Uludağ University Journal of The Faculty of Engineering, Vol. 25, No. 1, 2020 RESEARCH

DOI: 10.17482/uumfd.632815

WIRELESS SENSOR DEPLOYMENT ON 3-D SURFACE OF MOON TO MAXIMIZE COVERAGE BY USING A HYBRID MEMETIC ALGORITHM

Omer OZKAN*

Received: 14.10.2019; revised: 23.02.2020; accepted: 06.03.2020

Abstract: The moon has always been a goal for humanity in history to reach and discover. Since the 1950s, many missions have been carried out in order to achieve this goal. Wireless sensor networks can be a good tool for discovering some of the features of the moon and acquiring very important information for the missions to the moon and beyond to be performed soon. The deployed seismic, monitoring, light, temperature, pressure, etc. types of sensors on the surface of the Moon can collect vital data for the missions. Therefore, in this paper, the wireless sensor deployment problem on the surface of the Moon is studied to maximize coverage. Since the deployment of sensors on 3-D terrain is an NP-hard problem, a hybrid memetic algorithm is developed to solve. The real 3-D digital elevation model of the surface of the Moon for two different terrains near the South Pole is used to test the performance of the proposed algorithm with 64 scenarios and the results are compared with local search and simulated annealing algorithms. According to the results, the proposed hybrid memetic algorithm has better coverage values than the others in acceptable CPU times.

Anahtar Kelimeler: Local search, memetic algorithm, moon, sensor coverage, simulated annealing, wireless sensor deployment

Üç Boyutlu Ay Yüzeyine Kapsamayı Enbüyüklemek Üzere Melez Memetik Algoritma Kullanarak Kablosuz Algılayıcı Yerleştirilmesi

Öz: Ay, tarihte insanlığın her zaman ulaşması ve keşfetmesi için bir amaç olmuştur. 1950'lerden bu yana, bu hedefe ulaşmak için birçok görev gerçekleştirilmiştir. Kablosuz algılayıcı ağlar, ayın bazı özelliklerini keşfetmek ve yakında gerçekleştirilecek olan ay ve ötesindeki görevler için çok önemli bilgiler edinmek için iyi bir araç olarak görünmektedir. Ay yüzeyine konuşlandırılabilecek sismik, izleme, ışık, sıcaklık, basınç vb. algılayıcı tipleri görevler için hayati veriler toplayabilecektir. Bu nedenle, bu çalışmada kapsamayı en üst düzeye çıkarmak için Ay yüzeyine kablosuz algılayıcı konuşlandırma problemi incelenmiştir. Algılayıcıların üç boyutlu arazide konuşlandırılması NP-zor bir problem olduğundan, çözmek için melez bir memetik algoritma geliştirilmiştir. Güney Kutbu yakınındaki iki farklı arazi için Ay yüzeyinin gerçek üç boyutlu dijital yükseklik modeli 64 senaryo ile önerilen algoritmanın performansını test etmek için kullanılmış ve sonuçlar yerel arama ve tavlama benzetimi algoritmaları ile karşılaştırılmıştır. Sonuçlara göre, önerilen melez memetik algoritma kabul edilebilir CPU zamanlarında diğerlerinden daha iyi kapsama değerlerine sahiptir.

Keywords: Yerel arama, memetik algoritma, ay, algılayıcı kapsaması, tavlama benzetimi, kablosuz algılayıcı yerleştirme

* Industrial Engineering Department, Turkish Air Force Academy, National Defense University, 34149, Yesilyurt, Istanbul, Turkey
Corresponding Author: Omer Ozkan (o.ozkan@hho.edu.tr)

1. INTRODUCTION

In the history of humankind, Earth's only natural satellite, Moon has always been influenced the beliefs, thoughts, and ideas in many cultures. It has been a source to understand and measure the time and tides. But especially after the 1950's, the technology allowed humans to discover the Moon more deeply. There are more than hundred successful and unsuccessful missions dedicated explaining the Moon and its features (NASA, 2019). In recent years, adding to the USA and Russia (USSR), Japan, China, Europe, India, and Israel have joint to the competition. Fifty years passed for the landing of the first human on the Moon by Apollo 11, and nowadays the scientists are planning to discover the far side of the Moon (Wang and Liu, 2016; Jia et al., 2018).

As the outcome of the development in the technology of micro-electro-mechanical-systems, wireless sensor networks (WSNs) have earned an increasing interest in recent years. A WSN is the wireless network of small-sized, low-costed, short-ranged, high-reliable, low-powered, easy-deployed and multi-functional sensors connected to collect and deliver the desired types of data from the environment for different kinds of missions. The abilities of WSNs are ranging from detection, measuring, monitoring, surveillance, reconnaissance, targeting to tracking and the subjects of the missions can be humans, animals, places, vehicles, terrains, etc. (Akyildiz et al., 2002; Yick et al., 2008). Because of these features, WSNs can be used in undiscovered and dangerous places in space, such as Moon, other natural satellites, asteroids or planets. WSNs can help to provide important seismic, monitoring, light, temperature, pressure or other types of data to understand the nature of the terrain and maybe to help to a lunar rover or to facilitate the establishment of the first human settlements. It is a difficult task to place a WSN on the Moon because there are lots of craters on the surface of the Moon, there can not spend a huge amount of sensors to the Moon because it is expensive, and also there is no atmosphere.

Despite the Moon is the closest and reached target in space, there are not many studies in the literature about the usage of the WSNs on the Moon. Pabari, Acharya, and Desai (2009) investigated the theoretical sides of sensor deployment on the Moon. The study covers different deployment topologies with their costs, coverage and connectivity properties. The following study by Pabari et al. (2010) searched the radio signal coverage patterns of four lunar terrains by examining real digital elevation model data of the Moon. Prasad and Murty (2011) and Pabari et al. (2013) proposed to use WSNs for in situ exploration of lunar water or ice. Zhai and Vladimirova (2015) developed data aggregation algorithms and simulation models to reduce the number of deployed sensors and their energy consumption for the WSNs on the Moon. In another study by Zhai and Vladimirova (2016) data-processing/fusion algorithms are presented to integrate the scientific sensor data collected by a WSN with protecting the quality of data and decreasing the volume of the data while satisfying energy constraints. Lopez-Matencio (2016) studied a multi-objective WSN deployment problem with the selection of optimal observation positions and maximization of the lifetime of the WSN. An ant colony optimization metaheuristic algorithm is proposed to solve the defined problem. Parrado-Garcia (2017) proposed a simulated annealing (SA) algorithm to solve a comprehensive mass-constrained WSN deployment problem on the Moon with some properties, such as; heterogeneous relevance on the map, detailed lifetime, node airborne launch, or propellant calculations. In recent years, the Moon-based Earth observation sensor studies are increased. The main objective is deploying sensors to observe Earth from Moon (Hamill, 2016; Ye, Guo, and Liu, 2017; Ye et. al., 2018a; Ye et. al., 2018b; Wang et. al., 2019; Guo et. al., 2020).

In the literature, WSNs are also been suggested to use in other space environments (Dubois et al., 2009; Sun, Guo, and Gill, 2010; Wilson and Atkinson, 2011). For example, the usage of WSNs for exploring the planets are presented by Gaura and Newman (2006), Medina et al. (2010), Pabari, Acharya, and Desai (2012), Prasad, Bhattacharya, and Murty (2012), Sanz et al. (2013), Rodrigues et al. (2014) and Oddi et al. (2017). As a real project example on this subject, European Space Agency (ESA) has funded a research project named RF Wireless for Planetary Exploration (RF-WIPE) to investigate the deployment of a number of self-organizing wireless sensor nodes

for exploration of a planetary body (Sanz et al., 2013). Sun, Guo, and Gill (2010) presented some WSN application scenarios on space as; autonomous formation flying, very-small-satellite cluster/swarm, fractionated spacecraft, onboard sensor network, surface vehicles on the Moon, Mars and other planets or asteroids. The deployment issues of WSNs on Mars are also investigated by Del Re, Pucci, and Ronga (2009) and Ulmer, Yalamanchili, and Alkalai (2000).

