Applied Physics Research; Vol. 7, No. 1; 2015
ISSN 1916-9639 E-ISSN 1916-9647
Published by Canadian Center of Science and Education

The Optimization of Satellite Landing Site based on Particle Swarm Optimization

Zhen Du[1], Yuanbiao Zhang[2], Bo Liu[1] & Zhili Liang[2]

[1] Finance Department of International Bussiness School, Jinan University, Zhuhai 519070, China

[2] Packaging Engineering Institute, Jinan University, Zhuhai 519070, China

Correspondence: Zhen Du, Finance Department of International Bussiness School, Jinan University, Zhuhai City, Guangdong Province, China. Tel: 86-136-3125-9825. E-mail: 864683360@qq.com

Received: November 7, 2014 Accepted: November 24, 2014 Online Published: December 18, 2014

doi:10.5539/apr.v7n1p12 URL: http://dx.doi.org/10.5539/apr.v7n1p12

Abstract

Hazard avoidance is one of the key stages in the satellite's soft landing process, during which the selection of the landing site would have a huge influence on the satellite safe landing. In this paper, a new design method is presented for determining the landing site selection using Particle Swarm Optimization based on the Moon's grounds image taken by moon satellite. Applying this method, the problem of hazard avoidance can be converted into the optimization of the value of "obstacle function" after processing the image employing the median filter algorithm. In order to discuss and to compare the efficiency of the different optimization methods, three algorithms including Particle Swarm Optimization, Genetic Algorithm and Global Search Algorithm are tested to solve this problem. The results show that Particle Swarm Optimization has the more satisfactory optimization results and the quickest optimization speed in the landing site selection. The detailed comparisons are introduced in the body paragraph.

Keywords: genetic algorithm, hazard avoidance, landing site selection, particle swarm optimization, satellite

1. Introduction

The achievement of lunar exploration is a massive progress in aerospace science and human technology development. In the lunar exploration, hazard avoidance plays a significant role in the satellite soft landing process, during which landing zone selection could have a huge impact on the satellite safe landing. As a consequence, in this paper, a novel design method is presented for determining the landing site selection using Particle Swarm Optimization (PSO). It is the first time that PSO is applied for satellite landing site selection. The obstacle function, which can be found in the second part of 2.1, is constructed based on the Moon's ground image taken by moon satellite.

The Optimization of Satellite Landing Site has been a popular topic since the launch of the moon satellite. At first, manual control was used to avoid obstacles in the Apollo series (Lyndon, 1975). However, the manual control method was not the optimal one to select landing site because the satellite could have a higher risk to crash during the landing. In order to lower this risk, the Mars Phoenix Lander used the instruments of landing ellipse predicts and hazard maps to select the high probability of safe landing region and successfully realized the method of soft landing (Bonfiglio et al., 2011). This method reduces the risks of crash during the period of landing. But the method mentioned above is based on the subjective judgment of human beings, so it will be extremely hard to reach an optimal landing sites based on this principle. Chang'E-3 is the satellite launched by china in the Xichang Satellite Launch Center on December 2, 2013 and it first implements the autonomous hazard avoidance during the period of soft landing using the image data collected by the satellite. H.H. Zhang et al. introduced the technology of "relay hazard avoidance control method" used in Chang'E-3 (Zhang, Liang, & Huang, 2014). NASA made a project in order to send humans back to the moon. This plan is called Autonomous precision Landing Hazard Avoidance Technology (ALHAT) project, in which they developed a new hardware called Guidance, Navigation and Control (GNC) (Epp, & Smith, 2007). EC Wong et al. introduced a method which called the autonomous six degrees-of-freedom controls and this method is used to ensure successful Mars landing and accomplished using engines and thrusters (Wong, Masciarelli, & Singh, 2002). Johnson A E et al. gave an introduction of the lidar-based hazard avoidance. This process is realized by utilizing a synthetic Martian terrain generator, a scanning lidar model and powered landing guidance algorithms (Johnson, Klumpp, Collier, & Wolf, 2002). Matthies L et al. identified an approach based on stereo vision and shadow analysis and addressed broadest set of the NSA New

Millennium Program (NMP) ST9 mission (Matthies, Huertas, Cheng, & Johnson, 2007). Few papers refer to the era of the optimal problem. As a result, this paper creatively combines the optimization algorithm with the landing sites selection, comparing the efficiency of PSO, Genetic Algorithm (GA) and Global Search Algorithm (GSA), and gives the best algorithm based on the data of the image of lunar.

