Intelligent Decision-Making Method for On-Orbit Service with Multiple Geosynchronous Earth Orbit Targets Based on Reinforcement Learning

Hongfeng He, Xiaofang Chen, and Zeyang Yin(✉)

School of Automation, Central South University, Changsha 410083, China
yinzeyang@csu.edu.cn

Abstract. For the decision-making of on-orbit service with multiple geosynchronous earth orbit targets, an offline and online combined intelligent decision-making method is proposed based on reinforcement learning. Firstly, the decision-making problem is given and formulated. Then, considering the computational complexity of online decision-making, this work proposed an offline and online combined intelligent decision-making framework. A cost function related to the fuel consumption and rendezvous time is constructed offline for one spacecraft serving one target. And a RBF neural network-based method is proposed to approximate and fit the offline calculated data. For the on-orbit real-time decision-making problem, a multi-target decision-making cost function is constructed, and a low-complexity and intelligent decision-making method is proposed based on reinforcement learning to allocate the proper spacecrafts to serve the multiple targets. Simulation results show that the proposed method can achieve fast and accurate online decision-making for the service of geosynchronous earth orbit targets.

Keywords: On-orbit service · Geosynchronous earth orbit · Decision-making · Reinforcement learning · RBF neural network

1 Introduction

With the rapid development of space technology, on-orbit service mission has attracted more and more attention in recent years [1,2]. As there are many high-value satellites in the geosynchronous earth orbit (GEO), the on-orbit service of GEO targets is strategic and important [3,4]. In order to raise the efficiency of serving multiple GEO targets, selecting and employing several on-onbit service spacecrafts (servers for brevity) to simultaneously carry out the service tasks is an effective way. The decision-making problem of on-orbit service with multiple GEO targets is how to select enough and proper servers, which is a task allocation problem considering the orbit dynamics.

Existing task allocation methods usually establish a comprehensive optimization goal by considering a variety of factors. In order to solve the established

Z. Ren et al. (Eds.): *Proceedings of 2021 5th Chinese Conference on Swarm Intelligence and Cooperative Control*, LNEE 934, pp. 733–741, 2023.
https://doi.org/10.1007/978-981-19-3998-3_70

optimization problem, some methods have been proposed, *e.g.*, the bundle algorithm [5,6], particle swarm optimization (PSO)-based method [7], genetic algorithm (GA)-based method [8,9]. The bundle algorithm mainly deals with the "one-to-many" decision-making problems and is not suitable for the "many-to-many" problem in this work. Have shortcomings in computational efficiency and global optimality, the PSO- and GA-based methods have difficulties in real-time decision-making problem.

Motivated by the foregoing analyses, this work proposes an offline and online combined intelligent decision-making method. The decision-making problem is given and formulated in Sect. 2. The cost function of "one-to-one" service is constructed offline in Sect. 3, and a RBF neural network (RBFNN)-based method is proposed to fit the offline calculated data. A reinforcement learning-based method for online decision-making is proposed in Sect. 4. The proposed method is verified by simulations in Sect. 5. Some conclusions are finally drawn in Sect. 6.

2 Problem Formulation

In order to serve the passive GEO targets in time, assume that the servers initially move on an equatorial circular orbit with different phase angles. Suppose there exist m servers, then the orbit dynamics of the m servers is given as [10]

$$\begin{cases} \dot{\boldsymbol{r}}_{Si} = \boldsymbol{v}_{Si} \\ \dot{\boldsymbol{v}}_{Si} = -\frac{\mu}{\|\boldsymbol{r}_{Si}\|^3}\boldsymbol{r}_{Si} + \boldsymbol{u}_{Si} \end{cases}, i = 1, \cdots, m \tag{1}$$

where $\boldsymbol{r}_{Si}$ and $\boldsymbol{v}_{Si}$ are the position and velocity vectors of the i-th server in earth centered inertial frame, $\boldsymbol{u}_{Si}$ is the control input, and μ is the earth gravitational coefficient.