While deploying a WSN on a terrain, there are different types of strategies and techniques in the literature (Younis and Akkaya, 2008; Türkoğulları et al., 2010; Guerriero et al., 2011; Kulkarni, Förster, and Venayagamoorthy, 2011; Cheng et al., 2012; Deif and Gadallah, 2014; Abdollahzadeh and Navimipour, 2016). The methodologies for the deployment problem in the literature can be classified as exact methods, heuristics, metaheuristics, approximation algorithms or artificial intelligence techniques, etc. Since the complexity of the deployment problem in WSNs is NP-hard (Younis and Akkaya, 2008), plenty of types of metaheuristics are preferred to solve the problem (i.e. Molina and Alba, 2011; Tsai et al., 2015). It is important to find optimal deployment positions on the terrain for the sensors to increase the cost-effect. The works in the literature are generally on 2D terrains but it is more realistic to use 3-D terrains as an example and the usage of 3-D terrain makes the problem harder.

The coverage of WSNs is also a well-studied problem in the literature and generally associates with the deployment and connectivity issues (Ghosh and Das, 2008; Fan and Jin, 2010; Seok et al., 2013; Mini, Udgata, and Sabat; 2014). There are also single and multi-objective metaheuristic algorithm examples to solve deployment and coverage problems while satisfying connectivity constraints simultaneously (Molina, Alba and Talbi, 2008).

In this paper, since the WSN deployment problem has high complexity, as a contribution to the literature a hybrid memetic algorithm (HMA) is proposed. The local search (LS), simulated annealing (SA) and genetic algorithm (GA) are hybridized to generate the HMA. The sensors are placed on the real 3-D lunar surface to maximize the aggregated quality of coverage (QoC) of all deployed sensors (Fig. 1). A probabilistic sensing model and a line-of-sight (LOS) algorithm are used for the coverage computations of the sensors. The different numbers of deployed sensors, with different sensor properties, and terrain types are used to generate 64 different scenarios. The performance of the algorithm is compared with pure algorithms.

The study is organized as follows: Section 2 gives the problem definition and the issues about the sensors and their coverage. The LS, SA, and HMA are described as the methodology in Section 3. Section 4 covers the computational results of the proposed algorithms on the real surface data of the Moon. Finally, section 5 concludes the paper.

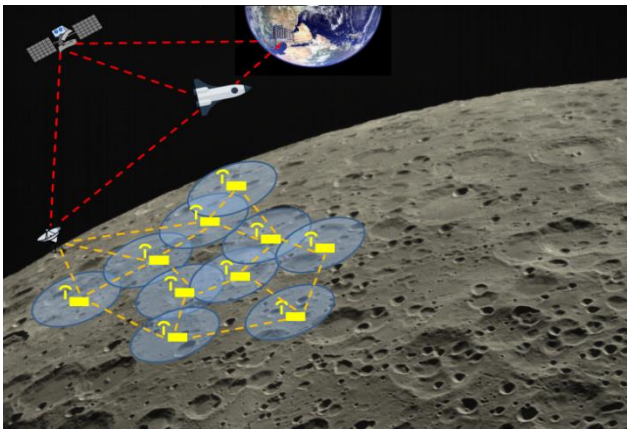

Figure 1:
WSN deployment on Moon

2. THE PROBLEM DEFINITION

The objective of the studied problem is maximizing the overall QoC by all deployed sensors. Therefore, in the algorithms, the most realistic and commonly used sensor sensing model is preferred as the probabilistic sensing model to calculate the QoC. The inspired model is proposed by Zou and Chakrabarty (2003) and Wu, Lee, and Chung (2006) and it calculates the probability of a pixel (P) to be covered via a deployed sensor (S_i) and provides a number between 0 and 1. In the calculations; the 3-D Euclidean distance between P and S_i is donated as $d(S_i, P)$, the predefined sensing range as S_{range}, the measure of uncertainty in sensing range as P_{range} where $P_{range} < S_{range}$, and the probabilistic sensing degree of the corresponding pixel P via S_i as $Cov(S_i, P)$. As seen in Fig. 2 and Eq. 1, if the P is located in the area of (S_{range} - P_{range}) and there is a LOS between S_i and P, the P is certainly covered and sensed by S_i. If the P is placed between (S_{range} - P_{range}) and (S_{range} + P_{range}) and there is a LOS between S_i and P, the detection probability can be expressed with an exponential function described in Eq. 1 and illustrated in Fig. 2. The function can be adjusted by changing different sensor parameters λ and β to simulate different sensor types. And if the P lies out of (S_{range} + P_{range}) or there is non-LOS (NLOS) between S_i and P, the P is not certainly sensed by S_i. The *dist* in Eq. 2 is the ratio of $d(S_i, P)$ within probabilistic range $2*P_{range}$. In this work, the 3-D Bresenham LOS algorithm is used to detect the visibility among pixel P and sensor S_i (Bresenham, 1965). A LOS is the 3-D Euclidean line between pixel P and sensor S_i and there is no cut by any other object or piece of terrain on the line.

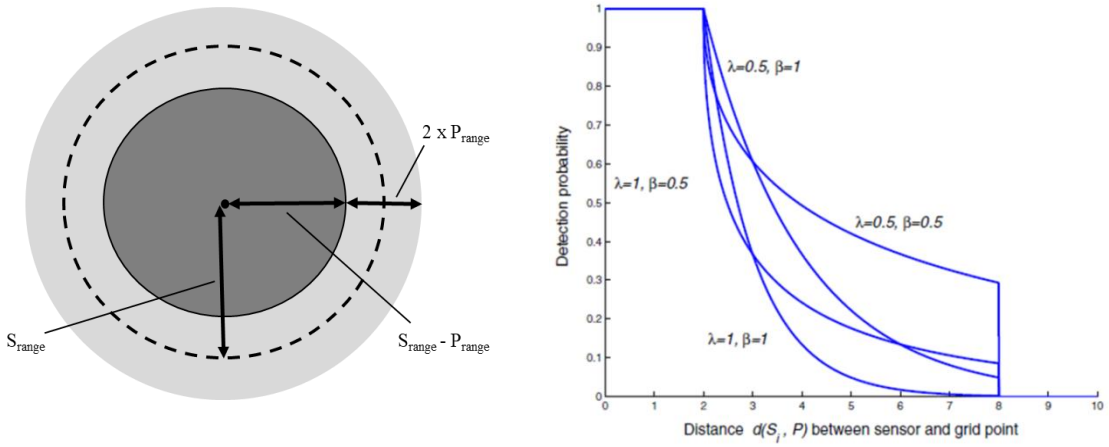

Figure 2:
Sensing range and probabilistic sensor detection model (Zou and Chakrabarty, 2003; Wu, Lee, and Chung, 2006)

$$Cov(S_i, P) = \begin{cases} 0, if\ d(S_i, P) > (S_{range} + P_{range}) \\ or\ if\ NLOS \\ e^{-\lambda . dist^{\beta}}, if\ (S_{range} - P_{range}) \leq d(S_i, P) < (S_{range} + P_{range}) \\ and\ \ if\ LOS \\ 1, if\ d(S_i, P) \leq (S_{range} - P_{range}) \\ and\ \ if\ LOS \end{cases} \quad (1)$$

$$dist = \frac{d(S_i, P) - (S_{range} - P_{range})}{2 * P_{range}} \quad (2)$$

In the study, it is considered that the terrain *Ter*, has *M* x *M* pixels. Initially, every pixel in the *Ter* is numbered between 1 and *M* x *M* and *Ter* is sub-divided into *N* sub-regions. The pixel length of each sub-region is denoted as l_s. The start and end coordinates of every sub-region in the horizontal axis, x_s and x_e and the start and end coordinates of every sub-region in the vertical axis, y_s and y_e are determined with Eq. 3 where Ter_i denotes the sub-region number (i = 1, 2, …, N).

$$\begin{gathered} x_s = x_e - l_s + 1 \,, x_e = \left\lceil \frac{Ter_i}{l_s} \right\rceil \times l_s \\ x_s = x_e - l_s + 1 \,, y_e = mod(Ter_i, l_s) \times l_s \\ y_s = x_e - l_s + 1 \end{gathered} \tag{3}$$

3. THE METHODOLOGY

3.1. Local Search

LS is one of the simplest (meta)heuristic in the literature that starts with an initial solution and proceeds with accepting only better neighbor solutions in iterations. This procedure of the LS may cause stuck on local optimal points, but it is still useful when the converged solution is close to the global optimal and when intensification is needed in the search process. The pseudo-code of the generated LS algorithm in this study is presented in Fig. 3.

```
1: begin
2: Set initial parameter values (iter_max)
3: Construct a random initial solution (X_0)
4: Evaluate the initial solution (f_0)
5: Generate memory (M): X_M = X_0 and f_M = f_0
6: iter = 0    /* iter: iteration number */
7: repeat
8:     Generate a new neighbor solution (X_iter)
9:     Evaluate the new solution (f_iter)
10:    if f_iter ≥ f_M then
11:        Accept X_iter and update M = (X_iter; f_iter)
12:    else
13:        Deny X_iter
14:    end if
15:    iter ← iter + 1
16: until iter = iter_max
17: display M
18: end
```

Figure 3:
Local Search (LS)

The LS and the all other developed algorithms are using an integer representation to symbolize a solution (Fig. 4). The integer numbers in the representation indicate the pixel numbers on the *Ter* that the sensors are deployed. The length of the array depends on the number of deployed sensors on the *Ter*. The initial solution of the LS is found randomly, i.e. the predetermined number of sensors is placed on the pixels in the *Ter* randomly. The fitness function of the LS is the total QoC value of all deployed sensors.

