Hindawi
International Journal of Aerospace Engineering
Volume 2023, Article ID 8923088, 10 pages
https://doi.org/10.1155/2023/8923088

Research Article

Automatic Task Planning and Its On-Orbit Verification of Agile Remote Sensing Satellite

Zhang Yahang[1,2] **Yang Haiyue**[3] **Yang Mengfei,**[2] **Yang Ruochu**[3] **and Leng Shuhang**[3]

[1]*College of Computer Science and Technology, Xidian University, Xi'an 710071, China*
[2]*Beijing Institute of Spacecraft System Engineering, Beijing 100094, China*
[3]*College of Aerospace and Civil Engineering, Harbin Engineering University, Harbin 150001, China*

Correspondence should be addressed to Zhang Yahang; zhangyahang@163.com

Received 2 December 2022; Revised 17 January 2023; Accepted 30 January 2023; Published 28 February 2023

Academic Editor: Jinchao Chen

An on-board autonomous task planning system is designed and implemented in this article, aiming at the problem that the current remote sensing satellite needs complex instruction support to perform tasks and depends on the ground system too much. The complex earth observation task description and injection decomposition modules are designed in the system. No more than 128 ordered points describe the curve area task or irregular polygon area task, and the complex task is decomposed into several strips according to the satellite imaging width. Then, the task's orbit, attitude, mobility, energy, and time constraints are calculated through the modules of on-board orbit prediction, agile attitude calculation, track planning, energy prediction, and visible arc calculation. Finally, the on-board autonomous task planning and execution are completed through task solution space search and metatask command generation modules. The on-orbit flight verification is carried out on the high-resolution multimode (GFDM) satellite. The results show that this paper's on-board autonomous task planning system can complete complex task injection and autonomous planning and finally execute.

1. Introduction

Space-based remote sensing systems play a critical role in scenarios of flood and forest disaster prevention, land survey, and other emerging fields. Ground operation and management are becoming more complex due to the increasing trends in the complexity of observation requirements and the number of satellites in constellation or swarm [1–4]. Hence, the traditional task planning mode is hard to apply [5, 6]. In addition, current satellite management and control procedure relies on the ground telemetry and telecontrol (TM & TC) station, which leads to a longer time duration from the input of task requirement to remote sensing product generation. In such a centralized mode, the task planning system has flaws in image information timeliness, emergency task response, multisatellite collaborative scheduling, etc. Therefore, it is urgent to develop on-board intelligence and autonomy, including on-board task planning, real-time evaluation, decision-making, multisatellite, multipayload cooperation, and other technical research.

Most current task planning systems are applied in the ground computing environment; they do not consider the computing and storage resources of on-board computers [7–14]. Correspondingly, most currently developed satellites have minimal computing ability and memory space, which is impossible to plan complex tasks on orbit. So, the algorithms have to be redesigned to fit the ability of on-board computers. To overcome shortage of computational resource, Zhang et al. designed a satellite and ground station hybrid method to plan remote sensing tasks [15]. An autonomous task plan framework that could run on-board was proposed to solve agile satellite imaging task planning problems [16, 17]. On JWST satellite, the event-driven system was established to deal on-board tasks [18]. Here, high-resolution

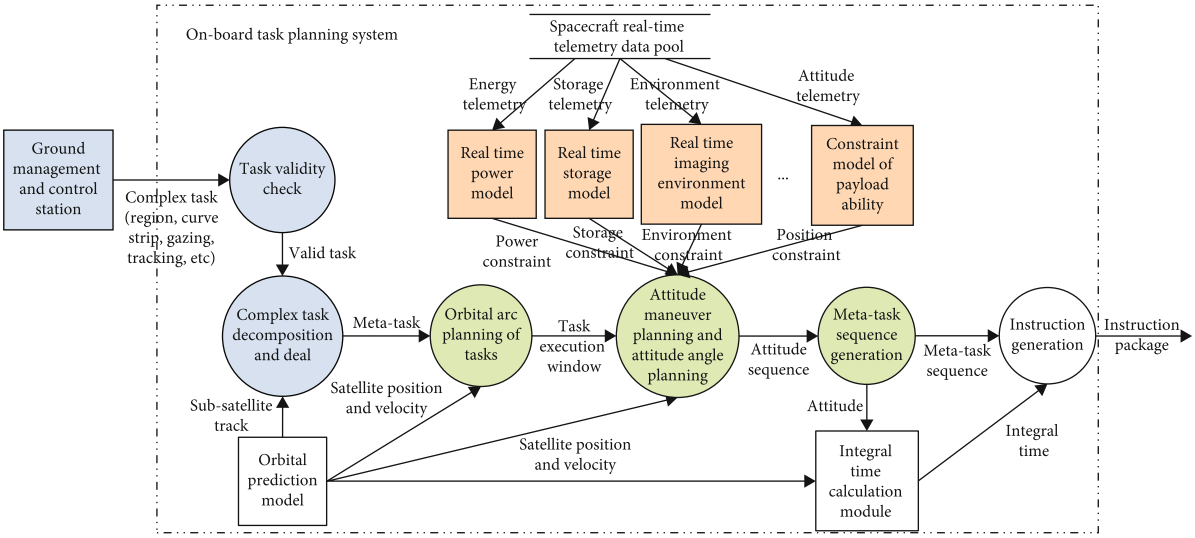

On-board task planning system
Spacecraft real-time telemetry data pool
Energy telemetry
Storage telemetry
Environment telemetry
Attitude telemetry
Ground management and control station
Complex task (region, curve strip, gazing, tracking, etc)
Task validity check
Valid task
Real time power model
Real time storage model
Real time imaging environment model
...
Constraint model of payload ability
Power constraint
Storage constraint
Environment constraint
Position constraint
Complex task decomposition and deal
Meta-task
Orbital arc planning of tasks
Task execution window
Attitude maneuver planning and attitude angle planning
Attitude sequence
Meta-task sequence generation
Meta-task sequence
Instruction generation
Instruction package
Sub-satellite track
Satellite position and velocity
Attitude
Orbital prediction model
Satellite position and velocity
Integral time calculation module
Integral time

Figure 1: On-board task planning architecture.

multimode satellite (GFDM) is taken as an attempt to experiment with on-orbit task and attitude planning, its processing capacity is no more than 50 MHz, and SRAM is no more than 6 MByte. To suit the GFDM satellite, the on-board task planning system was modified to ensure computational efficiency in limited on-board resources and complete planning in small storage space.

This paper designed an on-board automatic task planning system and conducted a set of on-orbit flight verification, aiming at the new requirements of task planning for agile remote sensing satellites. The second section will give the overall design, planning algorithm design, and processing flow design of the on-board task planning system. The third section will detail the on-orbit flight verification based on the GFDM satellite. Finally, the conclusion and prospect of this paper are given in the Section 4.

2. Design of On-Board Task Planning System

The existing agile satellite takes a single strip or a spot area as a *metatask* to complete remote sensing [19]; here, the metatask refers to the task that can be imaged or executed by switching on and off once by the payload [20]. The on-board task planning system designed in this paper is compatible with these metatasks; the architecture is shown in Figure 1. Except for metatasks, for a better description, the complex task is defined in this paper as the task that can only be imaged entirely requiring several contiguous strips, such as observation tasks of curve strip or polygonal area. The on-board task planning system of agile remote sensing satellites considers the satellite orbit, imaging width, and agile ability. The metatask is the basic observation task executed by the on-board task planning system, and the complex task needs to be decomposed into multiple metatasks. The task planning system will decompose the curve strips or irregular polygons into a series of strips or spots according to the comprehensive constraints of satellite imaging width, subsatellite tack, memory size, power, and acceleration of angle velocity.