In this paper, some research has been done to solve the problem of landing site selection in the process of satellite soft landing based on the Moon's ground image taken by the satellite. Firstly, we denoised the image using median filter algorithm and extracted the pixel data of the image as the input data for the following step. Secondly, we defined a risk factor function as the objective function in the next optimization process. The risk factor function gives a risk value to a square area whose center is the landing zone center. At last, we did the optimization to solve the problem and we compared the different results of PSO, GA and GSA. In Sec. 2 we present the methods we applied; they are used in Sec. 3; we conclude in Sec. 3 discussing our results and analyses.

2. The building of the Optimal Models

2.1 The Optimization Model of Satellite Landing Site

2.1.1 The Principles to Select Landing Site

Landing site selection is the most significant period of the soft landing. Several conditions should be considered to ensure safety in the process of satellite landing.

1) The height and depth of barriers (including the raised stones and the sunken valleys) should meet certain requirements to reduce the danger.
2) The terrain of the chosen region should be flat as much as possible, that is to say, the standard deviation of the height value in the chosen area should be small as much as possible to prevent satellite from tumbling.
3) The area of the chosen region should be bigger than the flat area the satellite occupies. This paper uses the data of Chang'E-3satellite whose flat area is 27 m^2 according to related literature. So the minimum unit in the search procedure can be chosen initially as a square area of side 6m.

2.1.2 The Design of Risk Factor Function

Based on the principles in 2.1.1, risk factor function was designed as the objective function in this paper. The objective function expressed as $\emptyset(\mathrm{x},\mathrm{y})$ can be divided into two parts: the first one based on the condition 1 in 2.1.1 represents the risk value due to the height of every point in the region and it's expressed as $\emptyset_1(\mathrm{x},\mathrm{y})$; the second one based on the condition 2 in 2.1.1 represents the risk value due to the unevenness of terrain in the region and it's expressed as $\emptyset_2(\mathrm{x},\mathrm{y})$.

$\emptyset_1(\mathrm{x},\mathrm{y})$ can be calculated using the formulas below (Sun, 2006):

$$\phi_1(\mathrm{x},\mathrm{y}) = \sum_{i=1}^{n} k_1 \left|z_i - \overline{z}\right| e^{\left(-\frac{(\mathrm{x}-\mathrm{x}_i)^2+(\mathrm{y}-\mathrm{y}_i)^2}{\sigma^2}\right)} \qquad (1)$$

In Equation (1) *x* and *y* are the coordinate of Chang'E-3. x_i and y_i are the coordinate of obstacles. z_i is the height or depth of obstacles. $\overline{Z}$ is the mean value of the terrain in the region. K_1 and σ are constants， i is the index of the total number of obstacles *n, n* is the total number of obstacles. This function represents the risk value due to the height of every point in the region. If the function value is bigger, the region selected is risker and vice versa.

$\emptyset_2(\mathrm{x},\mathrm{y})$ can be calculated using the formulas below:

$$\phi_2(x,y) = \sqrt{\frac{\sum_{i=1}^{n}(z_i - \overline{z})^2}{n}} \qquad (2)$$

That is, the unevenness of terrain in this region can be evaluated by the standard deviation of the altitude of this area.

The objective function $\emptyset(\mathrm{x},\mathrm{y})$ can be calculated using the formulas below with the help of using both $\emptyset_1(\mathrm{x},\mathrm{y})$ and $\emptyset_2(\mathrm{x},\mathrm{y})$:

$$\phi(x,y) = k_2 \times \phi_1(x,y) \times \phi_2(x,y) \qquad (3)$$

In Equation (3) $k_2$is constant. The function value calculated represents the risk factor value of one region and the region's center is the landing zone center.

2.1.3 The Optimization Model for Satellite Landing Site

The mathematical model of this problem is as follows:

$$\begin{aligned} &\min \ \phi(x,y) \\ &st \quad 30 \le x \le 970 \\ &\qquad 30 \le y \le 970 \\ &\qquad x, y \in N^* \end{aligned}$$

Where, x and y is the abscissa and ordinate of landing site in the coordinate. x and y are positive integer because the data are discrete point after image processing.

2.2 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is one of the efficient global optimization algorithms and the modern heuristic algorithms by using the cooperation and coordination mechanism between the particles. This algorithm has many advantages including parallel processing, easy implementation, good robustness, stable convergence characteristic and nice computational efficiency. The algorithm inspired by the simulation about swarming or collaborative behavior of the birds flying process is widely used in the complex optimization problems of many areas.