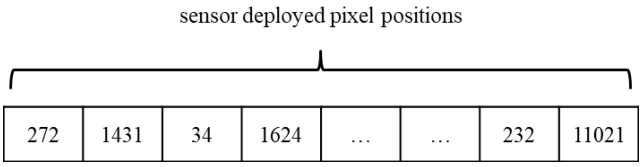

Figure 4:
Representation

The neighborhood function of the LS has a resetting procedure (Fig. 5). The function randomly selects a sensor and it changes its deployment pixel. The LS continues with the new neighbor if the neighbor has only a better QoC value than the previous one. The $iter_{max}$ value determines the stopping iteration of the LS and it is determined in the experimental study.

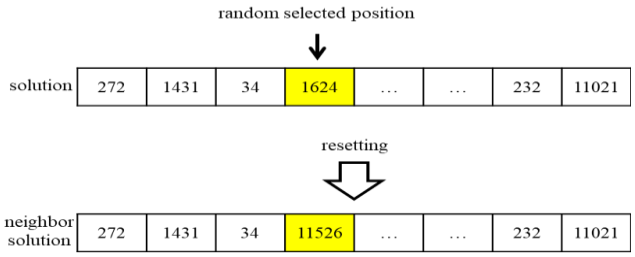

Figure 5:
Neighborhood function

3.2. Simulated Annealing

The SA is a metaheuristic that is inspired by the physical annealing and the SA has the ability to escape from local optimal points by using a temperature parameter (Aarts and Korst, 1989). The pseudo-code of the designed SA for the problem is presented in Fig. 6.

The integer representation, initial solution determination, fitness value calculation, neighborhood function of LS are the same for the SA. The initial temperature (T_0) of SA is determined by a Markov chain-based algorithm proposed by Aarts and Korst (1989) presented in Fig. 7. The T_0 is set at a temporary large value for the Markov chain in the beginning and after a full Markov chain is completed, T_0 is derived using Eq. 4.

$$\gamma = \frac{m_1 + m_2 \, x \, e^{\frac{-\Delta f}{T_0}}}{m_1 + m_2} \tag{4}$$

In Eq. 4., m_1 represents the number of moves that have resulted in an increased QoC, m_2 represents the moves when the QoC decreased relative to the previous step, and Δf represents the average QoC increase after m_2 moves. The acceptance ratio (γ) of the worse results is set to 95% in the beginning, it is decreased subsequently. The defined maximum iteration number ($iter_{max}$) is divided into ρ parts equally and at the end of each division, the temperature is decreased to $T_{iter+1} = \alpha * T_{iter}$ ($iter$ = 0, 1, 2, ...) starting with T_0. The α, ρ, and $iter_{max}$ parameters are tuned before computational experiments.

1: **begin**
2: Set initial parameter values (T_0, α, ρ, $iter_{max}$)
3: Construct a random initial solution (X_0)
4: Evaluate the initial solution (f_0)
5: Generate memory (M): $X_M = X_0$ and $f_M = f_0$
6: $iter = 0$, $T_{iter} = T_0$ /* $iter$: iteration number */
7: **repeat**
8: Generate a new neighbor solution (X_{iter})
9: Evaluate the new solution (f_{iter})
10: **if** $f_{iter} \geq f_M$ **or** rand(0,1) $\leq e^{\frac{-(f_M - f_{iter})}{T_{iter}}}$ **then**
11: Accept X_{iter} and update $M = (X_{iter}; f_{iter})$
12: **else**
13: Deny X_{iter}
14: **end if**
15: Decrease T_{iter} using α and ρ
16: $iter \leftarrow iter + 1$
17: **until** $iter = iter_{max}$
18: display M
19: **end**

Figure 6:
Simulated Annealing (SA)

1: **Initial temperature (T_0)**
2: POST: T_0 provides about 95% chance of changing solutions
3: Select an initial solution X_0
4: **repeat** n **times**
5: generate a neighboring solution X_{iter}
6: compute $\Delta f = f(X_0) - f(X_{iter})$
7: **if** $\Delta f > 0$ **then**
8: $sum = sum + \Delta f$
9: **end loop**
10: set $ave = sum / n$
11: find a T_0 such that $0.95 < e^{\frac{-ave}{T_0}}$

Figure 7:
Initial temperature calculation algorithm (Aarts and Korst,1989)

3.3. Hybrid Memetic Algorithm

GA is one of the most preferred metaheuristics in the literature inspired by evolution (Holland, 1975). The GA starts searching with a group of solutions (i.e. a population) and generates better solutions (i.e. individuals) by using crossover, mutation, and selection operators through generations. The idea of GA is that the fittest individuals (i.e. having high fitness values) produce better children. The hybridization of GA and LS creates a new algorithm called memetic algorithm (MA). The proposed HMA is the hybrid version of GA, LS, and SA used together. The pseudo-code of the proposed HMA is in Fig. 8.

1: **begin**
2: Set initial parameters (μ, k, P_c, P_m, gen_{max})
3: Use μ # of SA solutions as initial population
4: Evaluate initial population
5: gen = 0 /* gen: generation number */
6: **repeat**
7: Apply tournament to select parents using k
8: Generate μ # of new children with crossover by using P_c
9: Apply LS as mutation operator by using P_m
10: Evaluate new children
11: Apply environmental selection (update population)
12: $gen \leftarrow gen + 1$
13: **until** $gen = gen_{max}$
14: Show best solution
15: **end**

Figure 8:
Hybrid Memetic Algorithm (HMA)

The hybridization of algorithms aims to use the powerful sides of the algorithms together in a single algorithm. Therefore, in HMA it is considered to use the SA as the initial population generator algorithm and the LS as the mutation operator. In the GA literature, instead of starting with a random population, generating (meta)heuristic-based populations helps to the algorithm in the searching process. The SA is a fast algorithm and is good at both diversification and intensification in the search space, therefore it is embedded to a GA as the initial population generator algorithm. The LS is also working well when intensification in searching is needed, hence it is combined with the GA as the mutation operator. The integer representation and fitness value calculation of the LS and the SA are the same for the HMA. The tournament selection operator is preferred to select parents in the proposed HMA. In the tournament selection, the algorithm selects k number of individuals randomly from the population, and the solution with the highest fitness (i.e. highest QoC) wins the tournament and becomes a parent. The crossover probability (P_c) is used to determine whether the selected parent can be a candidate in the crossover. To generate new children, the one-point crossover operator is selected (Fig. 9). The operator uses two parents and it selects a random breakpoint in the representation of the parents. The operator swaps the cross tails of both selected parents to generate a new child.