2.1. Overall Design. The ground operation and control center receives users' observation requirements and injects multiple complex tasks or metatasks into satellite. The on-board task planning system autonomously completes task planning, instruction generation, and other operations. The system architecture is shown in Figure 1. After receiving and decomposing the complex tasks, the satellite implements task planning according to the real-time on-orbit state telemetry. A task planning process is divided into 8 subprocedures, that is, from requirement gathering to instruction generating, including the following:

Step 1. Gather requirements. The users send tasks to the ground station by means of polygons, curve strips, gaze, and tracking tasks and then forward to satellites through the satellite-ground link.

Step 2. Examine the validity of the task. Check task time, attributes, and types to determine whether the task is valid or not.

Step 3. Decompose complex tasks. The on-board task planning system decomposes and processes complex tasks according to the subsatellite track vector, generating the metatask.

Step 4. Planning execution window. Select imaging orbital arc.

Step 5. Calculate external constraints. Calculate the constraints of power [21], storage, attitude, and so on.

Step 6. Plan the attitude and attitude maneuver trajectory.

Step 7. Generate metatasks.

Step 8. Generate instructions.

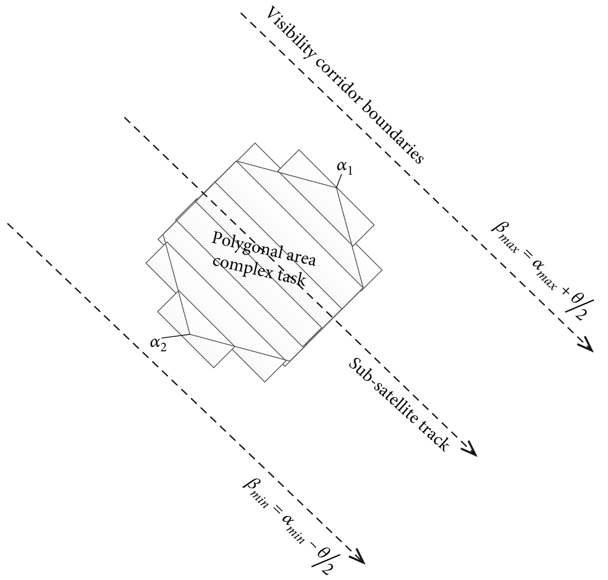

Visibility corridor boundaries
α_1
$\beta_{max} = \alpha_{max} + \theta/2$
Polygonal area
complex task
α_2
Sub-satellite track
$\beta_{min} = \alpha_{min} - \theta/2$

Figure 2: A polygonal area complex task decomposition.

2.2. *Complex Task Description and Decomposition*

2.2.1. Task Description. Complex tasks include irregular curve strips, irregular areas, point group target, etc. Tasks can be described by the following essential attributes:

(1) Task ID: the unique ID of the task, which is used to retrieve, delete, or modify the task

(2) Task type: it includes point target task, nontrack strip task, curve strip task, and region task

(3) Task priority: it includes high, medium, and low priorities. Task priority is mainly used to ensure the execution of tasks with high priority after multiple tasks are mutually exclusive

(4) Payload type: it can be divided into visible light, multispectral, infrared, and other types according to the type of remote sensor

(5) Time range of observation task: the observation and data return of the task must be completed within the time range of the observation task. Some emergency observation tasks, such as flood and other disaster monitoring, will lead to the loss of due value of observation data beyond a specific time range

(6) Solar elevation angle: solar elevation angles for visible light imaging tasks

(7) Cloud cover: the prediction of cloudiness over imaging target and the probability of covering to target

2.2.2. Decompose Tasks. Wu et al. presented a novel splitting algorithm to region target for satellite swarm [8], which is the first comprehensive solution for satellite task planning. Taking the method in the paper as a reference, a region target is handled by strips paralleled to subsatellite track.

(1) Decomposition of Polygonal Area Task. It is necessary to decompose the polygonal areas according to the trend of the subsatellite track since a single strip cannot cover a complex task area, as shown in Figure 2.

The first thing is to examine whether all area points are within the visibility corridor boundaries related to the subsatellite track and satellite maneuver before decomposing. The area would be decomposed according to the boundaries' width if it passed the examination. Otherwise, this area becomes unobservable this time, so there is no need to decompose.

The visibility corridor range can be calculated by

$$\begin{aligned}\beta_{\min} &= \alpha_{\min} - \frac{\theta}{2},\\ \beta_{\max} &= \alpha_{\max} + \frac{\theta}{2},\end{aligned} \tag{1}$$

where $\beta_{\min}$ and $\beta_{\max}$ are the satellite's minimum and maximum observation angle, $[\alpha_{\min}, \alpha_{\max}]$ is the satellite maneuver range, and θ is the FOV (field of view).

The polygon area is decomposed based on the width k after obtaining the maneuver range, and the dynamic decomposition process is as follows:

Step 1. The irregular polygon is initialized into a convex polygon P by convex hull algorithm, and the subtask (metatask) set is initialized, $M = \Theta$.

Step 2. Find the left most point position x_1 and the rightmost point position x_2 in the irregular area, set the left direction to negative and the right direction as positive, and let $x = x_1$.

Step 3. Obtain the strip l, and let $M = M \cup l$. Strip l is obtained by finding the intersected figure of polygon P and straight lines μ_1 and μ_2, where μ_1 is obtained by drawing a line through x_1 paralleled to the subsatellite track, and μ_2 is obtained by translating μ_1 by width k.

Step 4. Update the current location, $x = x + k$; the updated result is shown in Figure 3.

Step 5. Check whether x is less than x_2; if true, turn to Step 2.

Step 6. Output M, end decomposition.

(2) Decomposition of Curve Strip Task. The curve strip task is described by point set $\mathbf{P}$ in which P_i $(1 < i \leq n)$ is the characteristic point of a curve and is checked according to agile ability of satellite and imaging width k. Only the curve whose characteristic points set $\mathbf{P}$ are all in observable corridor can be decomposed. The procedure is as follows:

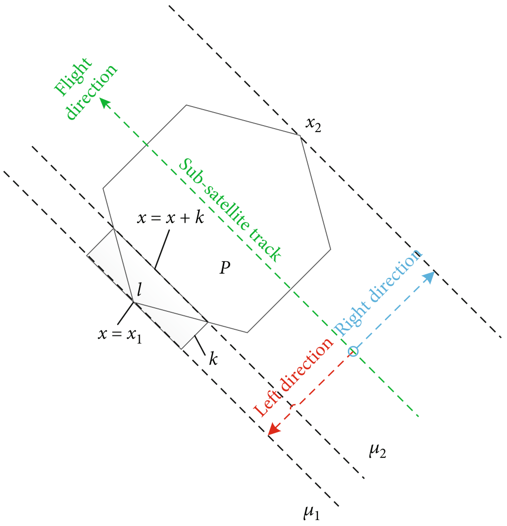


Figure 3: Obtaining the strip.

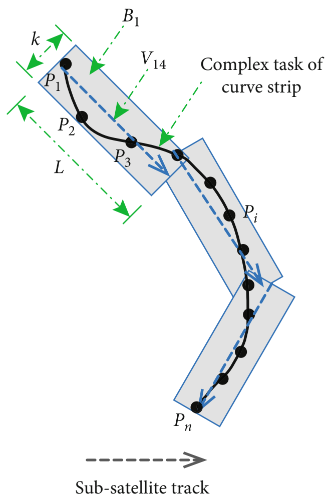


Figure 4: Decomposition of the curve strip task.

Step 1. Selecting P_i as a start point of vector V_{ij} and selecting point P_j $(i < j \leq n)$, one by one, as end point of vector V_{ij}, then, the vector $V_{ij} = <P_i, P_j>$ is obtained.

