Experimental Analyses of Terrain Factors to Performance of Sensor Placement Optimization in 3D Environments

Dang Thanh Hai, Nguyen Thi Tam, Le Hoang Son, and Vinh Trong Le

***Abstract*—The Sensor Placement Optimization (SPO) in 3D environments aims to find an optimal solution of putting sensors of a Wireless Sensor Network on a 3D terrain such as Digital Elevation Model (DEM). This is an optimization problem involving network infrastructures and terrain factors. An improvement of Particle Swarm Optimization (PSO) has been presented to deal with this matter. It achieved satisfactory results in comparison with the relevant ones. A concerning issue that was not be totally settled in the PSO algorithm is the impact of terrain factors to the performance of the algorithm. It is obvious that the performance of the planning algorithm would change according to various types of terrains having different morphologies and distributions of events. In this paper, we perform an experimental analysis for such the research question. Specifically, we aim to investigate i) Which types of terrains would make the PSO algorithm achieve the best performance? ii) How will various distributions of events affect the performance? iii) Which are the most appropriate values of terrain parameters that should be opted for the algorithm in order to get desired performance? Those analyses would help practitioners make usage of the algorithm in real world applications.**

***Index Terms*—3D terrains, particle swarm optimization, sensor placement optimization, terrain factor, wireless sensor network.**

I. Introduction

The Sensor Placement Optimization in 3D environments problem and its variants have been widely investigated [1]-[3]. A major challenge of this problem in comparison with those in 2D environments is the capability to handle the complex of 3D structures in a terrain; thus making the planning harder than that in 2D areas. A terrain has many environmental factors namely blocks of high-rise buildings, hills, rivers, lakes, etc. A planning algorithm should take into account those factors to the calculation of global coverage. It is necessary to design such the algorithm in order to have good network planning solutions which are then used for decision-making processes.

In our previous work [4], we have proposed a wireless sensor network (WSN) model on a 3D terrain that considers structures, sensor angles, obstacles, holes in their real forms instead of flat planes as in the existing 2D models. Coverage capabilities of sensors are measured within the use of Line-of-Sight function, which validates whether a sensor can observe an event or not. Physical holes, which are restricted regions in a terrain like ponds, lakes, etc., are automatically detected and attached to the model's parameters [5]. A Particle Swarm Optimization (PSO) based planning algorithm is used to determine optimal locations of sensors that maximize global coverage to all events. The achieved planning solutions are useful for practitioners to make effective network planning on a given terrain.

The remain question of our previous researches is to determine the impact of terrain factors to performance of the planning algorithm. That is to say, how the performance will be when either terrain morphology, sensor distribution or parameters of the planning algorithm vary. Is there any method to determine a possible range of performance of the algorithm for a given terrain? Those questions raise the motivation for this paper.

In this paper, we perform an experimental analysis for such the research question. Specifically, we aim to investigate: 1) which types of terrains would make the PSO algorithm achieve the best performance? 2) how will various distributions of events affect the performance? 3) which are the most appropriate values of terrain parameters that should be opted for the algorithm in order to get desired performance? Those analyses would help practitioners make usage of the algorithm in real world applications.

The rest of the paper is organized as follows. Section II presents an overview of our previous works regarding the WSN model on a 3D terrain (a.k.a. 3D sensing model) and the PSO-based planning algorithm. The experimental results and discussion will be presented in Section III. Finally, conclusions are given in Section IV.

II. Pleminary

A. 3D Sensing Model

Given a 3D terrain with H physical holes and a WSN consisting of N sensors. Our aim is to determine locations of those sensors on the terrain that maximize global coverage to M events. The 3D sensing model for this problem is demonstrated below [4].

- *T* is a Digital Elevation Model (DEM) terrain, which is a matrix whose values representing for elevations of grid points. Some parameters are:

  a) *cellsize*: the size of grid cell;
  b) *nrows* and *ncols*: the number of rows and columns of DEM respectively;

Manuscript received September 10, 2015; revised November 10, 2015.

Dang Thanh Hai is with Da Lat University, Vietnam (e-mail: haidt@dlu.edu.vn).

Nguyen Thi Tam, Le Hoang Son, and Vinh Trong Le are with VNU University of Science, Vietnam National University, Vietnam (e-mail: tamnt@vnu.edu.vn, sonlh@vnu.edu.vn, vinhlt@vnu.edu.vn).

DOI: 10.7763/IJET.2016.V8.920

c) $h(x_i, y_i)$: values representing for the elevations of grid points.