The orbit dynamics of the n GEO targets is given as

$$\begin{cases} \dot{\boldsymbol{r}}_{Tj} = \boldsymbol{v}_{Tj} \\ \dot{\boldsymbol{v}}_{Tj} = -\frac{\mu}{\|\boldsymbol{r}_{Tj}\|^3}\boldsymbol{r}_{Tj} \end{cases}, j = 1, \cdots, n \tag{2}$$

where $\boldsymbol{r}_{Tj}$ and $\boldsymbol{v}_{Tj}$ are the position and velocity vectors of the j-th targets.

An assumption has been made that $m > n$ to ensure that there are enough servers to be selected to simultaneously serve the GEO targets. Then the decision-making problem can be stated that how to select n servers and plan their trajectories so that all the GEO targets can be served.

3 Offline Orbit Optimization and Fitting Based on RBFNN

There are countless different transfer orbits for one server to rendezvous with the specific GEO target. The fuel consumption and rendezvous time are also different. From Eqs. (1) and (2), it can be seen that a lot of calculations are required to find the optimal path for a server to rendezvous with the target. If the rendezvous trajectory is planned online, the decision-making process may not be finished in time. For this problem, this work proposes an offline and online combined decision-making method to solve the multi-target decision-making problem. The time-consuming orbit optimization design process for "one-to-one" on-orbit service is carried out offline, and an efficient intelligent method based on reinforcement learning is proposed for online decision-making.

3.1 Transfer Orbit Designed by Solving Lambert Problem

The most fuel-efficient transfer orbit between coplanar circles is the Hohmann orbit, but the transfer time is always long. In order to improve the rendezvous efficiency, a series of two-pulse transfer orbits are considered by solving the Lambert problem.

Define the initial and rendezvous time for server i to target j as t_{ij1} and t_{ij2}, respectively, then a transfer orbit for server i is expected to reach the target j when time $t = t_{ij2}$. Based on Eqs. (1) and (2), the initial and final states of server i are $\boldsymbol{r}_{Si}(t_{ij1})$, $\boldsymbol{v}_{Si}(t_{ij1})$ and $\boldsymbol{r}_{Tj}(t_{ij2})$, $\boldsymbol{v}_{Tj}(t_{ij2})$. The rendezvous time can be calculated as $\Delta t_{ij} = (t_{ij2} - t_{ij1})$. For a specific rendezvous time $\Delta t_{ij} \in (t_0, t_f)$ with $t_f > t_0 > 0$, the transfer orbit can be designed by solving the Lambert problem [10], and the control vectors $\boldsymbol{u}_{Sij}(t_{ij1})$ and $\boldsymbol{u}_{Sij}(t_{ij2})$ are then obtained.

3.2 Transfer Orbit Optimization

There are many two-pulse transfer orbits by solving the Lambert problem when $\Delta t_{ij} \in (t_0, t_f)$. In order to find the optimal transfer orbit for a specific mission, a cost function is expected by considering different performance indicators. In this work, the energy and time consumption are considered. Notice that the servers and targets are moving on a same plane, therefore the energy consumption u_{ij} for the server i to target j can be defined as

$$u_{ij} = \|\boldsymbol{u}_{Sij}(t_{ij1})\| + \|\boldsymbol{u}_{Sij}(t_{ij2})\|, \tag{3}$$

is only related to the relative phrase angle $\Delta\theta_{ij}$ and rendezvous time Δt_{ij}. As a result, a transfer orbit optimization problem has been established by considering u_{ij} and Δt_{ij}, that is

$$J_{ij}^* = \min_{\Delta t_{ij}} J_{ij} = CJ_{u,ij} + (1 - C)J_{t,ij} \tag{4}$$

where J_{ij} is the cost function, J_{ij}^* is the optimal cost function, $C \in [0, 1]$ is the proportional coefficient which reflects the relative importance of energy and time consumption, $J_{u,ij}$ and $J_{t,ij}$ are defined as

$$J_{u,ij} = (u_{ij} - u_{\min}) / (u_{\max} - u_{\min}) \tag{5}$$

$$J_{t,ij} = (\Delta t_{ij} - t_0) / (t_f - t_0) \tag{6}$$

wherein $u_{\min}$ and $u_{\max}$ are constants for normalization.