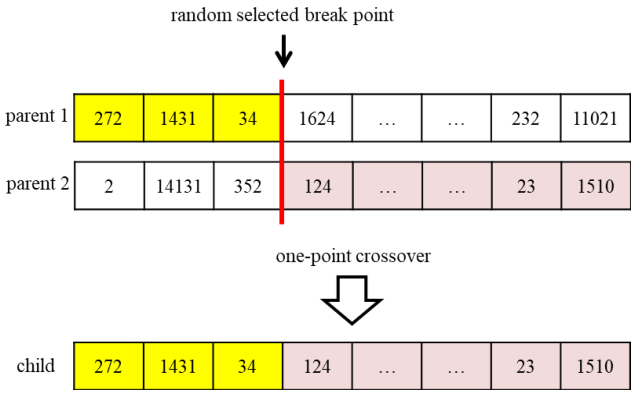

Figure 9:
One-point crossover operator

The LS is used as the mutation operator and mutation probability (P_m) is used to determine whether the generated new child can be mutated. The HMA generates μ number of new children and with an elitist strategy, the best μ number of individuals of the last two populations (i.e. the

fittest in $\mu+\mu$) become the new generation in the environmental selection. The HMA stops when it reaches to gen_{max} number of generations. The k, P_c, P_m, and gen_{max} parameters are tuned in experiments.

4. COMPUTATIONAL RESULTS

In the experimental study, the performance of the proposed HMA is tested in real 3-D terrains from the Moon and compared with the pure LS and SA algorithms with 64 different scenarios. The real lunar DEMs are taken from Selene mission between 89.9 – 86.0 degrees longitude (south) and 0-15 degrees latitude (east) on the Moon and the 3-D illustration of the selected region is presented in Fig. 10 (Map, 2019).

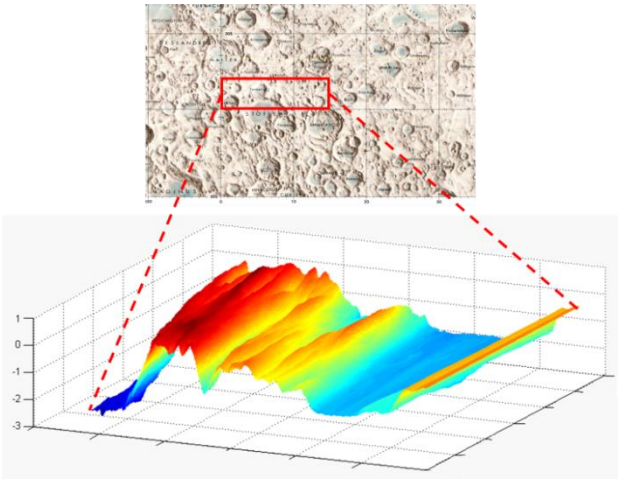

Figure 10:
The 89.9 – 86.0 degrees longitude (south) and 0-15 degrees latitude (east) on the Moon

By examining the DEMs of the selected region on the Moon, two terrains (i.e. called smooth and harsh terrains) are selected and their 3-D models are presented in Fig. 11. The width and length of the terrains are assigned as 128 pixels. 64 different scenarios are developed for the terrains and are described in Table 1. In scenarios, the S_{range} can be 5, 10, 15 or 20 pixels and the P_{range} can be 1, 2, 3, or 4 pixels. Since deploying a WSN on the Moon is expensive, only 16 or 64 number of sensors are deployed considering the sizes of the terrains.

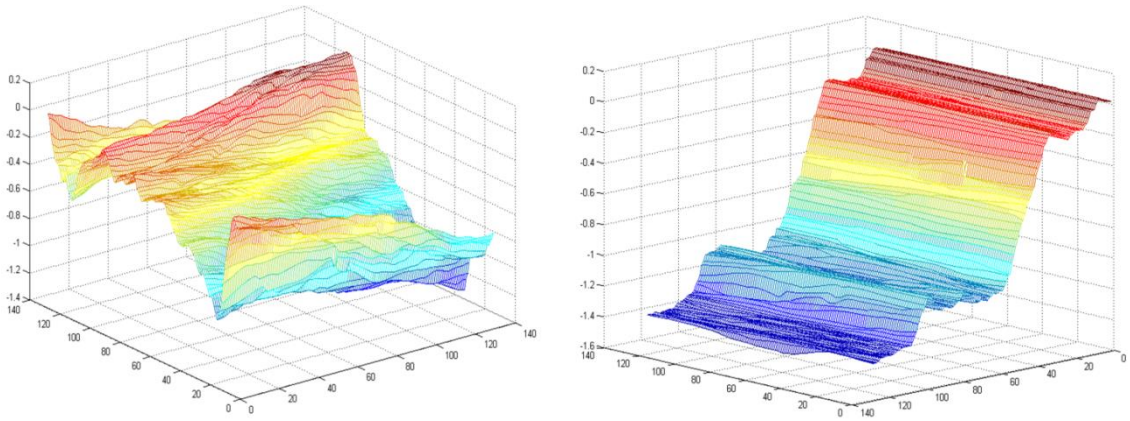

Figure 11:
Smooth (on the left) and harsh (on the right) terrains

Table 1. The scenarios

Ter. Type	# of S	S_{range} (pixels)	P_{range} (pixels)	Scen. #	Ter. Type	# of S	S_{range} (pixels)	P_{range} (pixels)	Scen. #
Smooth	16	5	1	1	Harsh	16	5	1	33
			2	2				2	34
			3	3				3	35
			4	4				4	36
		10	1	5			10	1	37
			2	6				2	38
			3	7				3	39
			4	8				4	40
		15	1	9			15	1	41
			2	10				2	42
			3	11				3	43
			4	12				4	44
		20	1	13			20	1	45
			2	14				2	46
			3	15				3	47
			4	16				4	48
	64	5	1	17		64	5	1	49
			2	18				2	50
			3	19				3	51
			4	20				4	52
		10	1	21			10	1	53
			2	22				2	54
			3	23				3	55
			4	24				4	56
		15	1	25			15	1	57
			2	26				2	58
			3	27				3	59
			4	28				4	60
		20	1	29			20	1	61
			2	30				2	62
			3	31				3	63
			4	32				4	64

The algorithms are coded in MATLAB (ver. R2014a). The experiments are done on a computer with Intel Core-i5-7500 CPU, 3.40 GHz, and 8.00 GB RAM specifications. The parameters of the algorithms are assigned or tuned by conducting some experiments on the smooth terrain with 16 number of deployed sensors and the values are presented in Table 2.

Table 2. Tuned or assigned parameters of the algorithms

Algorithm	Parameter	Candidate Values	Assigned Value
LS	$iter_{max}$	-	1000
SA	$iter_{max}$	-	1000
	T_0	-	10.32 °C
	α	0.7 ; 0.8 ; 0.9	0.8
	ρ	200 ; 500 ; 1000	500
HMA	gen_{max}	-	500
	μ	-	30
	P_c	-	1
	P_m	0.1 ; 0.2 ; 0.3	0.3
	k	3 ; 5 ; 7	5

Each scenario is solved 30 times with all algorithms. The best, mean and worst results of the algorithms for the smooth terrain is presented in Table 3. The results show that the HMA has better values according to the means and the worst results of 30 runs for all scenarios. The HMA found the best QoC values in 27 of 32 scenarios, just for 5 scenarios LS found better results in the best category, SA found the same results with the HMA in 4 of 32 scenarios in the best column. The HMA improved the LS and SA QoC results up to 15% which is a very noteworthy value for a coverage problem.

The results of the algorithms for the harsh terrain is illustrated in Table 4. Similar to the smooth terrain results, HMA outperformed the LS and the SA in the mean and worst column. The HMA reached better solutions in 26 of 32 scenarios and the LS in 6 of 32 scenarios in the best category. The SA found the same results with the HMA in 2 scenarios in the best column. The HMA has up to 19% better solutions than the SA and the LS in the best results category that is a good improvement in QoC.

The CPU times (s.) of the algorithms are presented in Table 5 and Table 6 for the smooth and harsh terrains, respectively. The CPU times for the LS ranging from 13 to 1143 seconds, for the SA 12 to 1132 seconds, and for the HMA 194 to 16903 seconds for the smooth terrain. The times are changing between 12 to 1079, 12 to 1055, and 195 to 16515 seconds for the LS, SA, and HMA, respectively for the harsh terrain. When the deployed number of sensors is constant, the change in the S_{range} and the P_{range} values in the scenarios are dramatically affecting the run times. There is not much difference between the terrain types for the run times for the scenarios with similar properties. The CPU times of the algorithms are acceptable for that kind of mission.

For both terrains, when the deployed number of sensors, the S_{range}, and the P_{range} values are changed, the covered area diversifies on a very wide scale (i.e. 6% to 99%). The effect of changing the S_{range} and the P_{range} values in the performance of the HMA in the best results category for both terrains is illustrated in Fig. 12.