Step 2. According to imaging width k and vector V_{ij}, the rectangle **B** whose length L is variable can be obtained.

Step 3. Selecting P_j in $i < j \leq n$, to get m_max end point, then to form vector $V_{ij}^{\mathrm{m\_max}} = <P_i, P_j^{\mathrm{m\_max}}>$. Further, strip B_i that can cover all characteristic points between P_i and P_j can be obtained.

Step 4. Using $i = i + \mathrm{m\_max}$ to update i, then repeating Step 1 to Step 3, until $i + \mathrm{m\_max} > n$.

The decomposition procedure of the curve strip task is shown in Figure 4.

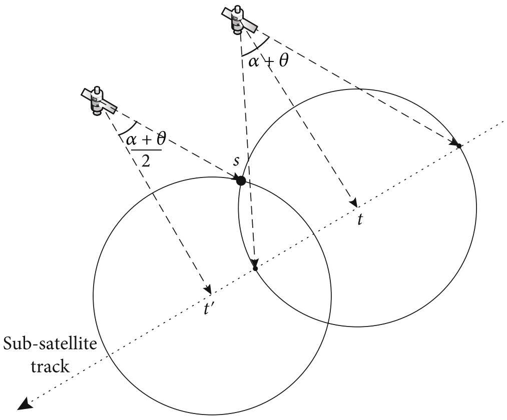


Figure 5: Scheme of visible window calculation.

2.3. Task Execution Time Window Calculation Algorithm. Visible window can be calculated based on the satellite's maximum maneuver angle $\alpha_{\max}$ and FOV θ, as is shown in Figure 5.

A brief process of the visible window algorithm is as follows:

Step 1. Initialize the satellite observation range $\delta = \alpha + \theta$ and the execution window set T.

Step 2. The starting execution window $T_1^s = (t_1^s, t'^s_1)$ and the ending execution window $T_1^e = (t_1^e, t'^e_1)$ for metatask $m_1\{s_1, e_1\} \in M$ are calculated according to the satellite orbit prediction and δ, where s_1 is the starting point and e_1 is the ending point.

Step 3. Let $T = T \cup T_1$.

Step 4. For $\forall m_i \in M$, repeat Step 2 and Step 3.

Step 5. Final set T will be the prediction of execution time window for every metatask decomposed from this complex task.

2.4. Task Planning. $M\langle m_1, m_2, \cdots, m_n\rangle$ is recorded as the complex task M after decomposing (briefly introduced in Section 2.2.2), where n is the number of the meta tasks.

For meta-task $m_i \in M$,

(i) $\langle \mathrm{slon}_i, \mathrm{slat}_i\rangle$ is longitude and altitude at starting point s_i

(ii) $\langle \mathrm{elon}_i, \mathrm{elat}_i\rangle$ is longitude and altitude at ending point e_i

(iii) l_i is strip length

The execution time is finally determined by the attitude and track planning, satisfying multiple constraints including attitude angle, angle velocity, and execution time window.

2.4.1. Attitude Planning. The attitude representation method is based on 1-2-3 Euler angular rotation sequence. The roll angle is φ, pitch angle is θ, and yaw angle is ψ. The attitude planning algorithm is as follows:

Step 1. Calculate the observation point position at the execution time in J2000 coordinate system.

$\vec{r}_{p,\mathrm{ECI}} = [x_1^p, y_1^p, z_1^p, vx_1^p, vy_1^p, vz_1^p]$; the starting point s_1 position at t_1^s in the J2000 coordinate system can be calculated by its longitude and latitude, where the execution time window $T_1 = (t_1^s, t'^e_1)$ for metatask $m_1\langle s_1, e_1\rangle$ is obtained in Section 2.3.

Step 2. Calculate the position and velocity of the satellite at the execution time in the J2000 coordinate system.

Orbital data at t_1^s can be obtained from the satellite GPS receiver, including a: semimajor axis; e: eccentricity; i: orbit inclination angle; Ω: right ascension of ascending node; w: argument of perigee; and M: mean anomaly.

According to spacecraft dynamics equation [22, 23], $\vec{r}_{\mathrm{SAT,ECI}} = [x_1^s, y_1^s, z_1^s, vx_1^s, vy_1^s, vz_1^s]$, the position and velocity in the J2000 coordinate system can be obtained by the following formulas, after calculating the true anomaly f:

$$\begin{gathered}
\begin{cases}
x = r(\cos u \cos \Omega - \sin u \cos i \sin \Omega), \\
y = r(\cos u \sin \Omega + \sin u \cos i \cos \Omega), \\
z = r \sin u \sin i, \\
vx = v[\sin \gamma(\cos u \cos \Omega - \sin u \cos i \sin \Omega) + \cos \gamma(-\sin u \cos \Omega - \cos u \cos i \sin \Omega)], \\
vy = v[\sin \gamma(\cos u \sin \Omega + \sin u \cos i \cos \Omega) + \cos \gamma(-\sin u \sin \Omega + \cos u \cos i \cos \Omega)], \\
vz = v(\sin \gamma \sin u \sin i + \cos \gamma \cos u \sin i),
\end{cases} \\
r = \frac{a(1 - e^2)}{1 + e \cos f}, \\
v = \sqrt{\mu\left(\frac{2}{r} - \frac{1}{a}\right)}, \\
u = \omega + f, \\
\sin \gamma = \frac{e \sin f}{\sqrt{1 + e^2 + 2e \cos f}},
\end{gathered} \tag{2}$$

where r is the geocentric distance of the satellite; v is the satellite velocity; u is the argument of latitude of the satellite; γ is the track angle; μ is the geocentric gravitational constant, and $\mu = 398600.44\,\mathrm{km}^3/\mathrm{s}^2$.

Step 3. Calculate the satellite attitude parameters.

Roll angle φ and pitch angle θ are obtained by the following formulas:

$$\begin{gathered}
\theta = \begin{cases}
\arcsin\left(\dfrac{r_{x,\mathrm{orbit}}}{r_{\mathrm{orbit}}}\right), & r_{x,\mathrm{orbit}} > 0, r_{y,\mathrm{orbit}} > 0, \quad \theta \in \left[0, \dfrac{\pi}{2}\right], \\
-\arcsin\left(\dfrac{r_{x,\mathrm{orbit}}}{r_{\mathrm{orbit}}}\right), & r_{x,\mathrm{orbit}} < 0, r_{y,\mathrm{orbit}} < 0, \quad \theta \in \left[-\dfrac{\pi}{2}, 0\right], \\
\dfrac{\pi}{2} - \arcsin\left(\dfrac{r_{x,\mathrm{orbit}}}{r_{\mathrm{orbit}}}\right), & r_{x,\mathrm{orbit}} < 0, r_{y,\mathrm{orbit}} > 0, \quad \theta \in \left[\dfrac{\pi}{2}, \pi\right], \\
-\dfrac{\pi}{2} + \arcsin\left(\dfrac{r_{x,\mathrm{orbit}}}{r_{\mathrm{orbit}}}\right), & r_{x,\mathrm{orbit}} < 0, r_{y,\mathrm{orbit}} > 0, \quad \theta \in \left[-\pi, -\dfrac{\pi}{2}\right],
\end{cases} \\
\theta = \arcsin \frac{r_{x,\mathrm{orbit}}}{r_{\mathrm{orbit}}},
\end{gathered} \tag{3}$$

where $\Delta\vec{r}_{\mathrm{ECI}} = \vec{r}_{\mathrm{SAT,ECI}} - \vec{r}_{p,\mathrm{ECI}}$ is the vector difference between the imaging point and the satellite in the J2000 coordinate system and $\Delta\vec{r}_{\mathrm{orbit}} = [r_{x,\mathrm{orbit}}, r_{y,\mathrm{orbit}}, r_{z,\mathrm{orbit}}]$ is the orbit coordinate system converted from $\Delta\vec{r}_{\mathrm{ECI}}$.