- $WSN=\{s_1,s_2,...,s_N\}$ is a sensor network where,

$$s_j = \{x_j^s, y_j^s, h_j^s(x_j^s, y_j^s), \alpha_j, \theta_j, \xi_j, \beta_j\}, \forall j \in [1,2,...,N] \quad (1)$$

a) (x_j^s, y_j^s) is the coordinate of s_j in Oxy ;

b) $h_j^s(x_j^s, y_j^s)$ is the heigh of s_j in position (x_j^s, y_j^s);

c) r_j^s is the sensing radius of s_j;

d) θ_j is the pan angle of s_j around the vertical axis (X direction);

e) α_j is the angle to define the orientation of the directional sensor s_j around X direction, $0 \le \alpha_j \le 2\pi$;

f) ξ_j is the tilt angle s_j around the horizontal axis (Z direction);

g) β_j is the angle to define the orientation of the directional sensor s_j around Z direction, $0 \le \beta_j \le 2\pi$.

- $R=\{(x_1, y_1), (x_2, y_2),...,(x_H, y_H)\}$ is a set of physical holes [5].
- $E= \{e_1, e_2,...,e_M\}$ is a set of events,

$$e_i = \{(x_i^e, y_i^e), h_i^e(x_i^e, y_i^e), \omega_i\}, \forall i \in [1,2,...,M] \quad (2)$$

h) M is the number of grid points which is not in physical holes;

i) (x_i^s, y_i^s) is coordinate of point e_i in Oxy which is a grid point;

j) ω_i is the weight of e_i.

- A point e_i is said to be covered by sensor s_i if and only if the following conditions are satisfied:
  a) The Euclidean distance between the location of sensor s_j and point e_i less than or equal sensing radius of s_j;
  b) The angle between the sensor s_j and point e_i along the X direction less than or equal the pan angle of s_j;
  c) The angle between the sensor s_j and point e_i along the Z direction less than or equal the tilt angle of s_j;
  d) Visibility from the sensor s_j to point e_i.

Therefore, the sensing model mainly depends on distance, orientation, and visibility.

a) μ_d is the binary function to measure the distance between s_j and e_i:

$$\mu_d = \begin{cases} 1, & d(s_j, e_i) \le r_s^j \\ 0, & otherwise \end{cases} \quad (3)$$

$$d(s_j, e_i) = \left\| (x_i^e, y_i^e, h_i^e) - (x_j^s, y_j^s, h_j^s) \right\|;$$

b) μ_p is the binary function to measure the coverage capabilities of sensor s_j to the point e_i by angle of the sensors along vertical axis:

$$\mu_p = \begin{cases} 1, & \arctan(\frac{y_i^e - y_j^s}{x_i^e - x_j^s}) \in [\alpha_j, \alpha_j + \theta_j] \\ 0, & otherwise \end{cases} \quad (4)$$

where $\arctan(\frac{y_i^e - y_j^s}{x_i^e - x_j^s})$ is the angle between the sensor s_j and the point e_i along the X direction

c) μ_t is the binary function to measure the coverage capabilities of sensor s_j to the point e_i by angle of the sensors along horizontal axis:

$$\mu_t = \begin{cases} 1, & \arctan(\frac{h_i^e - h_j^s}{d(s_j, e_i)}) \in [\xi_j, \beta_j + \xi_j] \\ 0, & otherwise \end{cases} \quad (5)$$

where $\arctan(\frac{h_i^e - h_j^s}{d(s_j - e_i)})$ is the angle between the sensor s_j and the point e_i along the Z direction

d) v_{ij} represent visibility between s_j and e_i:

$$v_{ij} = \begin{cases} 1, & \mu_d = 0 \quad or \quad \mu_t = 0 \quad or \quad \mu_d = 0 \\ \frac{1}{1 + num\_Obstacles(s_j, e_i)}, & otherwise \end{cases} \quad (6)$$

where $num\_Obstacles(s_j, e_i)$ is the number of abstacles between sensor s_j and point e_i, it is determined by LoS method.