For a specific relative phrase angle $\varDelta\theta_{ij}$ and any $\varDelta t_{ij} \in [t_0, t_f]$, the cost function $J_{ij}\left(\varDelta t_{ij}\right)$ can be obtained by Eqs. (3)–(6). In order to obtain the minimum value J_{ij}^*, enough different $\varDelta t_{ij}$ are sampled within interval $[t_0, t_f]$ and the cubic spline curve is used to fit the discrete $\varDelta t_{ij}$. As the cubic spline curve is analytical, the optimal cost function J_{ij}^* and the optimal rendezvous time $\varDelta t_{ij}{}^*$ are obtained by analytically calculating the minimum point of the cubic spline curve.

3.3 Optimal Transfer Orbit Fitting by RBFNN

The optimal transfer orbit for a specific relative phrase angle $\varDelta\theta_{ij}$ has been obtained in Sect. 3.2. Actually, any $\varDelta\theta_{ij} \in [0, 360]^\circ$ may appear in practical missions. In order to cover all conditions, enough different $\varDelta\theta_{ij}$ are sampled and the corresponding optimal cost functions are calculated offline. To avoid storing too many data in the servers, a RBFNN is constructed to approximate and fit the discrete data due to its excellent nonlinear approximation capability.

The input data of the RBFNN are designed as the normalized values of $\varDelta\theta_{ij}$, which is defined as $\varDelta\theta_{Nij} = \varDelta\theta_{ij}/360$. The output data of the RBFNN are set as the corresponding optimal cost functions J_{ij}^* and $J_{t,ij}^*$. The employed RBFNN is a forward-type network composed of three layers. The state of the first layer, that is the input layer, is $x = \varDelta\theta_{Nij} \in [0, 1]$. The second layer is the hidden layer, whose state $\boldsymbol{h}\left(x\right) = \left[h_1\left(x\right), \cdots, h_p\left(x\right)\right]^T$ is defined as

$$h_k\left(x\right) = \exp\left(-\frac{\left|x - c_k\right|^2}{2b_k^2}\right), k = 1, \cdots, p \tag{7}$$

where c_k, b_k are the Gaussian function center and width, respectively, of the k-th node. The state of the output layer $\boldsymbol{y} = \left[y_1, y_2\right]^T$ is defined as

$$y_l = \sum_{k=1}^{p} w_{pl} h_p\left(x\right), l = 1, 2 \tag{8}$$

where w_{pl} is the weights to be trained. By using the input data $\varDelta\theta_{Nij}$ and the output data J_{ij}^* and $J_{t,ij}^*$, the employed RBFNN is trained to fit the data with an adjustable admissible error.

4 Intelligent Online Decision-Making Based on Reinforcement Learning

With the trained RBFNN in Sect. 3, the optimal cost function J_{ij}^* and rendezvous time $\varDelta t_{ij}{}^*$ can be easily obtained online for any server i and target j. In this section, an online cost function for the decision-making of multiple-target task allocation is established and solved by reinforcement learning.

4.1 Online Decision-Making Problem Formulation

In order to clearly describe the online decision-making problem multiple-target task allocation, α_{ij} is defined as

$$\alpha_{ij} = \begin{cases} 1, \text{ if server } i \text{ serves target } j \\ 0, \qquad\quad \text{otherwise} \end{cases} \tag{9}$$

Then the online decision-making optimization problem is formulated as

$$J^*\left(\alpha_{ij}\right) = \min_{\alpha_{ij}} J\left(\alpha_{ij}\right) = \min_{\alpha_{ij}} \sum_{i=1}^{m} \sum_{j=1}^{n} \alpha_{ij} J_{ij} \tag{10}$$

where $J\left(\alpha_{ij}\right)$ is the online decision-making cost function. The state variables in optimization problem (10) should satisfy that

$$\begin{cases} \sum\limits_{i=1}^{m} \alpha_{ij} = 1, j = 1, ..., n \\ \sum\limits_{j=1}^{n} \alpha_{ij} \leq 1, i = 1, ...m \end{cases} \tag{11}$$

The physical meaning of the constraints in Eq. (11) is that each GEO target should be served with one service spacecraft, and each service spacecraft serves one target at most.

4.2 Online Decision-Making Optimization Based on Q-learning

The online decision-making optimization equation in Eq. (10) is nonlinear, which is difficult to be analytically solved. In this work, a reinforcement learning-based method is proposed to achieve the multi-target online decision-making. Reinforcement learning is an iterative optimization method, including value iteration and strategy iteration [11]. Wherein, Q-learning is the most commonly used value function iterative update algorithm for reinforcement learning [12].