And roll angular velocity $\dot{\varphi}$ and the pitch angular velocity $\dot{\theta}$ can be calculated using quadratic interpolation method, to improve the calculating speed and suit the limited onboard computational resources.

Then, obtain the yaw angle ψ according to φ, θ, $\dot{\varphi}$, and $\dot{\theta}$. The yaw angular velocity $\dot{\psi}$ is calculated in the same way as obtaining $\dot{\varphi}$ and $\dot{\theta}$.

Step 4. Calculate the attitude parameters at the ending point.

t_1^e is obtained by the formula

$$t_1^e = t_1^s + \frac{l_1}{v_{\mathrm{obs}}}, \tag{4}$$

where v_{obs} is the push-broom speed of the satellite and l_i is the strip length of a metatask.

If $t_1^e \notin (t_1^s, t'^{e}_{1})$, which means the execution window T_1 cannot cover t_1^e; this metatask has no solution. Otherwise, [$\varphi_1^e, \theta_1^e, \psi_1^e, \dot{\varphi}_1^e, \dot{\theta}_1^e, \dot{\psi}_1^e$] can be obtained by repeating Step 1–Step 3 according to the orbit data at t_1^e.

2.4.2. Track Planning. The satellite needs to switch attitude through track planning, from $[\varphi_{i-1}^e, \theta_{i-1}^e, \psi_{i-1}^e, \dot{\varphi}_{i-1}^e, \dot{\theta}_{i-1}^e, \dot{\psi}_{i-1}^e]$ at t_{i-1}^e to $[\varphi_i^s, \theta_i^s, \psi_i^s, \dot{\varphi}_i^s, \dot{\theta}_i^s, \dot{\psi}_i^s]$ at t_i^s between executing metatasks m_{i-1} and m_i, with constraints of

(i) maximum switching time $\Delta t_{i-1,i} = t_i^s - t_{i-1}^e$

(ii) maximum angular velocity v_{lim}

(iii) maximum angular acceleration a_{lim}

Other than that, the variation curves of angle $\theta(t)$, angular velocity $\dot{\theta}(t)$, and angular acceleration $\ddot{\theta}(t)$ in every axis (yaw, pitch, and roll) of the satellite need to be smooth. This is to ensure that the remote sensing satellite can perform tasks without jitter.

The specific algorithm is as follows:

(i) $\Delta t_{i-1,i} = \tau$, where τ is the minimum track planning time (we set $\tau = 5$ in this paper)

(ii) Calculate the satellite attitude parameters $[\varphi_{i,\mathrm{tmp}}^s, \theta_{i,\mathrm{tmp}}^s, \psi_{i,\mathrm{tmp}}^s, \dot{\varphi}_{i,\mathrm{tmp}}^s, \dot{\theta}_{i,\mathrm{tmp}}^s, \dot{\psi}_{i,\mathrm{tmp}}^s]$ at $t_{i,\mathrm{tmp}}^s = t_{i-1}^e + \Delta t_{i-1,i}$

(iii) Obtain the acceleration $[\ddot{\varphi}_{i,\mathrm{tmp}}^s, \ddot{\theta}_{i,\mathrm{tmp}}^s, \ddot{\psi}_{i,\mathrm{tmp}}^s]$ using the quadratic interpolation method

Step 1. Initialize $\Delta t_{i-1,i}$, and calculate the attitude parameters.

Step 2. Obtain the attitude parameters at every control period time (0.1 second in this paper) in $\Delta t_{i-1,i}$, based on the 5th-order track planning method.

The formulas of this method are shown below:

$$\begin{cases} \theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5, \\ \dot{\theta}(t) = a_1 + 2a_2 t + 3a_3 t^2 + 4a_4 t^3 + 5a_5 t^4, \\ \ddot{\theta}(t) = 2a_2 + 6a_3 t + 12a_4 t^2 + 20a_5 t^3, \end{cases}$$

$$\begin{cases} a_0 = \theta_0, \\ a_1 = \dot{\theta}_0, \\ a_2 = \dfrac{\ddot{\theta}_0}{2}, \\ a_3 = \dfrac{20\theta_f - 20\theta_0 - \left(8\dot{\theta}_f + 12\dot{\theta}_0\right)t_f - \left(3\ddot{\theta}_0 \text{-} \ddot{\theta}_f\right)t_f^2}{2t_f^2}, \\ a_4 = \dfrac{30\theta_0 - 30\theta_f - \left(14\dot{\theta}_f + 16\dot{\theta}_0\right)t_f - \left(3\ddot{\theta}_0 \text{-} 2\ddot{\theta}_f\right)t_f^2}{2t_f^2}, \\ a_5 = \dfrac{12\theta_f - 12\theta_0 - \left(6\dot{\theta}_f + 6\dot{\theta}_0\right)t_f - \left(\ddot{\theta}_0 \text{-} \ddot{\theta}_f\right)t_f^2}{2t_f^2}, \end{cases} \tag{5}$$

where $\theta_0, \dot{\theta}_0, \ddot{\theta}_0$ is the track planning starting angle, angular velocity, and angular acceleration at t_{i-1}^e and $\theta_f, \dot{\theta}_f, \ddot{\theta}_f$ is the track planning ending angle, angular velocity, and angular acceleration at $t_{i,\mathrm{tmp}}^e$.

Step 3. Calculate the compound angular velocity v and compound angular acceleration a using 1-2-3 Euler angle rotation kinematical equation, which can obtain maximum compound angular velocity $v_{\max}$ and maximum compound angular acceleration.

Step 4. Let $\ell_v = |v_{\max}/v_{\mathrm{lim}}|$ and $\ell_a = |a_{\max}/a_{\mathrm{lim}}|$. If $\ell_v \le 1$ and $\ell_a \le 1$, let $t_i^s = t_{i,\mathrm{tmp}}^s$, track planning end. Otherwise, let $\ell_{\max} = \max(\ell_v, \ell_a)$, update $\Delta t_{i-1,i} = \Delta t_{i-1,i} \times \ell_{\max}$, and repeat Step 1–Step 3.

2.4.3. Analyzing Time Complexity. The proposed method calculates angle relations between each orbital points and target vertexes that compose strip, region, and curve. Orbital points are orbital position in ECI (Earth Central of Inertial) coordinate system, and the points are sorted by time. The count of orbital points is n_t second; for example, there are $n_t = 24\,\mathrm{h} \times 60\ \ \mathrm{min} \times 60\ \ \mathrm{sec} = 86400$ seconds in a day. If the count of vertexes is m_v, the time complexity of the proposed algorithm is $O(m_v n_t)$.

3. On-Orbit Verification Results and Analysis

An on-orbit flight test on satellite (GFDM) with a single-core BM3803 processor (main frequency: 50 MHz) has been taken to verify the system and method proposed in this paper. Each satellite subsystem executes the generated instruction by receiving it from the on-board routing system using the 1553B avionics bus.

This verification was carried out on July 5th, 2020. This verification was carried out using 1/6 of the computing resources for automatic task planning. The result turns out that the proposed method can accomplish complex task, including decomposition and automatic planning.