Table 3. The QoC results for the smooth terrain

Scen. #	HMA (%)			SA (%)			LS (%)		
	Best	Mean	Worst	Best	Mean	Worst	Best	Mean	Worst
1	**7.98**	**7.95**	**7.93**	7.72	7.62	7.48	7.67	7.60	7.49
2	**8.00**	**7.95**	**7.90**	7.75	7.58	7.46	7.71	7.58	7.43
3	**8.99**	**8.99**	**8.99**	**8.99**	8.73	8.50	8.92	8.71	8.54
4	**9.49**	**9.46**	**9.44**	9.04	8.70	8.49	9.00	8.71	8.51
5	**30.73**	**30.66**	**30.61**	28.32	27.42	26.56	28.38	27.53	26.62
6	**29.72**	**29.60**	**29.43**	27.43	26.55	25.75	27.24	26.60	25.99
7	**29.05**	**28.95**	**28.75**	25.86	25.09	24.33	26.28	25.12	24.43
8	**29.05**	**28.93**	**28.73**	26.18	25.13	24.12	26.38	25.22	24.64
9	**79.25**	**79.25**	**79.25**	**79.25**	74.80	71.16	77.10	74.66	72.68
10	76.68	**76.68**	**76.68**	76.68	73.71	71.06	**78.67**	73.86	70.76
11	72.27	**72.27**	**72.27**	72.27	69.90	67.89	**73.67**	70.09	67.31
12	73.34	**73.34**	**73.34**	73.34	69.94	66.48	**73.93**	70.52	68.25
13	**92.21**	**91.42**	**89.81**	78.38	74.72	72.08	78.58	74.64	72.75
14	**90.52**	**89.74**	**88.41**	77.23	73.95	71.61	76.83	73.74	70.96
15	**87.97**	**87.34**	**85.75**	74.60	72.18	69.23	73.10	70.00	67.50
16	**84.70**	**84.14**	**83.31**	72.18	70.03	67.13	75.54	70.20	68.33
17	**31.13**	**31.08**	**31.01**	28.53	27.99	27.62	28.58	27.94	27.51
18	**30.97**	**30.87**	**30.76**	27.52	27.11	26.69	27.73	27.20	26.71
19	**31.97**	**31.78**	**31.41**	28.17	27.22	26.82	27.71	27.20	26.79
20	**33.30**	**33.00**	**32.79**	27.74	27.34	26.71	27.95	27.26	26.74
21	**92.24**	**91.69**	**91.03**	78.57	76.84	74.98	79.17	77.18	75.34
22	**85.87**	**85.49**	**84.84**	74.65	73.25	72.23	75.35	73.22	72.00
23	**77.98**	**77.55**	**76.96**	63.55	62.39	61.43	63.38	62.29	61.30
24	**71.01**	**70.43**	**70.06**	62.93	62.20	61.25	63.79	62.37	61.01
25	98.66	**98.66**	**98.66**	98.66	98.06	97.56	**98.73**	98.09	97.11
26	98.41	**98.41**	**98.41**	98.41	97.96	96.92	**98.73**	97.94	97.06
27	**98.37**	**98.37**	**98.37**	**98.37**	97.60	96.89	98.17	97.57	96.96
28	**98.51**	**98.51**	**98.51**	**98.51**	97.70	97.13	98.35	97.57	96.78
29	**99.94**	**99.89**	**99.81**	98.59	98.01	97.01	98.72	97.97	97.35
30	**99.93**	**99.88**	**99.77**	98.49	97.97	97.42	98.72	97.99	96.80
31	**99.91**	**99.85**	**99.78**	98.02	97.53	96.92	98.13	97.69	97.03
32	**99.92**	**99.87**	**99.77**	98.15	97.68	97.12	98.42	97.75	96.65

Table 4. The results for the harsh terrain

Scen. #	HMA (%)			SA (%)			LS (%)		
	Best	Mean	Worst	Best	Mean	Worst	Best	Mean	Worst
33	**7.80**	**7.75**	**7.71**	7.07	6.92	6.78	7.03	6.91	6.78
34	**7.80**	**7.68**	**7.62**	7.02	6.77	6.59	6.92	6.77	6.60
35	**8.03**	**7.96**	**7.71**	7.71	7.44	7.17	7.88	7.43	7.02
36	**8.93**	**8.88**	**8.81**	7.63	7.37	7.15	7.77	7.43	7.15
37	**28.67**	**28.42**	**28.22**	24.15	23.02	21.88	24.58	22.74	21.47
38	**27.72**	**27.49**	**27.26**	23.27	21.98	21.19	23.17	22.23	21.22
39	**26.86**	**26.72**	**26.55**	22.20	20.95	19.74	22.07	20.77	20.14
40	**26.77**	**26.45**	**26.16**	22.11	21.02	19.93	21.79	20.92	19.99
41	67.55	**67.55**	**67.55**	67.55	63.26	59.74	**67.64**	62.52	60.27
42	66.24	**66.24**	**66.24**	66.24	62.59	59.44	**67.55**	62.85	60.48
43	62.72	**62.72**	**62.72**	62.72	59.45	56.73	**64.23**	60.10	55.95
44	**63.03**	**63.03**	**63.03**	**63.03**	59.46	56.82	61.87	59.43	57.08
45	**83.45**	**82.25**	**80.96**	67.71	63.77	61.47	66.60	62.89	59.79
46	**82.18**	**81.03**	**80.01**	67.20	62.09	58.99	68.09	62.48	59.46
47	**80.90**	**79.62**	**78.25**	62.27	59.59	56.38	62.66	59.80	57.94
48	**78.06**	**77.26**	**76.13**	62.48	59.52	57.39	62.53	59.43	56.67
49	**29.91**	**29.80**	**29.63**	25.69	25.00	24.48	25.83	25.07	24.54
50	**29.37**	**29.20**	**29.03**	24.77	24.02	23.56	24.53	23.96	23.54
51	**29.90**	**29.59**	**29.24**	24.11	23.32	22.68	24.19	23.45	22.70
52	**30.63**	**30.34**	**30.01**	23.95	23.32	22.62	23.82	23.40	22.74
53	**85.15**	**84.25**	**82.94**	66.90	65.48	63.72	68.29	65.86	64.63
54	**80.12**	**79.29**	**78.76**	64.46	62.85	61.54	64.39	62.64	61.50
55	**72.56**	**71.91**	**71.48**	55.52	54.50	53.24	56.29	54.66	53.38
56	**65.86**	**65.56**	**65.13**	56.40	54.84	53.75	56.03	54.63	53.45
57	95.51	**95.51**	**95.51**	95.51	94.18	93.12	**95.35**	94.17	93.12
58	94.81	**94.81**	**94.81**	94.81	93.96	92.64	**95.12**	94.14	92.80
59	93.99	**93.99**	**93.99**	93.99	92.89	91.93	**94.16**	92.94	91.90
60	**94.02**	**94.02**	**94.02**	**94.02**	92.90	91.50	93.59	92.79	91.41
61	**99.86**	**99.56**	**99.08**	95.15	94.45	93.43	95.01	94.13	93.41
62	**99.77**	**99.51**	**99.10**	95.32	93.98	93.08	95.25	94.07	92.85
63	**99.81**	**99.44**	**99.07**	93.72	92.91	91.74	94.20	92.96	91.42
64	**99.62**	**99.29**	**98.87**	94.62	93.09	91.82	93.89	93.03	92.32