In each process of iteration the particle is updated by tracking two end points: the first one called individual optimal point（using $\mathrm{P}_{i,j}^{k}$ to denote its position）is the best solution for the particle; the second one called global optimal point（using $\mathrm{G}_{i,j}^{k}$ to express its position）is the best solution for the whole population of particle at the current iteration. In each iteration, the velocity and position of each particle can be modified using the current velocity and the distances as shown in the following equations (García-Villoria & Pastor, 2009):

$$\mathrm{v}_{i,\mathrm{j}}^{k+1} = \omega \mathrm{v}_{i,j}^{k} + c_1 r_1 (\mathrm{P}_{\mathrm{i},j}^{k} - s_{i,j}^{k}) + c_2 r_2 (\mathrm{G}_{i,j}^{k} - s_{i,j}^{k})$$

$$s_{i,j}^{k+1} = s_{i,j}^{k} + \mathrm{v}_{i,\mathrm{j}}^{k+1}$$

Where, k is the iteration index; ω is the inertia weight; c_1 and c_2 are commonly known as acceleration factors. c_1 and c_2 are cognitive and social factor respectively; r_1 and r_2 are random number between 0 and 1.

The fitness of all particles will be evaluated by the optimization function f(x). The aim of the PSO algorithm is to find the value of the $x_{i,j}^{k+1}$ position which minimizes the value of the risk function f(x); therefore, individual optimal point $\mathrm{P}_{i,j}^{k}$ can be updated until the value which minimizes the function is found according to the following formula based on the value of $\mathrm{f}(\mathrm{P}_{i,j}^{k})$:

$$\mathrm{P}_{\mathrm{i},j}^{k+1} = \begin{cases} \mathrm{x}_{i,j}^{k+1} & \textit{if } f(\mathrm{x}_{i,j}^{k+1}) < f(\mathrm{P}_{\mathrm{i},j}^{k}) \\ \mathrm{P}_{\mathrm{i},j}^{k} & \textit{otherwise} \end{cases}$$

Global optimal point $\mathrm{G}_{i,j}^{k}$ is updated by finding the best fitness in all individual optimal points.

2.3 Median Filter Algorithm

Median filter algorithm can not only denoise the image, but also can keep the image details. The main steps are as follows (Sadi, Lee, & Hasan, 2010):

*step*1: Move the template in the image and coincide the center of the template with one of the pixel location in the image;

*step*2: Read gray value of every corresponding pixel in the template;

*step*3: Line up all these gray values from smallest to largest;

*step*4: Find the middle one in these values;

*step*5: Assign the value of the middle one to the pixel in the center of the template.

The *medfilt*2 function, already implemented in *MATLAB* is used to calculate the two dimensional median filtering of a give image.

3. Results and Analysis

In this paper, the object we studied is included in the problem of references (Jin, 2008). We use the digital elevation picture which Chang'E-3 photographed above the Moon at the distance of 100m away from the ground in the annex 4 in the reference (CUCM, 2014). The horizontal resolution of the digital elevation picture is 0.1m per pixel. The optimization problem is solved using PSO, GA and GSA in order to compare optimization results easily.

The data can be used in the PSO, GA and GSA after filtering the image by using median filtering algorithm.

In order to analyze the image, a relative system of coordinates should be defined. We defined it from our analyzed image which has 1000×1000 pixels. The origin point of coordinate is the left bottom edge of image, the x-axis is plotted along the bottom edge of the image and the y-axis is plotted along the left edge of the image. We assumed that: Chang'E-3 is located in the center of the image, that is, (500,500) in our reference system. The minimum search unit is 60×60, that is, the search area is 36 m^2. The search process begins from the point （30,30） and ends to the point (970,970) and the step size is one. In each point, the function value of each point is calculated according to the method in 2.1.2 and the smallest point can be found using optimization algorithm.

Then we solve the problem in the programming environment of *MATLAB*. The parameters of PSO are set as follows: the inertia weight ω=0.9, the maximum number of iteration equals to 150, termination error is 0.0001, that is to say, if the difference value between two iteration results is less than 0.0001, PSO can terminate at that point.

By testing the final results, the parameters of optimization function are set as follows: k_1 =0.2, k_2=1000, σ=2. These parameters do not have a great influence on the result, so their values do not matter.

3.1 The Optimization Results of PSO

The optimization results are unstable when the search radius of PSO is small, which indicates the existence of several local near-optimal solutions. However, the optimization results are stable and the optimal positions tend to a point nearby when the search radius of PSO is large enough, for example, if the search radius increases from 30 to 40. The parameters of PSO are set as follows: the population size N=80 and the social factor c_2 = 1.5.