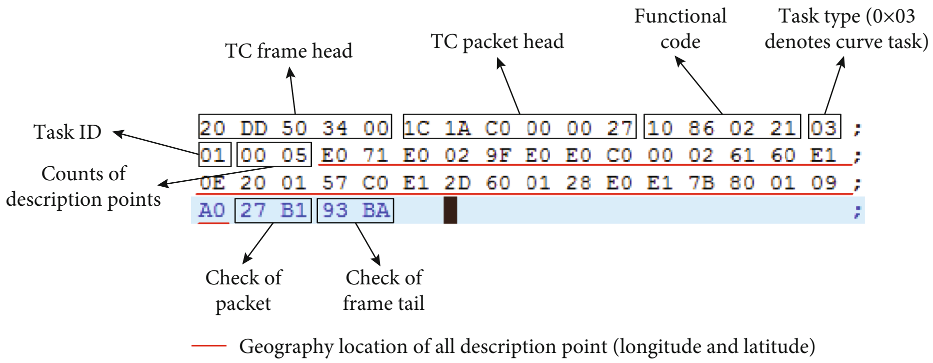
A complex task consisting of a non-along-track strip task and a curve area task is taken as an example in this paper to introduce the on-orbit verification briefly. The specific process and results of this verification are as follows:

(1) Orbit data in verification

GFDM satellite takes the trajectory between two descending nodes as one orbit. Data output interval is set as 1 second. In on-orbit verification, the orbit data is between 2020-7-5 11:40:00 (UTCG) and 2020-7-5 19:40:00 (BJT), semimajor axis 7020.634788 km, eccentricity 0.000381, inclination angle 97.905112°, right ascension of ascending node 260.231848°, argument of perigee 155.560876°, and mean anomaly 229.827296°.

(2) Task injection in verification

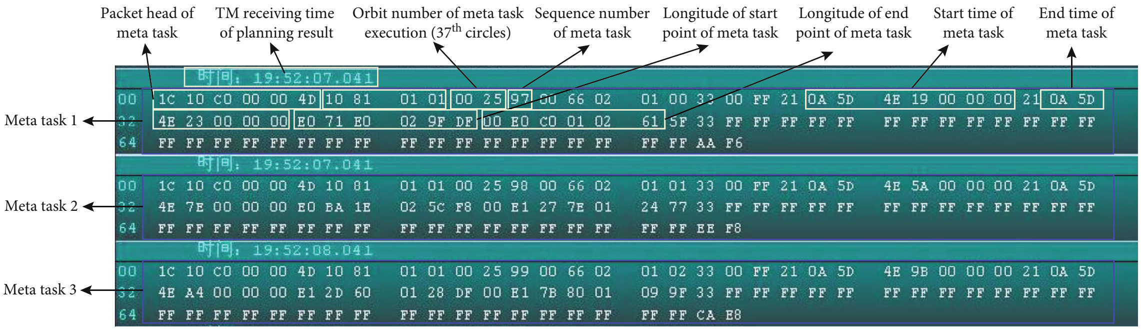
Data of the two subtasks (a curve area task and a non-along-track strip task) are as follows:



TC frame head
TC packet head
Functional code
Task type (0×03 denotes curve task)
Task ID
Counts of description points
Check of packet
Check of frame tail
Geography location of all description point (longitude and latitude)

```
20 DD 50 34 00 1C 1A C0 00 00 27 10 86 02 21 03 ;
01 00 05 E0 71 E0 02 9F E0 E0 C0 00 02 61 60 E1 ;
0E 20 01 57 C0 E1 2D 60 01 28 E0 E1 7B 80 01 09 ;
A0 27 B1 93 BA                                  ;
```

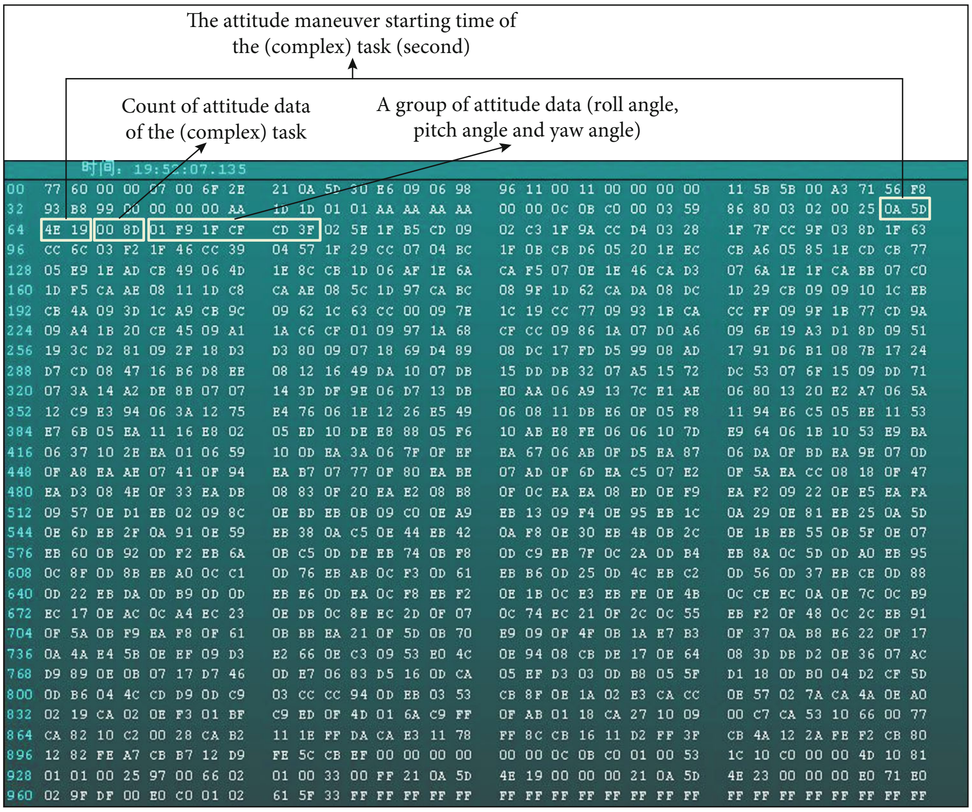
Figure 6: Instruction package on curve complex task.



Packet head of meta task
TM receiving time of planning result
Orbit number of meta task execution (37th circles)
Sequence number of meta task
Longitude of start point of meta task
Longitude of end point of meta task
Start time of meta task
End time of meta task
Meta task 1
Meta task 2
Meta task 3

```
      时间：19:52:07.041
00  1C 10 C0 00 00 4D 10 81   01 01 00 25 97 00 66 02   01 00 33 00 FF 21 0A 5D   4E 19 00 00 00 21 0A 5D
32  4E 23 00 00 00 E0 71 E0   02 9F DF 00 E0 C0 01 02   61 5F 33 FF FF FF FF FF   FF FF FF FF FF FF FF FF
64  FF FF FF FF FF FF FF FF   FF FF FF FF FF FF FF FF   FF FF AA F6
      时间：19:52:07.041
00  1C 10 C0 00 00 4D 10 81   01 01 00 25 98 00 66 02   01 01 33 00 FF 21 0A 5D   4E 5A 00 00 00 21 0A 5D
32  4E 7E 00 00 00 E0 BA 1E   02 5C F8 00 E1 27 7E 01   24 77 33 FF FF FF FF FF   FF FF FF FF FF FF FF FF
64  FF FF FF FF FF FF FF FF   FF FF FF FF FF FF FF FF   FF FF EE F8
      时间：19:52:08.041
00  1C 10 C0 00 00 4D 10 81   01 01 00 25 99 00 66 02   01 02 33 00 FF 21 0A 5D   4E 9B 00 00 00 21 0A 5D
32  4E A4 00 00 00 E1 2D 60   01 28 DF 00 E1 7B 80 01   09 9F 33 FF FF FF FF FF   FF FF FF FF FF FF FF FF
64  FF FF FF FF FF FF FF FF   FF FF FF FF FF FF FF FF   FF FF CA E8
```