Table 5. The CPU times (s.) for the smooth terrain

Scen. #	HMA (s.)			SA (s.)			LS (s.)		
	Best	Mean	Worst	Best	Mean	Worst	Best	Mean	Worst
1	194.01	195.85	197.80	12.57	12.75	13.19	13.01	13.24	13.64
2	252.36	258.84	264.29	17.17	17.61	18.64	17.80	18.50	19.46
3	361.06	362.21	364.11	30.39	31.15	32.50	31.26	31.87	33.90
4	475.03	486.42	491.93	30.39	30.69	31.11	31.14	31.46	31.88
5	763.34	767.17	769.99	46.57	47.14	47.92	47.85	48.41	49.05
6	911.51	919.21	932.77	55.97	56.78	63.12	56.94	57.57	58.19
7	1109.01	1123.49	1139.94	80.65	81.69	83.05	82.26	83.47	84.34
8	1344.71	1366.81	1382.30	80.22	81.89	83.63	82.72	83.73	84.50
9	1689.29	1694.81	1698.62	198.03	203.20	209.40	202.58	205.22	209.73
10	1917.70	1924.10	1933.01	222.91	227.22	238.13	225.62	230.08	233.45
11	2230.06	2236.62	2246.94	269.88	273.14	278.45	274.99	279.14	284.21
12	2569.79	2581.72	2589.93	265.01	272.75	278.57	272.67	278.81	286.62
13	3301.01	3350.25	3418.71	196.54	201.04	204.62	202.23	204.59	207.42
14	3661.53	3717.68	3764.57	221.90	225.50	228.01	226.63	229.67	233.90
15	4010.81	4095.16	4156.18	248.93	253.62	263.78	274.94	279.36	283.79
16	4347.18	4430.84	4534.72	271.37	277.92	286.47	273.77	280.05	295.05
17	766.08	773.40	812.35	50.47	50.82	51.31	51.48	51.80	52.18
18	1033.28	1044.98	1057.85	68.33	68.88	69.57	69.34	70.13	70.92
19	1420.67	1440.35	1455.31	121.57	122.66	123.95	124.19	125.01	125.82
20	1871.27	1910.50	2000.34	121.60	122.88	124.15	123.60	125.20	126.36
21	2916.56	2933.15	2962.13	185.50	189.09	193.19	188.92	191.16	193.72
22	3433.60	3479.80	3526.15	224.12	228.28	244.61	227.05	229.40	231.78
23	4091.66	4131.39	4169.36	324.50	330.22	334.29	329.64	332.82	338.63
24	4845.05	4903.68	4990.25	322.97	330.08	338.02	329.30	332.05	335.69
25	6587.18	6623.10	6664.73	799.46	816.77	852.08	820.45	825.20	834.47
26	7536.74	7571.53	7604.55	905.74	919.69	950.08	913.43	926.40	934.36
27	8630.04	8663.62	8694.99	1091.81	1110.70	1131.54	1115.79	1127.49	1140.26
28	9743.89	9804.78	9872.85	1098.35	1109.92	1127.39	1118.41	1129.72	1141.81
29	11931.76	12123.38	12303.92	803.45	817.85	833.83	818.29	824.26	833.45
30	13429.34	13643.65	13861.55	900.18	907.85	919.28	917.59	926.52	937.52
31	14913.40	15143.31	15418.22	1090.23	1107.98	1132.16	1118.61	1129.41	1143.48
32	16326.26	16585.14	16903.84	1093.72	1112.34	1139.40	1117.03	1128.15	1140.55

Table 6. The CPU times (s.) for the harsh terrain

Scen. #	HMA (s.)			SA (s.)			LS (s.)		
	Best	Mean	Worst	Best	Mean	Worst	Best	Mean	Worst
33	195.79	197.51	198.99	12.52	12.61	12.72	12.63	12.72	12.84
34	258.83	263.43	267.78	16.82	16.97	17.11	17.04	17.48	17.94
35	355.46	363.02	374.68	29.60	29.91	30.15	29.87	30.45	31.24
36	483.24	488.76	492.28	29.53	29.88	30.38	29.85	30.15	30.83
37	756.64	761.32	765.93	45.23	45.77	49.65	45.25	45.86	46.51
38	897.33	902.13	910.00	53.72	54.32	55.04	53.77	54.67	55.29
39	1075.72	1096.38	1113.95	76.99	78.20	79.79	77.53	78.35	80.66
40	1254.75	1306.35	1337.16	76.92	78.15	79.29	77.02	78.22	79.04
41	1676.65	1683.31	1696.97	187.55	190.45	192.40	188.11	191.08	193.07
42	1893.43	1901.67	1908.51	211.13	214.22	217.49	211.75	214.98	219.75
43	2200.91	2207.37	2222.09	256.62	260.43	262.63	257.73	261.79	264.87
44	2437.43	2474.07	2488.26	257.13	260.61	264.83	258.65	260.98	264.68
45	3224.55	3283.24	3327.51	187.48	190.79	192.98	188.87	191.63	195.33
46	3537.90	3635.54	3702.30	211.68	214.14	217.46	212.85	215.28	218.43
47	3904.92	4018.42	4079.29	255.91	260.19	263.69	259.12	262.13	265.32
48	4342.79	4407.35	4515.28	256.42	260.27	263.17	257.85	261.62	266.73
49	771.56	775.44	778.07	49.59	50.12	50.64	50.01	50.26	50.69
50	1037.53	1045.37	1056.51	66.90	67.47	68.41	67.34	67.84	69.02
51	1410.77	1430.20	1438.73	117.52	118.86	119.81	118.54	119.57	120.53
52	1856.11	1877.09	1901.08	117.98	118.81	119.64	118.50	120.48	125.96
53	2860.03	2898.84	2928.57	179.29	181.30	183.53	179.37	182.09	184.31
54	3382.22	3426.27	3459.28	213.91	216.39	218.78	216.11	217.50	219.18
55	4020.89	4075.77	4107.54	308.85	312.15	316.90	309.71	312.33	315.49
56	4751.52	4818.31	4909.54	308.25	311.95	316.85	309.11	312.99	318.19
57	6286.80	6312.62	6342.76	757.35	764.75	772.44	758.80	765.55	771.92
58	7134.75	7158.82	7179.36	850.37	857.99	867.46	852.37	860.70	867.62
59	8202.12	8230.22	8261.85	1036.58	1044.74	1051.37	1037.57	1049.70	1079.60
60	9365.18	9409.09	9481.11	1037.33	1043.83	1052.18	1036.09	1045.03	1050.56
61	11678.01	11809.00	11962.90	758.10	763.34	770.51	761.60	765.78	769.91
62	13116.83	13299.58	13553.73	850.62	858.57	869.01	850.42	862.29	874.76
63	14556.41	14735.05	14879.00	1037.74	1045.61	1055.67	1039.08	1046.18	1057.20
64	16141.32	16340.83	16515.35	1033.34	1044.47	1054.37	1039.31	1048.49	1057.91

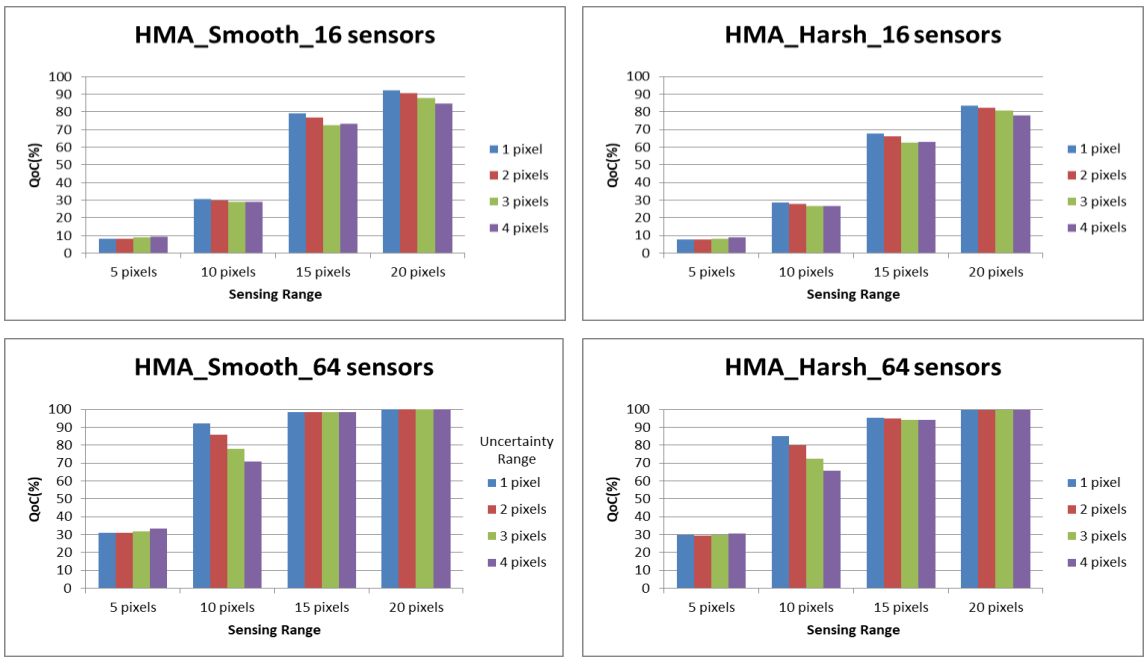

Figure 12:
The performance of HMA for the smooth and the harsh terrains

According to Fig. 12, as expected the variation in the S_{range} and the P_{range} values are very effective in QoC values. Most of the smooth terrain can be covered with 64 sensors when the S_{range} is 15 or 20 pixels. This situation is valid when the S_{range} is 20 pixels with deploying 64 sensors in the harsh terrain. The increase in the P_{range} values is also increasing the QoC up to 21.22% for a constant S_{range}. For instance, the best QoC result of the HMA for the smooth terrain when the S_{range} is 10 and the P_{range} is 1, it is 92.24% (scenario 21) and when the S_{range} is 10 and the P_{range} is 4, it is 71.01% (scenario 24). The difference in both QoC values is 21.22% which represents the more than 1/5 covered area. An illustrative example for the initial and final QoC values reached by the HMA for the harsh terrain with 16 sensors is presented in Fig. 13.