For the online decision-making optimization problem in this work, a Q-Learning algorithm-based method is proposed. After acquiring relative phrase angles and calculating the optimal “one-to-one” cost function by RBFNN, the online decision-making process is implemented through the steps shown in Fig. 1. Firstly, build a $q_1 \times q_2$ Q table, where q_1 is the number of states, q_2 is the number of the actions, and initialize the elements in table to zero. Then calculate the cost function of the current allocation matrix (Q value) and judge whether the Q value is the optimal solution. If not, a new action is selected by combining the current Q value and the ε-greedy learning strategy [13]. By calculating the cost function to carry out the decision, Q table is updated by the Q-learning algorithm as

$$Q(s, a) \leftarrow Q(s, a) + \mu \left[R + \gamma \max Q(s', a) - Q(s, a)\right] \tag{12}$$

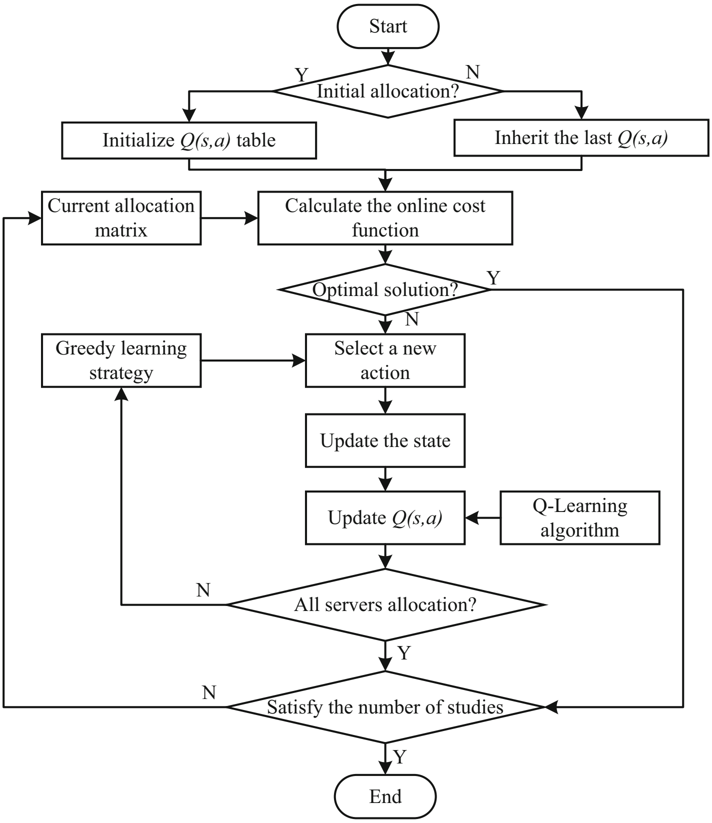

Fig. 1. Q-learning algorithm-based intelligent decision-making method

where $\mu \in (0,1)$ is the learning rate, $\gamma \in (0,1)$ is the discount coefficient, R is the reward by performing the current action a, which will be designed afterwards, s and s' are the current and next states, respectively. After repeated iterations to update the Q table, a good decision is learned to solve the optimization problem.

The core steps in the Q-learning algorithm-based decision-making method is presented as follows.

Action Space Design. Each element in Q table is corresponding to a allocation matrix, which is called a action. The allocation matrix $\boldsymbol{A}$ is $m \times n$ matrix and the i-th row and j-th column element of $\boldsymbol{A}$ is α_{ij}.

State Space Design. The state space in this problem is designed as a group of the online cost function corresponding to the allocation matrices. By defining the online cost function of the l-th allocation matrix as J_l, the upper and lower bound of the state space can be defined as $\max J_l$ and $\min J_l$ $(l = 1, \cdots q_2)$. The state space is then obtained by uniformly discretizing the space $[\min J_l, \max J_l]$.