- The coverage $C(s_j, e_i)$ of s_j at point e_i can be defined as functions of distance μ_d, pan angle μ_p, tilt angle μ_t and visibility v_{ij} from sensor:

$$C(s_j, e_i) = \mu_d \times \mu_t \times \mu_p \times v_{ij} \quad (7)$$

- The probability of the environment that covers point e_i is:

$$C_{e_i}(WSN, e_i) = 1 - \coprod_{i=1,N}(1 - C(s_i, q)) \quad (8)$$

- The global coverage:

$$C_g(WSN, E) = \frac{\sum_{e_i \in E} \omega_i C_{e_i}(WSN, e_i)}{M} \quad (9)$$

- The objective function of the problem is:

$$C_g(WSN, E) \to \max \quad (10)$$

- Constraints:

$$(x_j^s, y_j^s) \notin R, \forall j \in [1,2,...,N] \quad (11)$$

B. PSO-Based Planning Algorithm

PSO [6] is a population based stochastic optimization technique inspired by social behavior of bird flocking or fish schooling. Combining with the virtual forces algorithm [7], the authors [4] proposed a planning method to derive optimal solutions of the 3D sensing model. Some notations used in the algorithm are:

- $X_i=(X_{1i}, X_{2i},\dots,X_{Ni})$ and $Vi=(V_{1i},V_{2i},\dots,V_{Ni})$ represent for position and velocity p_i where $X_{ji}=(x_{ji},y_{ji})$ and $V_{ji}=(v_{ji}, v'_{ji})$ represent for the position and the velocity of sensor s_j in particle $p_i \forall j=\overline{1,N}$.
- $pbest_i=\{pbest_{1i}, pbest_{2i},\dots,pbest_{Ni}\}$ denotes the best particle of particle i^{th}, where $pbest_{ji}$ is the best position of sensor s_j in particle p_i.
- $gbest_i=\{pbest_1, gbest_2,\dots,gbest_N\}$ is the best particle in the swarm, where $pbest_j$ is the best position of sensors s_j in history of the swarm.
- $f(gbest)$ is the best fitness value of the swarm.

Details of this algorithm are shown below:

Step 1: (Initialization). The beginning population is initiated with Npop particles, where Npop is a designated parameter. Each particle is randomly initiated their position and velocity.

Step 2: Calculate fitness values of all particles. The procedure is shown as in Table I.

TABLE I: CALCULATE FITNESS

Fitness (f)
For i = 0 to M
For j = 0 to N
Calculate μ_d by equation (3)
Calculate μ_p by equation (4)
Calculate μ_t by equation (5)
Calculate v_{ij} by equation (6)
$C(sj, ei) \leftarrow \mu_d \times \mu_p \times \mu_t \times v_{ij}$
$C_{e_i}(WSN,e_i) \leftarrow 1-\prod_{i=1,N}(1-C(s_i,q))$
$C_g(WSN,e_i) \leftarrow C_g(WSN,E)+C_{e_i}(WSN,e_i)$
return $\frac{C_g(WSN,E)}{M}$

Step 3: Update pbest and gbest. The procedure is shown as in Table II.

TABLE II: UPDATE PBEST AND GBEST

Update pbest process	*Update gbest process*
For i=1 to N_{pop}	**For** i=1 to N_{pop}
If $f(p_i) > f(pbest_i)$	**If** $f(gbset) < f(pbest_i)$
$f(pbest_i) \leftarrow f(p_i)$	$f(gbset) \leftarrow f(pbest_i)$
For j=1 to N	**For** j=1 to N
$pbest_{ji} \leftarrow X_{ji}$	$gbest_j \leftarrow pbest_{ji}$
End For	**End For**

TABLE III: UPDATE THE VELOCITIES AND POSITION OF PARTICLES

Update
For i=1 to N_{pop}
For j=1 to N
$V_{ij} = \omega \times V_{ij} + r_1 \times c_1 \times (pbest_{ij}-X_{ij}) + r_2 \times c_2 \times (gbest_j-X_{ij}) + r_3 \times c_3 \times F_j$
$X_{ij} = \begin{cases} X_{ij}+V_{ij} & if \quad X_{ij}+V_{ij} \notin R \\ X_{ij} & otherwise \end{cases}$
End For
End For

Step 4: Update the velocities and positions of particles by virtual forces. The procedure is shown in Table III.

- $d(s_i, s_j)$ is the Euclidean distance between sensors.
- $adj(s_i)$ is the adjacency set of sensor s_i. sensor s_j is called adjacency of s_i sensor if and only if $d(s_i, s_j) \le r_c$, where r_c is communication radius, $r_c = 2 \times r_s$.
- F_{ij} is the virtual force exterted by the neighborhood s_j on s_i.
- F_i is the total virtual force action on sensor s_i.
- d_{ave} is the average distance between two sensors when they are evenly distributed in the area,