Table 1. the optimization results of PSO

search radius	minimum unit of search（m2）	iterations	C1	error precision （le-n）	optimal value	optimal position	optimization time (s)	optimization degree of stability
30	36	80	0.5	4	0.2620	（149,699）	14.26	stable
30	36	80	0.5	5	0.2620	（149,699）	10.33	instable
30	36	40	0.5	4	0.26-0.28	（140-160,699-720）	3	stable
40	64	80	1.5	4	0.2781	（149,699）	12-13	stable
40	64	40	0.5	4	0.27-0.28	（149,699-710）	4	more stable

The optimization results are unstable when cognitive factor c_1 is small, which indicates that there are several local near-optimal solutions. However, the optimization results tend to the point （149,699 when c_1 is large enough, for example, if c_1 increases to 1.5.

The parameters of PSO are set as follows: population size N=80, iterations=80, social factor c_2= 1.5, error precision=10^{-4}, search radius=40m, minimum unit of search=64 m^2.

Table 2. the results of different cognitive factor

C1	optimal value	optimal position	optimization time (s)	optimization degree of stability
1.5	0.2781	（149,699）	12-13	stable
1	0.2781	（149,699）and（915,143）	14	stable
0.8	0.27- 0.29	（149,699）and（915,143）	10-12	stable
0.5	0.27-0.34	（600-700,600-700）	13	more stable

3.2 The Optimization Results of GA

We solve the problem employing GA in the programming environment of *MATLAB*. Several parameters have a significant effect on the optimization results by debugging various parameters.

The mutation probability usually assigns from 0.001 to 0.1 according to the reference (Jin, 2008). In the initial search of this paper, it's found that with GA is difficult to find the global optimum since it's usually trapped into local optimum when the mutation probability is small. Best fitness from 0.25 to 0.3 can be found when the mutation probability is greater than 0.4.

The parameters of GA are set as follows: population size *N*=400, iterations=50, crossover probability = 0.4, error precision=10^{-4}, search radius=30m, minimum unit of search=36 m^2.

Table 3. the results of different mutation probability

mutation probability	optimal value	optimal position	optimization time (s)	optimization degree of stability
0.8	0.2620	(149,699)	37	instable
0.6	0.2661	(797,69)	37	stable
0.4	0.28-0.32	instable	22	stable
0.2	0.31-0.34	instable	22	stable
0.1	0.31-0.32	(60,439)	9.79	instable

The population size has a big influence on the results when it is less than 200; in fact, while is difficult to find the global optimum for a population size less than 200 it is easier when is greater than 200. But, unfortunately, the running time of program increases without improving significantly the optimization results.

The parameters of GA are set as follows: iterations=50, crossover probability = 0.4, mutation probability =0.6, error precision=10^{-4}, search radius=30m, minimum unit of search=36 m^2.

Table 4. the results of different population size

Population size	optimal value	optimal position	optimization time (s)	optimization degree of stability
400	0.2661	（797,69）	37	stable
300	0.26-0.28	（127,695）	17	stable
200	0.26-0.29	instable	11	instable
150	0.27-0.31	instable	15	instable
100	0.36-0.37	instable	10.2	instable
80	0.30-0.40	instable	8	instable
50	0.36-0.40	instable	instable	instable

3.3 The Comparison of the Results Between PSO and GA

Table 5. the result comparison between PSO and GA

	optimization time (s)	optimal result	optimization degree of stability
PSO	3-15	global optimum	stable
GA	10-35	easily trapped into local optimum	instable when population size is smaller

The table 5 shows that PSO has a fast optimization time, so we can use it in the practical problem. The GA has a weakness that it easily trapped into local optimum and can't find the satisfactory solution in this problem. What's more, the results are not stable with GA and as comparison, PSO has a stable solution.

3.4 The Optimization Results of GSA

The final result, i.e. the safest landing region, is a region whose area is 36 $m^2$and whose center is (119,669).

The risk value is 0.2620 and the running time is 574 seconds. This algorithm has a high precision and the function value calculated is accurate, but this method has an obvious weakness since the running time is too long to be used for practical purposes.

In Figure 1 we plotted the risk function calculated for each pair of (x, y) coordinates.