Reward Function Design. The quantified reward function is the core to judge the performance of one action. Reward function in this work is designed as

$$R = \begin{cases} -10, & \sum_{i=1}^{m} \alpha_{ij} \neq 1, j = 1, ..., n \\ -10, & \sum_{j=1}^{n} \alpha_{ij} > 1, i = 1, ..., m \\ -J, & \text{otherwise} \end{cases} \tag{13}$$

When a certain action satisfies all the constraints in Eq. (11), the reward function is the actual online cost function. Otherwise, a negative reward −10 is applied.

5 Simulation Analyse

5.1 Offline Orbit Optimization and Fitting

In Sect. 3, a RBFNN is constructed and trained to approximate and fit the discrete optimal cost functions. The input and output data of the RBFNN are $\Delta\theta_{Nij}$ and J_{ij}^*, $J_{t,ij}^*$. The original data and the fitting curves are presented in Figs. 2 and 3. From the figures we can see that the employed RBFNN is capable of accurately fitting the original data.

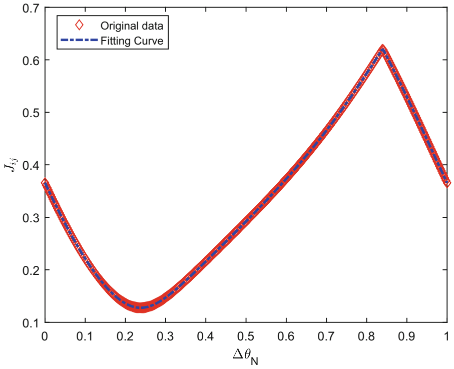

Fig. 2. Fitting curve of J_{ij}^*

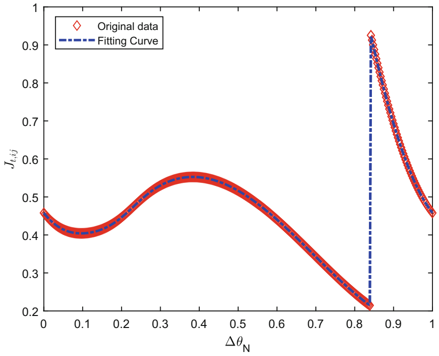

Fig. 3. Fitting curve of $J_{t,ij}^*$

5.2 Online Intelligent Decision-Making

A numerical simulation is designed to verify the effective of the proposed online intelligent decision-making method based on reinforcement learning. Suppose there are three targets moving on GEO with the true anomaly as 86.1°, 179.0°, 306.5°, respectively. The service spacecrafts are moving on a circular orbit in the same plane. The altitude of the servers is 6007 km, and suppose there are four servers with the true anomaly as 8°, 128°, 199.9°, 272.3°, respectively. The offline cost functions J^*_{ij} and $J^*_{t,ij}$ for every server i to target j can be directly obtained by the constructed RBFNN in Sect. 5.1 and given in Table 1.

Table 1. Optimal cost functions J^*_{ij} and $J^*_{t,ij}$ for every server i to target j

	Target 1	Target 2	Target 3
Server 1	0.1303&0.4611	0.2730&0.5296	0.6082&0.2220
Server 2	0.5555&0.7516	0.1768&0.4118	0.2891&0.5187
Server 3	0.4492&0.3591	0.4617&0.5689	0.1453&0.5281
Server 4	0.2793&0.5255	05085&0.3010	0.2280&0.4041

Set the proportional coefficient in Eq. (6) as $C = 0.7$. For the proposed reinforcement learning-based method, discount factor $\gamma = 0.9$, learning rate $\mu = 0.1$. Based on the proposed reinforcement learning method, the decision has been made and the optimal allocation matrix is given as

$$\boldsymbol{A} = \begin{bmatrix} 1\,0\,0 \\ 0\,1\,0 \\ 0\,0\,1 \\ 0\,0\,0 \end{bmatrix} \tag{14}$$

From Eq. (14) one can obtain that servers 1 to 3 are assigned to carry out the service mission for GEO targets. Based on the rendezvous trajectory planning method in Sec. 3, the rendezvous process is presented in Fig. 4. The three GEO targets are all served by the offline and online combined decision-making method.