The virtual force function is calculated below,

$$F_{ij} = \begin{cases} \frac{d_{ave}-d_{ij}}{2\times d_{ij}}(x_i-x_j, y_i-y_j) & if \quad d_{ave} > d_{ij} \\ -\frac{d_{ave}-d_{ij}}{2\times d_{ij}}(x_i-x_j, y_i-y_j) & if \quad d_{ave} < d_{ij} \\ 0 & if \quad d_{ave} = d_{ij} \end{cases} \tag{12}$$

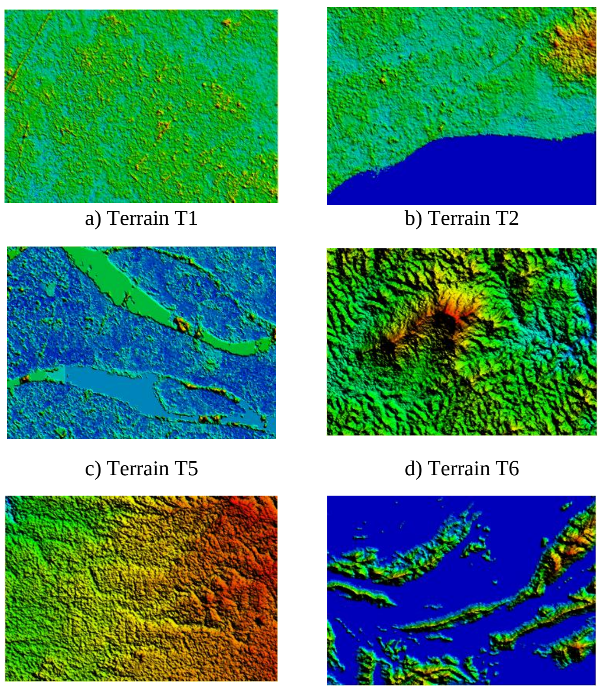
$$F_j = \sum_{s_i \in adj(s_j)} F_{ij} \tag{13}$$

Step 5: Repeat the whole process from Step 2 to Step 4 until the maximal interation step (PSO_MaxIter) is reached.

III. EXPERIMENTAL ANALYSIS

A. Data Description

Our experiments were implemented on DEM terrains of Vietnam that were collected by EarthExplorer software [8]. They have various morphologies with size being 200 x 250 and cell sizes being 25 meters. Fig. 1 illustrates some terrains. Table IV gives a brief summary of morphologies of these terrains.



a) Terrain T1 b) Terrain T2 c) Terrain T5 d) Terrain T6 e) Terrain T7 f) Terrain T10

Fig. 1. Terrains.

TABLE IV: BRIEF SUMMARY OF MORPHOLOGIES OF TERRAINS

Terrain	Morphology
T1	City region has many diverse buildings; no hills and rivers.
T2	City region has many buildings with medium height; low altitude hills; partial sea.
T3	The island has low hills with different altitudes, surrounded by the sea.
T4	Plain region with few buildings, rivers and canals, no hills.
T5	Plain region with few buildings, many rivers and canals, no hills.
T6	Highland region has few buildings, high number of mountains and hills.
T7	Highland region with high mountains, hills elevations ascending.
T8	City region has few buildings, with high number of mountains bordering the sea.
T9	City region has many buildings and uniformly distributed ponds.
T10	Coastal has many small islands of varying heights.

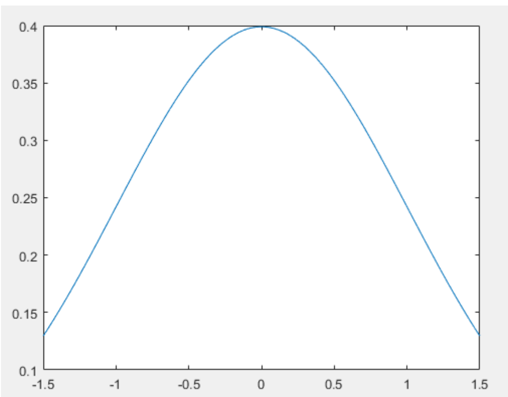
B. *Parameter Setting*

- *In the first experiment*, we aim to validate performance of the planning algorithm according to various terrains. Thus, other parameters are set as fixed values. The number of events is equal to 5% of total grid points of the terrain. The number of sensors are equal to 25% of the number of events. Each sensor is assumed to be a disk with radius of 50m, the tilt angle is 90 degree, and the pan angle is 180 degree. Table V. describes input parameters for this experiment.

