore, proper filtering of the serial number for assignment of the time slot is the key for solving this problem. For this purpose, a filter called *prime number filter* has been developed.

Prime number filter: The prime number filter is based on a set of three prime numbers and the remainder. An example is shown in Figure 4. This filter consists of three layers. In the first layer, the serial number of MU-1-1252 on a sensor node is divided by a prime number 3. Then, this serial number is categorized by the remainder. The same procedure is executed for the second layer with a prime number 5 and for the third layer with 17. Thus, a serial number is categorized in one of 255($=3\times5\times17$) time slots. Since the range of the serial number is 8 digits, many serial numbers are categorized in the same time slot. However, by applying a filter consisting of different set of prime numbers, the combination of the serial numbers categorized in the same time slot is changed. Thus, by applying 3 filters consisting of (3, 5, 17), (2, 7, 19) and (13, 23), serial numbers up to 7 digits (1 to 19,518,720) can be categorized in different time slots. If a prime number filter with a single prime number (11) is added to the abovementioned three filters, all 8 digits are fully categorized in different time slots. Introduction of this prime number filter drastically reduces the total time for ID assignment. ID assignment to the sensor nodes in a system consisting of 253 nodes can be completed in 1 minute.



(SN = serial number of MU-1-1252 on a sensor node)

Fig. 4. Example of prime number filter

4. GPS RELATIVE POSITIONING ALGORITHM WITH L1 CARRIER PHASE FOR ON-BOARD ANALYSIS

To make *GPS on Every Roof* possible, GPS relative positioning should have high accuracy (e.g., at least, in the order of several centimeters). Also, to enhance the robustness of the system, the communication of the whole GPS observation data should be avoided. GPS positioning analysis should be performed on-board (i.e., on each sensor node) and only the final result of the positioning analysis should be sent to the central server. The relative positioning algorithm presented here satisfies these requirements.

Double-Differenced carrier phases between nodes with short inter-node distance at time t, $\phi_{ij}^{kl}(t)$, can be modeled as

$$\lambda\phi_{ij}^{kl}(t) = \rho_{ij}^{kl}(\mathbf{x}, t) + \lambda N_{ij}^{kl} + e_{ij}^{kl}(t), \tag{2}$$

where $*_{ij}^{kl}$ are the double difference values for GPS satellites k, l and the sensor nodes i, j. $\rho_{ij}^{kl}(\mathbf{x}, t)$ is the double difference of the distances between satellites and the nodes, $\mathbf{x}$ is the position vector of a sensor node, λ is the wave length of the carrier, N_{ij}^{kl} is the double difference of the integer ambiguity and $e(t)$ is the noise. The assumption of the short inter-node distance allows us to approximate $e_{ij}^{kl}(t)$ as a sum of random noise and the noise due to antenna configuration [1].

Taylor expansion of Eqn. (2) with respect to $\mathbf{x}_0$ results in

$$U_{ij}^{kl}(t) = \partial_x\rho_{ij}^{kl}(\mathbf{x}_0, t)\Delta x + \partial_y\rho_{ij}^{kl}(\mathbf{x}_0, t)\Delta y + \partial_z\rho_{ij}^{kl}(\mathbf{x}_0, t)\Delta z + \lambda N_{ij}^{kl} + e_{ij}^{kl}(t) + h.o.t. \tag{3}$$

where, $U_{ij}^{kl}(t)$ is the corrected Double-Differenced carrier phases and can be computed as follows.

$$U_{ij}^{kl}(t) = \lambda\phi_{ij}^{kl}(t) - \rho_{ij}^{kl}(\mathbf{x}_0, t) \tag{4}$$

For different sets of satellites, Eqn. (3) with the corresponding coefficients can be obtained and these equations form the following simultaneous equations [2], [3], [4], [5].

$$\mathbf{U}(t) = \mathbf{A}(t)\Delta\mathbf{x} + \mathbf{N} + \mathbf{e}(t) \tag{5}$$

The unknowns in Eqn. (5) are $\Delta\mathbf{x} = (\Delta x, \Delta y, \Delta z)$ and a vector $\mathbf{N}$ which has N_{ij}^{kl} as components.

To solve Eqn. (5) in a robust manner, linear approximation of $U_{ij}^{kl}(t)$ with respect to time t is introduced. This approximation safely holds for small t (i.e., short GPS observation time). GPS observation time in the proposed system is 5 minutes and this is considered to be short enough for linear approximation and long enough for keeping independence of the GPS observation data. In this linear approximation, only the first and the last epoch in the observation are used to form linearized observation equations corresponding to Eqn. (3). As a result, with little amount of data communication, on-board GPS positioning analysis with the accuracy of a few centimeters (with better condition, accuracy of a few milimeters) can be achieved.