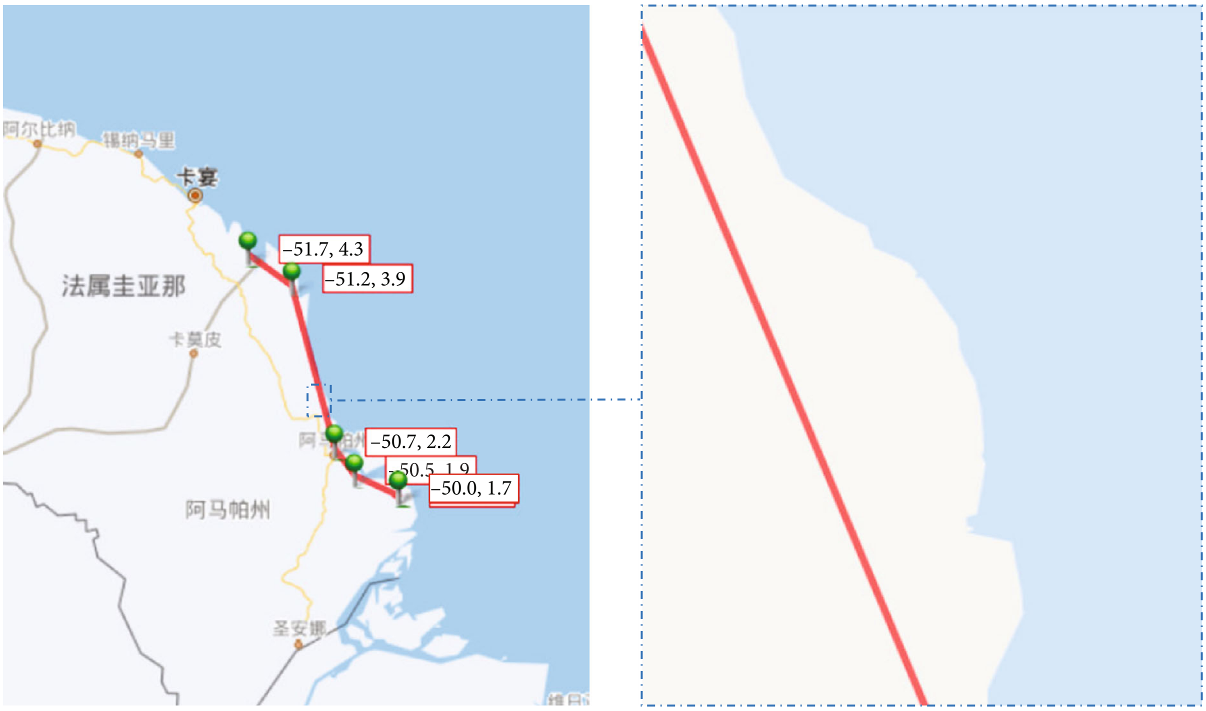
Figure 7: Results of metatasks planned on board for the curve area task.



The attitude maneuver starting time of the (complex) task (second)
Count of attitude data of the (complex) task
A group of attitude data (roll angle, pitch angle and yaw angle)

```
      时间：19:52:07.135
00   77 60 00 00 07 00 6F 2E   21 0A 5D 30 E6 09 06 98   96 11 00 11 00 00 00 00   11 5B 5B 00 A3 71 56 F8
32   93 B8 99 00 00 00 00 AA   1D 1D 01 01 AA AA AA AA   00 00 0C 0B C0 00 03 59   86 80 03 02 00 25 0A 5D
64   4E 19 00 8D 01 F9 1F CF   CD 3F 02 5E 1F B5 CD 09   02 C3 1F 9A CC D4 03 28   1F 7F CC 9F 03 8D 1F 63
96   CC 6C 03 F2 1F 46 CC 39   04 57 1F 29 CC 07 04 BC   1F 0B CB D6 05 20 1E EC   CB A6 05 85 1E CD CB 77
128  05 E9 1E AD CB 49 06 4D   1E 8C CB 1D 06 AF 1E 6A   CA F5 07 0E 1E 46 CA D3   07 6A 1E 1F CA BB 07 C0
160  1D F5 CA AE 08 11 1D C8   CA AE 08 5C 1D 97 CA BC   08 9F 1D 62 CA DA 08 DC   1D 29 CB 09 09 10 1C EB
192  CB 4A 09 3D 1C A9 CB 9C   09 62 1C 63 CC 00 09 7E   1C 19 CC 77 09 93 1B CA   CC FF 09 9F 1B 77 CD 9A
224  09 A4 1B 20 CE 45 09 A1   1A C6 CF 01 09 97 1A 68   CF CC 09 86 1A 07 D0 A6   09 6E 19 A3 D1 8D 09 51
256  19 3C D2 81 09 2F 18 D3   D3 80 09 07 18 69 D4 89   08 DC 17 FD D5 99 08 AD   17 91 D6 B1 08 7B 17 24
288  D7 CD 08 47 16 B6 D8 EE   08 12 16 49 DA 10 07 DB   15 DD DB 32 07 A5 15 72   DC 53 07 6F 15 09 DD 71
320  07 3A 14 A2 DE 8B 07 07   14 3D DF 9E 06 D7 13 DB   E0 AA 06 A9 13 7C E1 AE   06 80 13 20 E2 A7 06 5A
352  12 C9 E3 94 06 3A 12 75   E4 76 06 1E 12 26 E5 49   06 08 11 DB E6 0F 05 F8   11 94 E6 C5 05 EE 11 53
384  E7 6B 05 EA 11 16 E8 02   05 ED 10 DE E8 88 05 F6   10 AB E8 FE 06 06 10 7D   E9 64 06 1B 10 53 E9 BA
416  06 37 10 2E EA 01 06 59   10 0D EA 3A 06 7F 0F EF   EA 67 06 AB 0F D5 EA 87   06 DA 0F BD EA 9E 07 0D
448  0F A8 EA AE 07 41 0F 94   EA B7 07 77 0F 80 EA BE   07 AD 0F 6D EA C5 07 E2   0F 5A EA CC 08 18 0F 47
480  EA D3 08 4E 0F 33 EA DB   08 83 0F 20 EA E2 08 B8   0F 0C EA EA 08 ED 0E F9   EA F2 09 22 0E E5 EA FA
512  09 57 0E D1 EB 02 09 8C   0E BD EB 0B 09 C0 0E A9   EB 13 09 F4 0E 95 EB 1C   0A 29 0E 81 EB 25 0A 5D
544  0E 6D EB 2F 0A 91 0E 59   EB 38 0A C5 0E 44 EB 42   0A F8 0E 30 EB 4B 0B 2C   0E 1B EB 55 0B 5F 0E 07
576  EB 60 0B 92 0D F2 EB 6A   0B C5 0D DE EB 74 0B F8   0D C9 EB 7F 0C 2A 0D B4   EB 8A 0C 5D 0D A0 EB 95
608  0C 8F 0D 8B EB A0 0C C1   0D 76 EB AB 0C F3 0D 61   EB B6 0D 25 0D 4C EB C2   0D 56 0D 37 EB CE 0D 88
640  0D 22 EB DA 0D B9 0D 0D   EB E6 0D EA 0C F8 EB F2   0E 1B 0C E3 EB FE 0E 4B   0C CE EC 0A 0E 7C 0C B9
672  EC 17 0E AC 0C A4 EC 23   0E DB 0C 8E EC 2D 0F 07   0C 74 EC 21 0F 2C 0C 55   EB F2 0F 48 0C 2C EB 91
704  0F 5A 0B F9 EA F8 0F 61   0B BB EA 21 0F 5D 0B 70   E9 09 0F 4F 0B 1A E7 B3   0F 37 0A B8 E6 22 0F 17
736  0A 4A E4 5B 0E EF 09 D3   E2 66 0E C3 09 53 E0 4C   0E 94 08 CB DE 17 0E 64   08 3D DB D2 0E 36 07 AC
768  D9 89 0E 0B 07 17 D7 46   0D E7 06 83 D5 16 0D CA   05 EF D3 03 0D B8 05 5F   D1 18 0D B0 04 D2 CF 5D
800  0D B6 04 4C CD D9 0D C9   03 CC CC 94 0D EB 03 53   CB 8F 0E 1A 02 E3 CA CC   0E 57 02 7A CA 4A 0E A0
832  02 19 CA 02 0E F3 01 BF   C9 ED 0F 4D 01 6A C9 FF   0F AB 01 18 CA 27 10 09   00 C7 CA 53 10 66 00 77
864  CA 82 10 C2 00 28 CA B2   11 1E FF DA CA E3 11 78   FF 8C CB 16 11 D2 FF 3F   CB 4A 12 2A FE F2 CB 80
896  12 82 FE A7 CB B7 12 D9   FE 5C CB EF 00 00 00 00   00 00 0C 0B C0 01 00 53   1C 10 C0 00 00 4D 10 81
928  01 01 00 25 97 00 66 02   01 00 33 00 FF 21 0A 5D   4E 19 00 00 00 21 0A 5D   4E 23 00 00 00 E0 71 E0
960  02 9F DF 00 E0 C0 01 02   61 5F 33 FF FF FF FF FF   FF FF FF FF FF FF FF FF   FF FF FF FF FF FF FF FF
```