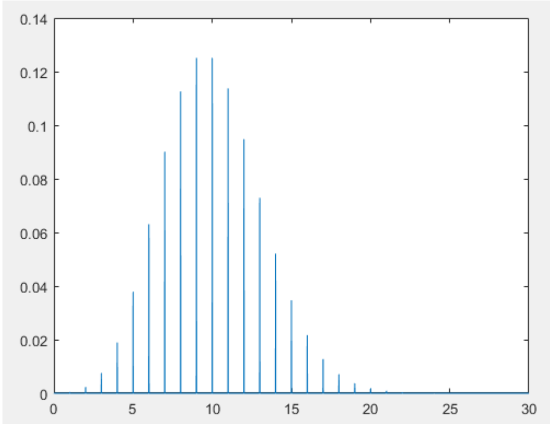
TABLE V: DESCRIPTION THE INPUT PARAMETERS OF TERRAINS

Terrain	Number of events	Number of holes	Number of sensors
T1	1206	25871	301
T2	998	30040	249
T3	1026	29465	256
T4	1879	12401	469
T5	731	35373	182
T6	2402	1950	600
T7	2286	4275	571
T8	2304	3911	576
T9	1704	15915	426
T10	919	31620	229

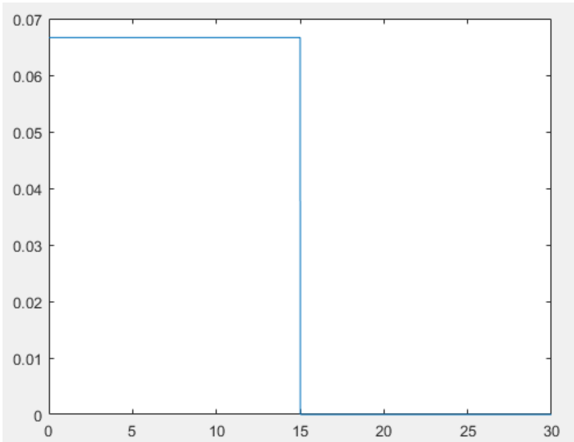
- *In the second experiment*, we aim to check how various distributions of events affect performance of the algorithm. In this test, 5 distributions namely Gaussian, Poisson, Uniform, Gamma and Beta are used (Fig. 2).



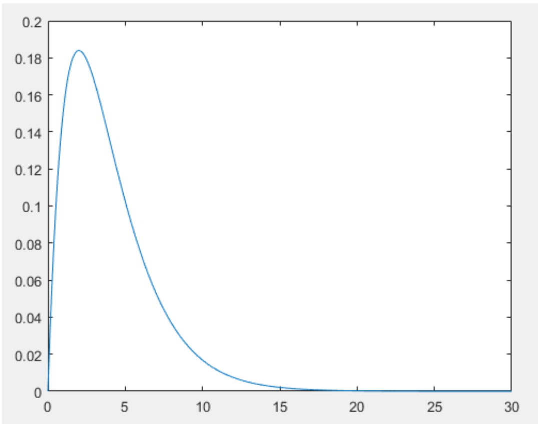
a) Gaussian distribution



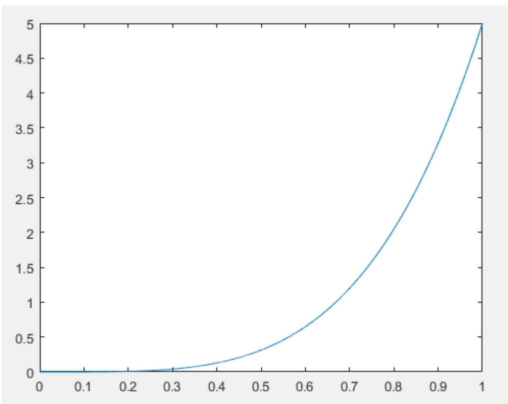
b) Poison distribution



c) Uniform distribution



d) Gamma distribution



e) Beta distribution

Fig. 2. Distributions of events.

- *In the third experiment*, we aim to check impact of different values of C_1, C_2, C_3 in Table III to the performance. We examine some cases as follows: C1=2, C2=2, C3=1; C1=2, C2=2, C3=2 ; C1=1, C2=2, C3=2; and C2=2, C2=1, C=2.
- *In the last experiment*, we aim to check impact of various number of events namely 1%, 2%, 3%, and 4% of total grid points of the terrain to the performance.