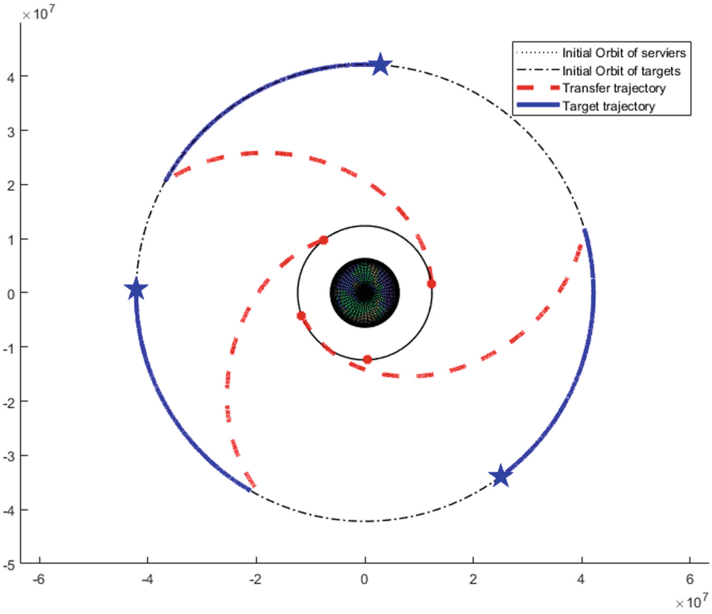

Fig. 4. Rendezvous process for 4 servers serving 3 targets

6 Conclusion

This paper investigates the problem of the on-orbit service with multiple GEO targets by multiple servers. An offline and online combined intelligent decision-making method is proposed based on reinforcement learning. The optimization problem for one server to serve one target is formalized and solved offline and fitted by a RBFNN. And a multi-target decision-making optimization problem is constructed online and solved by the reinforcement learning-based decision-making method. Simulation results verify that the effectiveness of the proposed method.

Acknowledgments. The authors would like to thank the support by the National Natural Science Foundation of China (62103446).

References

1. Li, W.J., et al.: On-orbit service (OOS) of spacecraft: a review of engineering developments. Prog. Aerosp. Sci. **108**, 32–120 (2019)
2. Flores-Abad, A., Ma, O., Pham, K., Ulrich, S.: A review of space robotics technologies for on-orbit servicing. Prog. Aerosp. Sci. **68**, 1–26 (2014)
3. Xu, W., Liang, B., Li, B., Xu, Y.: A universal on-orbit servicing system used in the geostationary orbit. Adv. Space Res. **48**(1), 95–119 (2011)
4. Zhu, X., Chen, J., Zhang, C., Qiao, B.: Optimal fuel station arrangement for multiple GEO spacecraft refueling mission. Adv. Space Res. **66**(8), 1924–1936 (2020)
5. Yu, X., Guo, J., Zheng, H.: Extended-CBBA-based task allocation algorithm for on-orbit assembly spacecraft. In: IEEE International Conference on Unmanned Systems, pp. 883–888. IEEE (2019)
6. Chu, J., Guo, J., Gill, E.: Distributed asynchronous planning and task allocation algorithm for autonomous cluster flight of fractionated spacecraft. Int. J. Space Sci. Eng. **2**(2), 205–223 (2014)
7. Zhang, Y., Zhang, Q.: On-orbit servicing task allocation for multi-spacecrafts using HDPSO. In: Applied Mechanics and Materials, vol. 538, pp. 150–153. Trans Tech Publ. (2014)
8. Bagchi, T.P.: Near optimal ground support in multi-spacecraft missions: a GA model and its results. IEEE Trans. Aerosp. Electron. Syst. **45**(3), 950–964 (2009)
9. Tripp, H., Palmer, P.: Distribution replacement for improved genetic algorithm performance on a dynamic spacecraft autonomy problem. Eng. Optim. **42**(5), 403–430 (2010)
10. Curtis, H.: Orbital Mechanics for Engineering Students. Butterworth-Heinemann, Oxford (2013)
11. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (2018)
12. Zhang, D., Ma, H.: A Q-learning-based decision making scheme for application reconfiguration in sensor networks. In: International Conference on Computer Supported Cooperative Work in Design, pp. 1122–1127. IEEE (2007)
13. Wang, Y.H., Li, T.H.S., Lin, C.J.: Backward Q-learning: the combination of Sarsa algorithm and Q-learning. Eng. Appl. Artif. Intell. **26**(9), 2184–2193 (2013)