Figure 13:
An initial (f=59,8476) and its final (f=80,2343) deployment sample by the HMA for the harsh terrain with 16 sensors in a single run

The convergence and the improvement ratio through generations of the HMA for the harsh terrain with 16 sensors is shown in Fig. 14. As seen in Fig. 14, the algorithm is starting with around 62% QoC value and it is improving the covered area above 77% in 500 generations. Near 500 generations, it is seen that the convergence ratio of the HMA is slowing, therefore 500 is assigned as gen_{max} value.

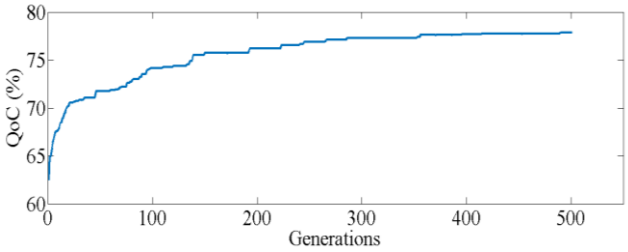

Figure 14:
A sample run and the convergence of HMA for the harsh terrain with 16 sensors

5. CONCLUSION

Optimal deployment of a predetermined number of sensors on the real 3-D lunar surface in order to achieve maximum sensing QoC is a difficult mission. In the scope of this paper, three metaheuristic deployment algorithms are developed and compared to determine the coverage cavities and to get near-optimal solutions to the QoC maximization problem. The methodologies followed in this paper are based on LS, SA and GA methods. These three metaheuristic algorithms are hybridized as HMA and the HMA outperformed the other pure algorithms with providing up to approximately %19 better QoC results in both smooth and harsh terrains. Overall, the performance results reveal that heuristics especially hybrid algorithms are good candidates for serving as effective methods for sensor deployment on Moon surfaces. The CPU times of the algorithms are acceptable fort hat kind of hard problem. And the sensor-based parameters such as; the number of deployed sensors, the S_{range}, and the P_{range} are affecting the QoC values. For future researches, different hybrid metaheuristics, matheuristics, hyperheuristics can be tried to solve the problem and the solutions can be compared with the ones that this study presents.

REFERENCES

1. Aarts, E., and Korst, J. (1989) *Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing*, Wiley, New York.

2. Abdollahzadeh, S., and Navimipour, N.J. (2016) Deployment strategies in the wireless sensor network: a comprehensive review, *Computer Communications*, 91–92, 1–16. doi:10.1016/j.comcom.2016.06.003

3. Akyildiz, I.F., Su., W., Sankarasubramaniam, Y., and Cayirci, E. (2002) Wireless sensor networks: a survey, *Computer Networks*, 38, 393-422. doi:10.1016/S1389-1286(01)00302-4

4. Bresenham, J.E. (1965) Algorithm for computer control of a digital plotter, *IBM Systems Journal*, 4(1), 25-30. doi:10.1147/sj.41.0025

5. Chakrabarty, K., Iyengar, S.S., Qi, H., and Cho, E. (2002) Grid coverage for surveillance and target location in distributed sensor networks, *IEEE Transactions on Computers*, 51(12), 1448-1453. doi:10.1109/TC.2002.1146711

6. Cheng, L., Wu, C., Zhang, Y., Wu, H., Li, M., and Maple, C. (2012) A survey of localization in wireless sensor network, *International Journal of Distributed Sensor Networks*, 8(12), id. 962523. doi:10.1155/2012/962523

7. Deif, D.S., and Gadallah, Y. (2014) Classification of wireless sensor networks deployment techniques, *IEEE Communications Surveys & Tutorials*, 16(2), 834-855. doi:10.1109/SURV.2013.091213.00018

8. Del Re, E., Pucci, R., and Ronga, L.S. (2009) IEEE802.15.4 wireless sensor network in mars exploration scenario, *in Proc. International Workshop on Satellite and Space Communications (IWSSC)*, Sep. 09-11, Tuscany, Italy, 284-288. doi:10.1109/IWSSC.2009.5286366

9. Dubois, P., Botteron, C., Mitev, V., Menon, C., Farine, P.-A., Dainesi, P., Ionescu, A., and Shea, H. (2009) Ad-hoc wireless sensor networks for exploration of solar-system bodies, *Acta Astronautica*, 64, 626–643. doi:10.1016/j.actaastro.2008.11.012

10. Fan, G.J., and Jin, S.Y. (2010) Coverage problem in wireless sensor network: a survey, *Journal of Networks*, 5(9), 1033-1040. doi:10.4304/jnw.5.9.1033-1040

11. Gaura, E., and Newman, R.M. (2006) Wireless sensor networks: the quest for planetary field sensing, *in Proc. 31st IEEE Conference on Local Computer Networks*, Nov. 14-16, Tampa, FL, USA, 596-603. doi:10.1109/LCN.2006.322021

12. Ghosh, A., and Das, S.K. (2008) Coverage and connectivity issues in wireless sensor networks: a survey, *Pervasive and Mobile Computing*, 4, 303–334. doi:10.1016/j.pmcj.2008.02.001

13. Guerriero, F., Violi, A., Natalizio, E., Loscri, V., and Costanzo, C. (2011) Modelling and solving optimal placement problems in wireless sensor networks, *Applied Mathematical Modelling*, 35, 230‑241. doi:10.1016/j.apm.2010.05.020

14. Guo, H., Ye, H., Liu, G., Dou, C., and Huang, J. (2020) Error analysis of exterior orientation elements on geolocation for a moon-based earth observation optical sensor, *International Journal of Digital Earth*, 13(3), 374-392. doi: 10.1080/17538947.2018.1513088

15. Hamill, P. (2016) Atmospheric observations from the moon: A lunar earth-observatory, *in Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 10-15, Beijing, China, 3719-3722.

16. Holland, J.H. (1975) *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor.

17. Jia, Y., Zou, Y., Ping, J., Xue, C., Yan, J., and Ning, Y. (2018) The scientific objectives and payloads of Chang'E_4 mission, *Planetary and Space Science*, 162, 207–215. doi:10.1016/j.pss.2018.02.011

18. Kulkarni, R.V., Förster, A., and Venayagamoorthy, G.K. (2011) Computational intelligence in wireless sensor networks: a survey, *IEEE Communications Surveys & Tutorials*, 13(1), 68-96. doi:10.1109/SURV.2011.040310.00002

19. Lopez-Matencio, P. (2016) An ACOR-based multi-objective WSN deployment example for lunar surveying, *Sensors*, 16(2), 209. doi:10.3390/s16020209

20. Map, https://www.mapaplanet.org/, Accessed on: Sep. 17, 2019.

21. MATLAB, ver. R2014a.

22. Medina, A., de Negueruela, C., Mollinedo, L., Gandía, F., Barrientos, A., Rossi, C., Sanz, D., Puiatti, A., and Dufour, J.F. (2010) Wireless sensor web for rover planetary exploration, *in Proc. 10th International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS)*, Aug. 29-Sep.01, Sapporo, Japan, 531-537.