C. Experimental Results

TABLE VI: COVERAGE VALUES OF THE PLANNING ALGORITHM

Terrain	Min Value	Max Value	Average Value	
T1	0.228891	0.265247	0.244538	
T2	0.590864	0.628873	0.605027	
T3	0.338342	0.426738	0.391295	
T4	0.295525	0.340363	0.324394	
T5	0.188031	0.210779	**0.195810**	*
T6	0.362412	0.373797	0.366907	
T7	0.360370	0.351660	0.357110	
T8	0.344343	0.386764	0.371549	
T9	0.280303	0.329146	0.302650	
T10	0.622013	0.656260	**0.645131**	**

* : Value indicates the worse in a column
**: Value indicates the best in a column

Table VI shows coverage values of the algorithm on 10 terrains represented in maximal, minimal and average values. It is obvious that performances of the algorithm in most of terrains are nearly identical. The best and worst terrains in term of performance are T10 and T5, respectively.

TABLE VII: COVERAGE VALUES BY DIFFERENT DISTRIBUTIONS

Terrain	Gaussian	Poisson	Uniform	Gamma	Beta
T1	0.386170	0.259292	0.265125	0.240101	0.190317
T2	0.430054	0.822218	0.530167	0.752624	0.843394
T3	0.415834	0.758162	0.801171	0.686666	0.704594
T4	0.431134	0.474998	0.347384	0.347384	0.324372
T5	0.359759	0.327509	0.273244	0.273244	0.185137
T6	0.438247	0.480433	0.390034	0.323516	0.360818
T7	0.474147	0.491770	0.401771	0.294031	0.321172
T8	0.507227	0.413100	0.426098	0.340400	0.485997
T9	0.429488	0.438889	0.341801	0.236682	0.239807
T10	0.489270	0.627169	0.506267	0.316434	0.517716
Average	0.436133	**0.509354**	0.428306	**0.381108**	0.417332
		**		*	

* : Value indicates the worse in a column
**: Value indicates the best in a column

TABLE VIII: COVERAGE VALUES BY DIFFERENT PARAMETERS OF PSO

Terrain	C1=2 C2=2 C3=1	C1=2 C2=2 C3=2	C1=1 C2=2 C3=2	C1=2 C2=1 C3=2
T1	0.386170	0.421415	0.443077	0.365518
T2	0.430054	0.420420	0.427584	0.373082
T3	0.415834	0.454275	0.418320	0.357536
T4	0.431134	0.454505	0.438380	0.410934
T5	0.359759	0.363359	0.354259	0.324321
T6	0.438247	0.531827	0.531827	0.385366
T7	0.474147	0.483313	0.483313	0.383494
T8	0.507227	0.490258	0.490258	0.446612
T9	0.429488	0.424910	0.426542	0.466463
T10	0.489270	0.509721	0.509721	0.412479
Average	0.436133	**0.45540**	0.452328	0.392581
		**		*

* : Value indicates the worse in a column
**: Value indicates the best in a column

Table VII shows coverage values of the algorithm by differerent distributions of events. It is obvious that Poisson and Gamma are the most and worst stable distribution respectively.

Table VIII shows coverage values of the algorithm by differerent parameter valus of PSO. The triple (C1 =2, C2=2, C3=2) is the best parameters whilst (C1=2, C2=1, C3=2) is the worst one.

Table IX shows coverage values of the algorithm by differerent proportions of events. It has been shown that 5% is the most ideal number of events that maximaize the coverage. On the contrary, 4% shows the worst results.

TABLE IX: COVERAGE VALUES BY DIFFERENT PROPORTIONS OF EVENTS

Terrain	**1%**	**2%**	**3%**	**4%**	**5%**
T1	0.357823	0.409585	0.364930	0.337227	0.386170
T2	0.310518	0.367759	0.341214	0.305657	0.430054
T3	0.380782	0.375361	0.355816	0.354816	0.415834
T4	0.403294	0.385856	0.359714	0.373458	0.431134
T5	0.254427	0.261202	0.317951	0.281454	0.359759
T6	0.382655	0.388907	0.378387	0.386097	0.438247
T7	0.382109	0.378631	0.384302	0.386165	0.474147
T8	0.455415	0.443898	0.441130	0.446682	0.507227
T9	0.354942	0.420348	0.425138	0.346300	0.429488
T10	0.377102	0.422437	0.451212	0.356970	0.489270
Average	365907	0.385398	0.381979	0.357483	**0.436133**
				*	**

* : Value indicates the worse in a column
**: Value indicates the best in a column

D. Discussion

According to the results obtained from experiments, the following remarks and notes are shown below.