Figure 8: On-board attitude planning and trajectory planning of curve complex tasks.



(a) Global view of curve strip task

(b) Zoomed strip at seashore

Figure 9: The curve strip task shown in map software.

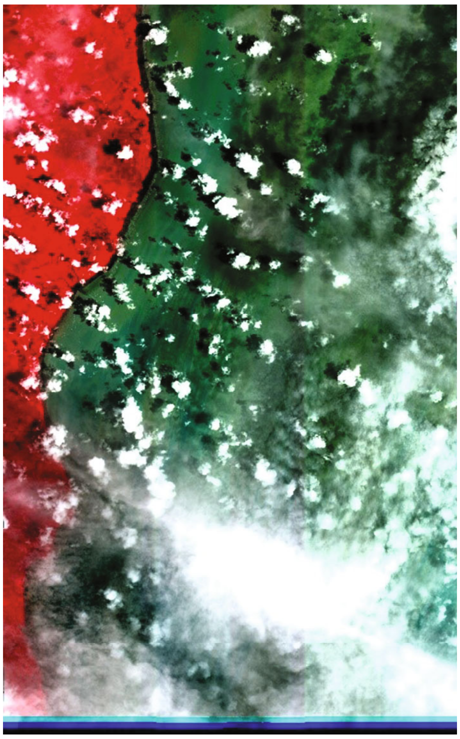


Figure 10: Imaging result of Figure 9(b) by on-board mission planning.

(1) Geographic position of curve area task: <-51.7°, 4.3°>; <-51.2°, 3.9°>; <-50.7°, 2.2°>; <-50.5°, 1.9°>; and <-50.0°, 1.7°>

(2) Geographic position of non-along-track strip task: <-58.4°, -33°>; <-58.4°, -35°>

The generated instruction package taking the curve area task as an example is shown in Figure 6.

(3) On-board task decomposition in verification

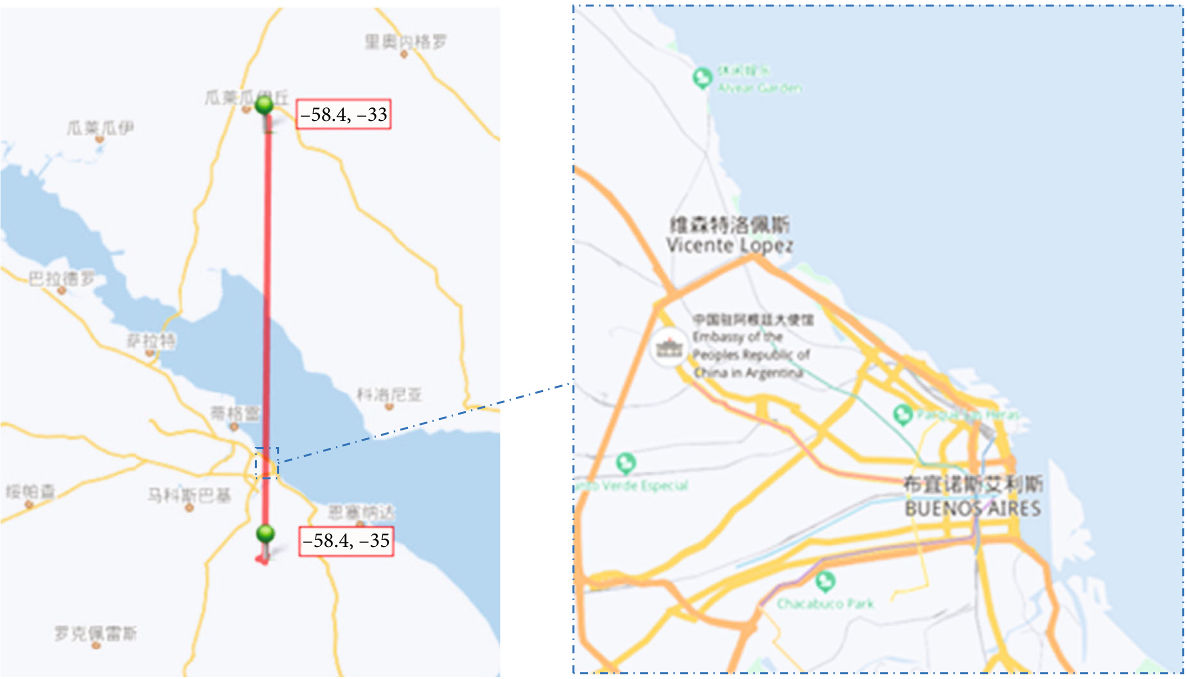
The command to start the on-board autonomous mission planning is sent at 19:50:11, July 5th, 2020, Beijing time. After about 117 seconds, the on-board mission decomposition and planning results are received at 19:52:08, July 5th, 2020, Beijing time, and transmitted to the ground center in the form of telemetry package. As shown in Figure 6, the curve task is decomposed into three imaging metatasks. Taking metatask 1 as an example, its planned start time is 173887001 (0x0a5d4e19) seconds of on-board time. The corresponding Beijing time is 2020-7-5 21:55:48. The time coding method is similar to most typical embedded system, like IoT (Internet of Things) [24, 25].

The command to boot the on-board autonomous task planning is at BJT 2020-7-5 19:50:11. And the results of on-board task decomposition and planning are at BJT 2020-7-5 19:52:08. The instruction package is shown in Figure 7.

(4) Attitude data after planning

The results of attitude planning and track planning are transmitted down to the ground station in the form of instruction package, as shown in Figure 8.

(5) Curve strip task verification



(a) Global view of non-along-track strip task

(b) Zoomed strip at Buenos Aires

Figure 11: Non-along-track strip task shown in map software.