5. FIELD EXPERIMENT

The on-board GPS positioning algorithm and other fundamental algorithms for wireless sensor network such as automatic ID allocation, reference GPS data broadcast, retrieval of lost packets and communication timing control are implemented on the sensor nodes and the local server. Then, a field experiment with 39 sensor nodes was performed.

The sensor nodes were deployed in grid as shown in Figure 5 (indicated as "before $+$"). The numbers in Figure 5 are the IDs of the sensor nodes. To simulate the displacement due to an earthquake, the sensor nodes 479, 459, 397 and 396 were displaced by about 50m, 2m, 1m and 3m, respectively. Then, relative positioning was performed again. Sensor nodes after displacement are indicated as "after $\circ$." The arrows are the displacement vectors. The arrangement of $'\circ'$ after displacement shows that the locations of all the sensor nodes (including the nodes 376, 475 and 481) were precisely identified. It seems as if the sensor node 481 was displaced and the nodes 376 and 475 suddenly appeared out of the blue. But this is because of the wrong estimation before displacement for the nodes 376, 475 and 481. The vectors for these nodes are the dummy displacement vectors. Other displacement vectors correspond to the actual displacement of the displaced sensor

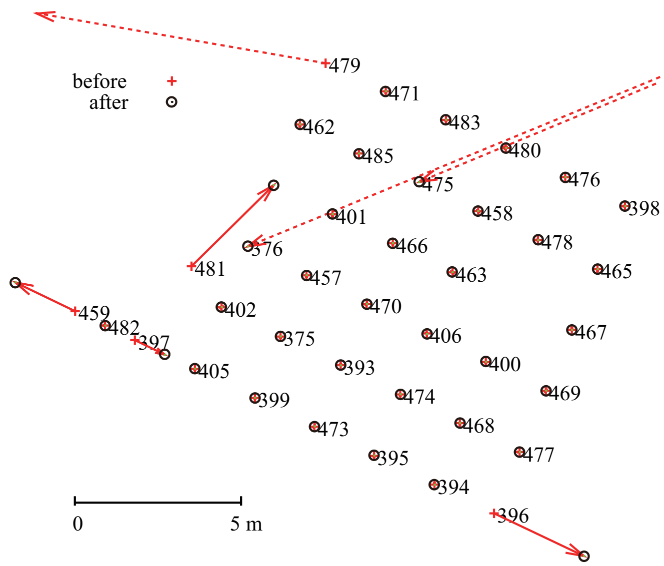


Fig. 5. Estimated sensor node position before and after displacement

nodes (479, 459, 397 and 396) and these displacements were precisely estimated within the accuracy of a few centimeters.

6. CONCLUSIONS AND FUTURE WORKS

A wireless sensor network system called *GPS on Every Roof* has been proposed in this paper. This system is a wireless GPS sensor network for post-seismic buildinf-wise damage identification. The system overview, hardware and on-board GPS positioning algorithm are discussed together with results from a field experiment.

The future work regarding *GPS on Every Roof* are as follows.

- Development of sensor node with better (longer distance, higher communication speed) wireless communication module
- Field experiment with sensor nodes deployed on the top of the roof of the buildings and with longer duration of time
- Development of sensor node with dual CPUs (one low-power, low-specification CPU and one high-power, high-specification CPU) for reduction of energy consumption

REFERENCES

[1] Hofmann, B., Wellenhof, H. Lichtenegger and J. Collins (2001), "GPS, Theory and Practice", *Springer Wien NewYork*.

[2] Kim, D. and Langley, R.B. (2000), "GPS Ambiguity Resolution and Validation: Methodologies, Trends and Issues", *7th GNSS Workshop, Int. Symp. GPS/GNSS*, Seoul, Korea, Nov.

[3] Teunissen, P. J. G. (1993), "Least-squares estimation of the integer GPS ambiguities", Invited lecture, Section IV Theory and Methodology, IAG General meeting, Beijing, China.

[4] Teunissen, P. J. G., P. J. de Jonge, C. C. J. M. Tiberius (1997), "The least-squares ambiguity decorrelation adjustment: its performance on short GPS baselines and short observation spans." *Journal of Geodesy*, 71, 589-602.

[5] Saeki, M., Kosaka, T. and Kaneko, S. (2006), "Phase ambiguity resolution based on linear modeling of DD carrier phase", *Proc., ION GNSS 2006*