TABLE X: COMPARISON OF TERRAINS IN FIGURE 1

Terrain	Events	Sensors	pCoverage of random dist	pCoverage of Gaussian dist
T1	1206	301	0.244538	0.386170
T2	998	249	0.605027	0.359759
T5	731	182	0.195810	0.359759
T6	2402	600	0.366907	0.438247
T7	2286	571	0.357110	0.474147
T10	919	229	0.645131	0.489270

It has been shown in *the first experiment* that coverage values of the algorithm by different terrain morphology are similar. Specifically, T2 and T10 have many islands and holes so that their coverages are much higher than those of other terrains. In Fig. 1 (c), we recognize that T5 interspersed with numerous canals dense. The numbers of event and sensors for T5 are 731 and 182, respectively. The covergage value of T5 is the worst among all since this terrain has too many holes with the number and radius of sensors being not large enough to cover the events.

As summarized in Table X, the coverage value of T1 is the second lowest among all due to its morphology is a city with many high buildings and physical holes so that sensor signal could not cover the events. T5 and T6 contain many hills with ascending elevations and in the same direction so that their coverage values are similar and approximate to the mean of all terrain.

The second experiment shows that terrains in Table X have similar coverage values with Gaussian distribution, but using Poisson distribution for those terrains is better. Especially, only coverage values of T2 and T10 decrease with Gaussian distribution since sensors focus on terrain's center which is not the central position of events. The Poison distribution is even efficient with terrains having many holes. On the other hand, compared to the first experiment, Gaussian distribution has higher coverage value than random one. T2 and T3 have high coverage values with Poison, Uniform, Gamma and Beta distributions. Fig. 3. shows the coverage values of the algorithm by distributions.

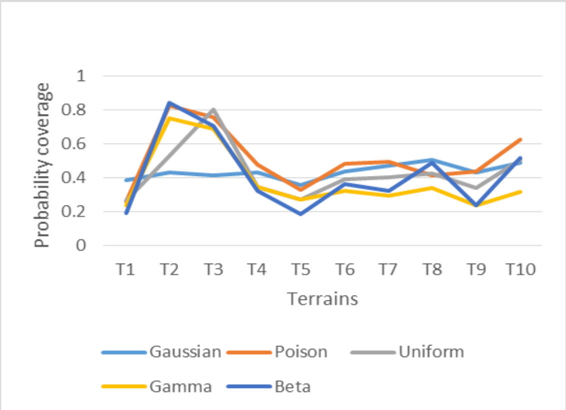


Fig. 3. Coverage by distributions.

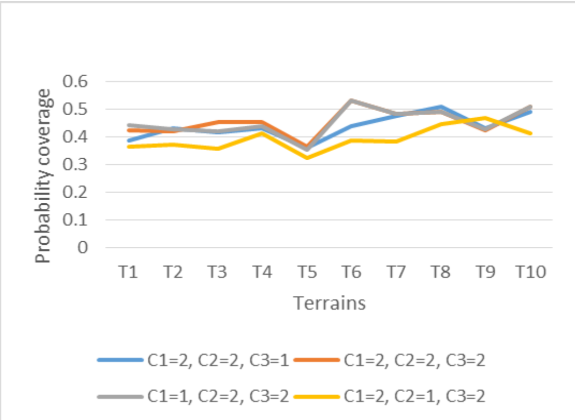


Fig. 4. Coverage by parameters of PSO.

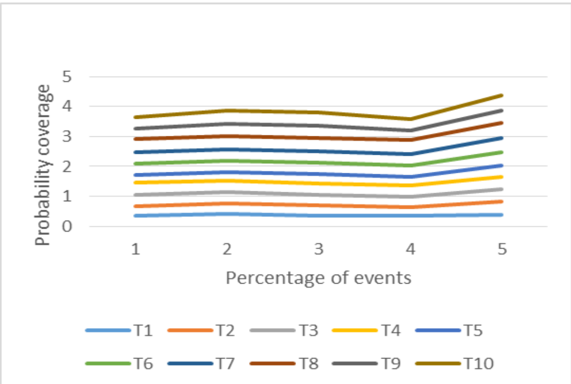


Fig. 5. Coverage by proportions of events.

In *the 3rd experiment*, we recognize that the triple (C1=2, C2=2, C3=2) achieve highest coverage value of the algorithm among all. The coverage in this triple increases 4.41% compared to that of the previous experiments. In order to enhance the coverage, this parameter triple and Poison distribution should be used. Another comment is that when changing the parameters, the changing rate is 16% of coverage. Changing values of (C1, C2, C3) does not affect the performance. For this parameter triple, T5 has smallest coverage value since it has a vast number of concentrated physical holes. T10 has highest coverage since the events are evenly distributed and continuous. There are seven terrains having coverage near the average value. This confirms the stability of this parameter for all kinds of terrains. Fig. 4 shows the curve parameter (C1=2, C2=2, C3=2) located above the highest.