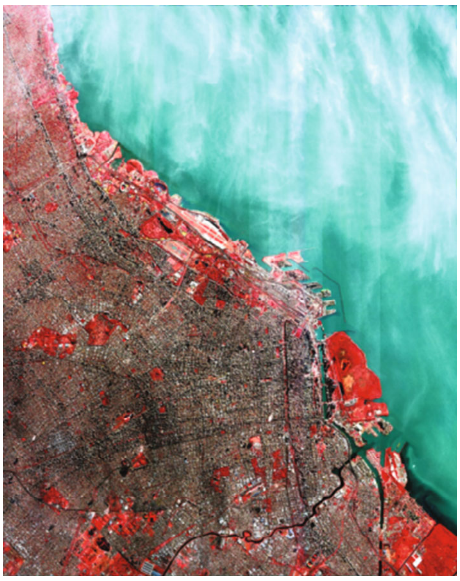


Figure 12: Imaging result of Figure 11(b) by on-board mission planning.

To verify the curve strip task, a curve task with 5 characteristic points is planned on board. Five points by longitude and latitude with degree are <-51.7, 4.3>; <-51.2, 3.9>; <-50.7, 2.2>; <-50.5, 1.9>; and <-50.0, 1.7>. The curve and its partial zoomed position are shown in Figure 9 in which Figure 9(a) is the global view. The curve strip task is planned and executed by GFDM satellite, and its partial image (corresponding to Figure 9(b)) is shown in Figure 10.

(6) Non-along-track strip task verification

A non-along-track strip with start point at <-58.4, -33> and end point at <-58.4, -35> is injected to on-orbit satellite. The non-along-track strip is lined in Figure 11(a), and its partial is zoomed into Figure 11(b).

Non-along-track strip task was successfully planned and executed on-board on July 5th, 2020, Beijing time, and its result of image at Buenos Aires is shown in Figure 12.

4. Conclusion

The results from the experiment of the GFDM show that the automatic on-board task planning system can bring the following capabilities' improvement:

(1) The on-orbit task planning is feasible. The automatic level of satellite is improved; meanwhile, the dependence of satellites on the ground has been decreased

(2) Facilitated customer manipulation. Customers can focus on task-related matters and do not need to consider the satellite's complex parameters and knowledge

(3) Fully utilized satellite capability. The task execution is optimized based on the real-time constraints on the satellite, including energy, storage, and orbit

Data Availability

The data used to support the findings of this study are included within this paper.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the Fundamental Research Funds for the Central Universities of China (No. 3072022JC0202), the project D030307.

References

[1] W. Jiang, H. Hao, and Y. Li, "Review of task scheduling research for the Earth observation satellites," *System Engineering and Electronics*, vol. 35, pp. 1878–1885, 2013.

[2] M. J. Hughes and D. J. Hayes, "Automated detection of cloud and cloud shadow in single-date Landsat imagery using neural networks and spatial post-processing," *Remote Sensing*, vol. 6, no. 6, pp. 4907–4926, 2014.

[3] E. Avgoustoglou and T. Tzeferi, "The effect of a sub-grid statistical cloud-cover scheme applied to the COSMO local numerical weather prediction model over the wider geographical domain of Greece," *Atmospheric Research*, vol. 152, pp. 69–73, 2015.

[4] H. Zhao, X. Wu, Y. Xie, Y. Du, Z. Zhang, and Y. Li, "Rotation matrix-based finite-time attitude synchronization control for flexible spacecrafts with unknown inertial parameters and actuator faults," *ISA Transactions*, vol. 128, Part A, pp. 276–289, 2022.

[5] K. Zhang, Y. Sun, L. Xia, Z. Zhu, and J. Wang, "A method of network satellite on-orbit distributed collaborative mission scheduling," *Journal of Harbin Engineering University*, vol. 40, pp. 393–399, 2019.

[6] L. Yu, X. Wu, Y. Mao, H. Gao, and Y. Hao, "Task allocation for distributed remote sensing satellites based on contract network algorithm," *Journal of Harbin Engineering University*, vol. 41, pp. 1059–1065, 2020.

[7] M. Lemaître, G. Verfaillie, F. Jouhaud, J. M. Lachiver, and N. Bataille, "Selecting and scheduling observations of agile satellites," *Aerospace Science and Technology*, vol. 6, no. 5, pp. 367–381, 2002.

[8] X. Wu, Y. Yang, Y. Sun, Y. Xie, X. Song, and B. Huang, "Dynamic regional splitting planning of remote sensing satellite swarm using parallel genetic PSO algorithm," *Acta Astronautica*In Press.

[9] Z. Yan, Y. Chen, and L. Xing, "Agile satellite scheduling based on improved ant colony algorithm," *System Engineering Theory & Practice*, vol. 34, pp. 793–801, 2014.

[10] P. Tangpattanakul, N. Jozefowiez, and P. Lopez, "A multi-objective local search heuristic for scheduling earth observations taken by an agile satellite," *European Journal of Operational Research*, vol. 245, no. 2, pp. 542–554, 2015.

[11] F. Bunkheila, E. Ortore, and C. Circi, "A new algorithm for agile satellite-based acquisition operations," *Acta Astronautica*, vol. 123, pp. 121–128, 2016.

[12] R. Xu, H. Chen, X. Liang, and H. Wang, "Priority-based constructive algorithms for scheduling agile earth observation satellites with total priority maximization," *Expert Systems with Applications*, vol. 51, pp. 195–206, 2016.

[13] Y. She, S. Li, and Y. Zhao, "Onboard mission planning for agile satellite using modified mixed-integer linear programming," *Aerospace Science and Technology*, vol. 72, pp. 204–216, 2018.

[14] Y. Li, Z. Sun, and D. Ye, "Robust PD+ controller for satellite attitude fast maneuvering," *Journal of Harbin Engineering University*, vol. 40, pp. 703–709, 2019.

[15] Z. Chao, C. Jinyong, L. Yanbin, L. Yuqing, and C. Weijie, "Satellite group autonomous operation mechanism and planning algorithm for marine target surveillance," *Chinese Journal of Aeronautics*, vol. 32, no. 4, pp. 991–998, 2019.

[16] L. Song, C. Yingwu, X. Lining, and G. Xiaojun, "Time-dependent autonomous task planning of agile imaging satellites," *Journal of Intelligent & Fuzzy Systems*, vol. 31, no. 3, pp. 1365–1375, 2016.

[17] F. N. Kucinskis and M. G. V. Ferreira, "Planning on-board satellites for the goal-based operations for space missions," *IEEE Latin America Transactions*, vol. 11, no. 4, pp. 1110–1120, 2013.

[18] B. Vicki, Z. Dean, and W. William, "The care and feeding of the JWST on-board event-driven system," in *Conference on Observatory Operations: Strategies, Processes, and Systems III*, San Diego, CA, 2010.

[19] L. Yu, J. Wang, and S. Zhao, "Design and verification of GFDM satellite ground-to-satellite task management procedure," *Spacecraft Engineering*, vol. 30, pp. 43–51, 2021.

[20] X. Dong, *Research on multi type task combination scheduling method of optical imaging agile satellite*, Harbin Institute of technology, 2017.

[21] G. Sebestyen, S. Fujikawa, N. Galassi, and A. Chuchra, *Low Earth Orbit Satellite Design*, Microcosm Press and Springer, 2017.

[22] B. Wie, *Space Vehicle Dynamics and Control*, American Institute of Aeronautics and Astronautics, Inc., 2nd edition, 2008.

[23] H. Schaub and J. L. Junkins, *Analytical Mechanics of Space Systems*, American Institute of Aeronautics and Astronautics, Inc., 2003.

[24] Z. Fang, J. Wang, Y. Ren, H. Zhu Han, V. Poor, and L. Hanzo, "Age of information in energy harvesting aided massive multiple access networks," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 5, pp. 1441–1456, 2022.

[25] M. A. Abd-Elmagid, P. Nikolaos, and H. S. Dhillon, "On the role of age of information in the Internet of Things," *IEEE Communications Magazine*, vol. 57, no. 12, pp. 72–77, 2019.