23. Mini, S., Udgata, S.K., and Sabat, S.L. (2014) Sensor deployment and scheduling for target coverage problem in wireless sensor networks, *IEEE Sensors Journal*, 14(3), 636-644. doi:10.1109/JSEN.2013.2286332

24. Molina, G., Alba, E., and Talbi, E.-G. (2008) Optimal sensor network layout using multi-objective metaheuristics, *Journal of Universal Computer Science*, 14(15), 2549-2565. doi:10.3217/jucs-014-15-2549

25. Molina, G., and Alba, E. (2011) Location discovery in wireless sensor networks using metaheuristics, *Applied Soft Computing*, 11, 1223–1240. doi:10.1016/j.asoc.2010.02.021

26. NASA, Moon Missions, https://moon.nasa.gov/exploration/moon-missions/, Accessed on: Sep. 17, 2019.

27. Oddi, G., Pietrabissa, A., Liberati, F., Di Giorgio, A., Gambuti, R., Lanna, A., Suraci, V., and Priscoli, F.D. (2017) An any-sink energy-efficient routing protocol in multi-hop wireless sensor networks for planetary exploration, *Int. J. Commun. Syst.*, 30(7), 1-25. doi:10.1002/dac.3020

28. Pabari, J.P., Acharya, Y.B., and Desai, U.B. (2009) Investigation of wireless sensor deployment schemes for in-situ measurement of water ice near lunar south pole, *Sensors & Transducers Journal*, 111(12), 86-105.

29. Pabari, J.P., Acharya, Y.B., Desai, U.B., Merchant, S.N., and Krishna, B.G. (2010) Radio frequency modelling for future wireless sensor network on surface of the moon, *Int. J. Communications, Network and System Sciences*, 3, 395-401. doi:10.4236/ijcns.2010.34050

30. Pabari, J.P., Acharya, Y.B., Desai, U.B., and Merchant, S.N. (2012) Development of impedance-based miniaturized wireless water ice sensor for future planetary applications, *IEEE Transactions on Instrumentation and Measurement*, 61(2), 521-529. doi:10.1109/TIM.2011.2164292

31. Pabari, J.P., Acharya, Y.B., Desai, U.B., and Merchant, S.N. (2013) Concept of wireless sensor network for future in-situ exploration of lunar ice using wireless impedance sensor, *Advances in Space Research*, 52, 321–331. doi:10.1016/j.asr.2012.09.006

32. Parrado-Garcia, F.J., Vales-Alonso, J., and Alcaraz, J.J. (2017) Optimal planning of WSN deployments for in situ lunar surveys, *IEEE Transactions on Aerospace and Electronic Systems*, 53(4), 1866-1879. doi:10.1109/TAES.2017.2674258

33. Prasad, K.D., and Murty, S.V.S. (2011) Wireless sensor networks – a potential tool to probe for water on moon, *Advances in Space Research*, 48, 601–612. doi:10.1016/j.asr.2011.04.004

34. Prasad, K.D., Bhattacharya, A., and Murty, S.V.S. (2012) An ambient light sensing module for wireless sensor networks for planetary exploration, *Planetary and Space Science*, 70, 10–19. doi:10.1016/j.pss.2012.06.012

35. Rodrigues, P., Oliveira, A., Alvarez, F., Cabas, R., Oddi, G., Liberati, F., Vladimirova, T., Zhai, X., Jing, H., and Crosnier, M. (2014) Space wireless sensor networks for planetary exploration: node and network architectures, *in Proc. NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, July 14-17, Leicester, UK, 180-187. doi:10.1109/AHS.2014.6880175

36. Sanz, D., Barrientos, A., Garzon, M., Rossi, C., Mura, M., Puccinelli, D., Puiatti, A., Graziano, M., Medina, A., Mollinedo, L., and de Negueruela, C. (2013) Wireless sensor networks for planetary exploration: experimental assessment of communication and deployment, *Advances in Space Research*, 52, 1029–1046. doi:10.1016/j.asr.2013.06.007

37. Seok, J.-H., Lee, J.-Y., Kim, W., and Lee, J.-J. (2013) A bipopulation-based evolutionary algorithm for solving full area coverage problems, *IEEE Sensors Journal*, 13(12) 4796-4807. doi:10.1109/JSEN.2013.2274693

38. Sun, R., Guo, J., and Gill, E.K.A. (2010) Opportunities and challenges of wireless sensor networks in space, *in Proc. 61st International Astronautical Congress*, Sep. 27 – Oct. 01, Prague, Czech Republic, 1-12.

39. Tsai, C.-W., Tsai, P.-W., Pan, J.-S., and Chao, H.-C. (2015) Metaheuristics for the deployment problem of WSN: a review, *Microprocessors and Microsystems*, 39, 1305–1317. doi:10.1016/j.micpro.2015.07.003

40. Türkoğulları, Y.B., Aras, N., Altınel, İ.K., and Ersoy, C. (2010) An efficient heuristic for placement, scheduling and routing in wireless sensor networks, *Ad Hoc Networks*, 8, 654-667. doi:10.1016/j.adhoc.2010.01.005

41. Ulmer, C., Yalamanchili, S., and Alkalai, L. (2000) Wireless distributed sensor networks for in-situ exploration of mars, *NASA Technical Report*, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.116.324&rep=rep1&type=pdf, Accessed on: Sep. 17, 2019.

42. Wang, H., Guo, Q., Li, A., Liu, G., Guo, H., and Huang, J. (2019) Impact of lunar terrain on moon-based earth observation, *in Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 28-Aug. 02, Yokohama, Japan, 9260-9262.

43. Wang, Q., and Liu, J. (2016) A Chang'e-4 mission concept and vision of future Chinese lunar exploration activities, *Acta Astronautica*, 127, 678–683. doi:10.1016/j.actaastro.2016.06.024

44. Wilson, W.C., and Atkinson, G.M. (2011) Space applications for wireless sensors, *in Proc. NSTI Nanotechnology Conference & Expo (Nanotech)*, June 13-16, Boston, MA, USA, 298-301.

45. Wu, C.-H., Lee, K.-C., and Chung, Y.-C. (2006) A Delaunay triangulation based method for wireless sensor network deployment, *in Proc. 12th International Conference on Parallel and Distributed Systems (ICPADS)*, July 12-15, Minneapolis, USA, 1-8. doi:10.1109/ICPADS.2006.11

46. Ye, H., Guo, H., and Liu, G. (2017) Observation parameters design of moon-based earth observation sensors for monitoring three-polar regions, *in Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 23-28, Fort Worth, Texas, USA, 5755-5758. doi:10.1109/IGARSS.2017.8128315

47. Ye, H., Guo, H., Liu, G., and Ren, Y. (2018a) Observation scope and spatial coverage analysis for earth observation from a Moon-based platform, *International Journal of Remote Sensing*, 39(18), 5809-5833. doi:10.1080/01431161.2017.1395976

48. Ye, H., Guo, H., Liu, G., and Ren, Y. (2018b) Observation duration analysis for Earth surface features from a Moon-based platform, *Advances in Space Research*, 62, 274-287. doi:10.1016/j.asr.2018.04.029

49. Yick, J., Mukherjee, B., and Ghosal, D. (2008) Wireless sensor network survey, *Computer Networks*, 52, 2292-2330. doi:10.1016/j.comnet.2008.04.002

50. Younis, M., and Akkaya, K. (2008) Strategies and techniques for node placement in wireless sensor networks: a survey, *Ad Hoc Networks*, 6, 621–655. doi:10.1016/j.adhoc.2007.05.003

51. Zhai, X., and Vladimirova, T. (2015) Data aggregation in wireless sensor networks for lunar exploration, *in Proc. Sixth International Conference on Emerging Security Technologies*, Sep. 03-05, Braunschweig, Germany, 30-37. doi:10.1109/EST.2015.9

52. Zhai, X., and Vladimirova, T. (2016) Efficient data-processing algorithms for wireless-sensor-networks-based planetary exploration, *Journal of Aerospace Information Systems*, 13(1), 46-66. doi:10.2514/1.I010373

53. Zou, Y., and Chakrabarty, K. (2003) Sensor deployment and target localization based on virtual forces, *in Proc. 22nd Annual Joint Conference of the IEEE Computer and Communication Societies (IEEE INFOCOM)*, Mar. 30 - Apr. 03, San Francisco, USA, 1293-1303. doi:10.1109/INFCOM.2003.1208965