The last experiment shows how coverage value changes when proportions of events varies. It is clear that coverage value with 4% events decreases compared to that with 1%. An interesting fact in Fig. 5 is that when the proportion increases from 1% to 5%, T1 to T10 has more varied coverage amplitude. The increasing of coverage from T1 to T5 is relatively stable.

IV. Conclusions

In this paper we performed an experimental analysis of terrain factors to the performance of the planning algorithm on 3D terrains. The findings of this paper are list below:

1) Terrain morphologies as in T2 and T10, which have events and physical holes evenly distributed, are the ideal environment to deploy the planning algorithm. On the other hand, terrain like T5 having many physical holes at the event alternating with high oscillations is the worst environment. In this case, we must increase the number of events and sensors in oder to get satisfactory results. Poison is the most suitable distribution of events.
2) The triple (C1=2, C2=2, C3=2) should be used.
3) Increasing the number of events means better coverage value, but this number should not be large to avoid computational complexity. Ideally, 5% is the most ideal number of events.
4) Source codes and datasets of this research are available at[1]. Users can download them for their own purposes.

This research suggests some further works such as handling computational complexity of the planning algorithm; solving the clustering of WSN 3D problem.

References

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292-2330, 2008.

[2] S. S. Dhillon and K. Chakrabarty, "Sensor placement for effective coverage and surveillance in distributed sensor networks," vol. 3, pp. 1609-1614, 2003.

[3] C. F. Huang, C. Y. Tseng, and L. C. Lo, "The coverage problem in three-dimensional wireless sensor networks," *Journal of Interconnection Networks*, vol. 8, no. 3, pp. 209-227, 2007.

[4] N. T. Tam, D. T. Hai, L. H. Son, and L. T. Vinh, "Optimization for the sensor placement problem in 3D environments," in *Proc. 2015 IEEE 12th International Conference on Networking, Sensing and Control Howard Civil Service International House*, Taipei, Taiwan, April 9-11, 2015.

[5] D. T. Hai, P. H. Thong, T. T. T. Giang, L. H. Son, and L. T. Vinh, "A novel multi-objective optimization model and an efficient physical holes detection method for the sensor placement problem in 3D terrains," *in Proc. @ 17th National Conference on Some Selected Issues of Information and Communications Technology -Dak Lak*, Viet Nam, 2014.

[6] R. C. Eberhart and J. Kennedy, "A new optimizer using particles swarm theory," in *Proc. The Sixth International Symposium on Micro Machine and Human Science*, pp. 39-43, 1995.

[1] https://sourceforge.net/p/wsncoverage/code/ref/master/

[7] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh, *IEEE Internet Computing on Deploying a Wireless Sensor Network on An Active Volcano*, vol. 10, no. 2, pp. 18-25, 2006.
[8] Earthexplorer. [Online]. Available: http://earthexplorer.usgs.gov/

Dang Thanh Hai received the bachelor on applied mathematics and informatics at Da Lat University and the master degree on computer science at University of Science, Viet Nam National University Ho Chi Minh City. Now, he is a PhD student at VNU University of Science, Vietnam National University. His research interests include cloud computing and wireless sensor network. He worked as a lecturer at Faculty of Information Technology, Da Lat University.

Nguyen Thi Tam is a young lecturer at the Faculty of Mathematics, Mechanics and Informatics, VNU University of Science. She received the B.S. degree in applied mathematics and informatics from VNU University of Science in 2012. She is currently working towards her M.S. degree in mathematical foundation of computer science. Her current research interest includes algorithms in computer network, wireless sensor network.

Le Hoang Son obtained the PhD degree on mathematics – informatics at VNU University of Science, Vietnam National University. Currently, he is a researcher at VNU University of Science, Vietnam National University. His major field includes soft computing, fuzzy clustering, recommender systems, geographic information systems and particle swarm optimization. He is a member of IACSIT and also an associate editor of the International Journal of Engineering and Technology (IJET).

Vinh Trong Le received the MSc degree in information technology from Faculty of Mathematics, Mechanics and Informatics, Hanoi University of Science, Vietnam National University in 1997, the PhD degree in computer science from Japan Advanced Institute of Science and Technology in 2006, respectively. He is currently an associate professor at the Faculty of Mathematics, Mechanics and Informatics, Hanoi University of Science, Vietnam National University. His research interests include algorithm theory, next generation network, wireless network and network security.