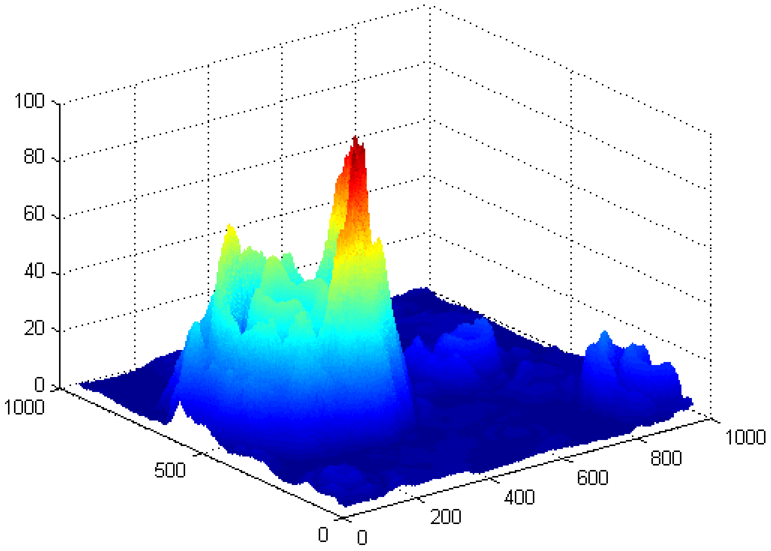

Figure1. Risk function picture resulting from the GSA

4. Conclusion

Compared with GA and GSA, PSO has the following several characteristic : PSO has advantages of small quantity of calculations; the probability of convergence to the local optimal solution decreases because particle swarm divide into several subgroup and the subgroup exist overlaps between each other. In this paper, the problem studied is about landing site selection during the hazard avoidance in the satellite soft landing process. The problem is successfully solved using both PSO and GA and the results is compared and further researched. Based on the Moon’s ground image taken by the satellite, the problem can be converted to the optimization problem by designing the risk factor function after median filtering the image. Overall, the results show that: both PSO and GA algorithms can be used in optimizing the landing site selection, PSO has a quickest rate and a better solution in the optimization process, GA has a poor result and the GSA provides has get an accurate result but the optimization time is too long for practical purposes. The results of this paper offer a powerful safeguard for satellite soft landing process and give a strong support for further research.

References

Bonfiglio, E. P., Adams, D., Craig, L., Spencer, D., Arvidson, R., & Heet, T. (2011). Landing-Site Dispersion Analysis and Statistical Assessment for the Mars Phoenix Lander. *Journal of Spacecraft and Rockets*, *48*(5), 784-797. http://dx.doi.org/10.2514/6.2008-7348

CUCM. (2014). CUMCM-2014 contest problems. Retrieved from http://www.mcm.edu.cn/problem/2014/cumcm2014problemsE.rar

Epp, C. D., & Smith, T. B. (2007, March). Autonomous precision landing and hazard detection and avoidance technology (ALHAT). In *Aerospace Conference, 2007 IEEE* (pp. 1-7). IEEE. http://dx.doi.org/10.2514/2.4988

García-Villoria, A., & Pastor, R. (2009). Introducing dynamic diversity into a discrete particle swarm optimization. *Computers & Operations Research*, *36*(3), 951-966. http://dx.doi.org/10.1016/j.cor.2007.12.001

Jin, F. (2008). *Application Research of Genetic Algorithm in Function Optimization*. Soochow University.

Johnson, A. E., Klumpp, A. R., Collier, J. B., & Wolf, A. A. (2002). Lidar-based hazard avoidance for safe landing on Mars. *Journal of guidance, control, and dynamics*, *25*(6), 1091-1099. http://dx.doi.org/10.2514/2.4988

Lyndon, B. (1975). Apollo Program Summary Report. *Johnson Space Center, NASA, TM X*, *68725*.

Matthies, L., Huertas, A., Cheng, Y., & Johnson, A. (2007). Landing hazard detection with stereo vision and shadow analysis. *AIAA Infotech@ Aerospace, Rohnert Park, CA*. http://dx.doi.org/10.2514/6.2007-2835

Sadi, F., Lee, S. Y., & Hasan, M. K. (2010). Removal of ring artifacts in computed tomographic imaging using iterative center weighted median filter. *Computers in biology and medicine*, *40*(1), 109-118. http://dx.doi.org/10.1016/j.compbiomed.2009.11.007

Sun, J. W. (2006). *Algorithm research on autonomous navigation and control of lunar explorer*. Harbin Institute of Technology.

Wong, E. C., Masciarelli, J., & Singh, G. (2002, August). Autonomous guidance and control design for hazard avoidance and safe landing on Mars. In *AIAA Atmospheric Flight Mechanics Conference and Exhibit* (pp. 5-8). http://dx.doi.org/10.2514/6.2002-4619

Zhang, H. H., Liang, J., & Huang, X.Y. (2014). Autonomous hazard avoidance control for Chang'E-3 soft landing (in Chinese). *Sci Sin Tech, 44*, 559–568. http://dx.doi.org/10.1360/092014-51

Copyrights