LETTER

Agile Earth Observation Satellite Constellation Mission Planning based on Multi-Agent Transformer

Xiaohe HE[†,††,†††], ***Member***, **Junyan XIANG**[†,††,†††], **Mubiao YAN**[†,††*], **Chengxi ZHANG**[††††], **Zhuochen XIE**[†,††*], ***and*** **Xuwen LIANG**[†,††,†††*], ***Nonmembers***

SUMMARY The Agile Earth Observation Satellite Constellation Mission Planning (AEOSCMP) problem focuses on optimizing target selection and scheduling for multiple satellites to maximize global observation rewards while adhering to resource constraints. To tackle this challenging task, this letter employs the Multi-Agent Transformer (MAT) to convert the joint policy search problem into a sequential decision-making process, optimizing observation policies through the attention mechanism. This approach could provide a theoretical guarantee of monotonic improvement during online training, ensuring consistent and reliable performance enhancements. Experimental results demonstrate that MAT achieves superior observation efficiency compared to state-of-the-art Multi-Agent Reinforcement Learning (MARL) methods.

key words: *Earth observation, Agile satellite constellation, Mission planning, MARL, Multi-Agent Transformer*

1. Introduction

Earth Observation Satellites (EOS) serve as critical tools for global monitoring and management, offering indispensable data for a wide range of applications. These include tracking environmental changes, forecasting weather patterns, managing natural disasters, optimizing agricultural practices, overseeing natural resource utilization, supporting urban planning, and advancing scientific research. Agile Earth Observation Satellites (AEOS) are distinguished by their advanced maneuverability, allowing for rapid and precise attitude adjustments along the roll, pitch, and yaw axes. This capability significantly enhances their operational flexibility compared to traditional EOS. As the demand for high-resolution, timely observational data increases alongside the expansion of satellite constellations, the development and implementation of autonomous, efficient mission planning methods for agile satellite constellations have become increasingly critical [1].

Satellite mission planning, since its seminal formulation by Lemaitre [2], has remained an active and challenging research domain for more than two decades, attracting sustained academic attention. To address this complex optimization problem, researchers have developed a diverse spectrum of methodological approaches, ranging from exact algorithms and heuristic methods to sophisticated metaheuristic techniques. However, both exact and heuristic methods exhibit inherent limitations when applied to dynamic environments, primarily due to their open-loop characteristics. Specifically, when faced with execution failures or emerging requirements, these approaches necessitate comprehensive plan recalculation, resulting in substantial computational overhead and reduced operational efficiency. Recent studies [3], [4] have demonstrated the potential of MARL in addressing the AEOSCMP problem. However, conventional MARL approaches, which typically follow the *Centralized Training for Decentralized Execution* (CTDE) paradigm, have limitations in capturing the full complexity of multi-agent interactions [5].

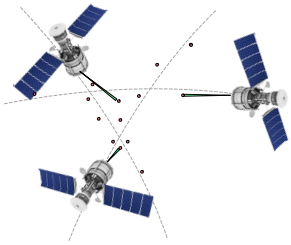


Fig. 1 Description of AEOSCMP problem. Multiple agile satellites cooperatively observe several terrestrial targets.

To address this challenge, we adopt the MAT framework [6], which effectively decomposes the complex joint policy search problem into a more tractable sequential decision-making process. This transformation not only simplifies the problem structure but also enhances the local decision-making capabilities of individual agents.

2. Problem Formulation

The AEOSCMP problem involves N_s AEOS satellites working collaboratively to observe N_t terrestrial targets. The primary objective is to maximize the cumulative value of uniquely imaged targets, as depicted in Fig. 1.

For analytical simplicity, we consider point targets in our formulation. Let $c_j \in \mathbf{C}$ denote an individual target within the complete target set $\mathbf{C}$. Initially, all observation requests are contained in the unfulfilled set $\mathbf{U}$. Upon successful image acquisition of a target $c_j \in \mathbf{U}$, its status transitions to the fulfilled set F and yields a reward equal to its priority

[†]University of Chinese Academy of Sciences, No.1 Yanqihu East Rd, Huairou District, Beijing,101408, China
[††]Innovation Academy for Microsatellites of CAS, No.1 Xueyang Rd, Pudong District, Shanghai,201304, China
[†††]Shanghaitech University,393 Middle Huaxia Rd, Pudong District, Shanghai,201210, China
[††††]Jiangnan University, Lihu Avenue, Wuxi, 214122, China
[*]Corresponding Authors
DOI: 10.1587/transfun.E108.A.1

p_j. The fulfilled set at the subsequent timestep is $\mathbf{F}'$.

Given the distributed nature of satellite operations , we formulate the AEOSCMP problem as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP). Each satellite can only access a limited portion of the global state information and must make decisions based on local observations while coordinating with other satellites to maximize the expected long-term reward. Formally, a Dec-POMDP is defined by a tuple $(\mathbf{S}, \mathbf{A}, \mathbf{\Omega}, \mathbf{T}, \mathbf{R}, \mathbf{O}, \gamma)$, where:

State Space (S): The global environment state, defined as $\mathbf{S} = \{\mathbf{s}^1, \mathbf{s}^2, \cdots, \mathbf{s}^{N_s}, s^{env}\}$, where s^i denotes the state space of satellite i, and s^{env} represents the environmental state. The state space encompasses satellite position and velocity vectors in the Earth-Centered, Earth-Fixed (ECEF) coordinate frame, target-specific information, attitude adjustment rates, battery charge levels, and available storage capacity.

Action Space (A): The action space of all satellites $\mathbf{A} = \mathbf{a}^{1:N_s} = \{\mathbf{a}^1, \mathbf{a}^2, \cdots, \mathbf{a}^{N_s}\}$, where a^i represents the action space of satellite i. The action space encompasses three primary operations: imaging one of the next ten unfulfilled targets, initiating a battery charging sequence, or executing reaction wheel desaturation maneuvers. Each satellite autonomously selects its actions based on its current state and the learned policy parameters.

Observation Space ($\mathbf{\Omega}$) and Observation Function (O): The observation space $\mathbf{\Omega} = \mathbf{\Omega}^{1:N_s} = \{\mathbf{\Omega}^1, \mathbf{\Omega}^2, \cdots, \mathbf{\Omega}^{N_s}\}$ is composed of individual observation spaces Ω^i for each satellite i. The observation function $\mathbf{O}$ specifies the probability of obtaining a combined observation $\mathbf{\Omega}$ given the new state $\mathbf{s}'$ and the combined action $\mathbf{a}$.

Transition Function (T): The transition function defines the probabilistic state transition. In this study, the transitions are deterministic and generated by the Basilisk simulator[7], resulting in $\mathbf{T}(\mathbf{s}'|\mathbf{s}, \mathbf{a}) = 1$.

Reward Function (R): Reward function specifies the immediate reward received after performing action $\mathbf{a}$ in state $\mathbf{s}$. The reward function in a step is:

$$\mathbf{R}(s, s') = \sum_{c_j \in \mathbf{C}} \begin{cases} p_j & c_j \in \mathbf{U} \text{ and } c_j \in \mathbf{F}' \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where p_j is the priority of target c_j.

Discount Factor (γ): $\gamma \in [0, 1]$ models the trade-off between immediate and future rewards.

3. Methodology

To tackle the complex challenges inherent in the AEOSCMP problem, we present a novel solution based on the MAT architecture. This sophisticated framework facilitates end-to-end optimization, enabling the derivation of robust policies through a rigorous and theoretically sound methodology.

Central to our methodology is the Multi-Agent Advantage Decomposition Theorem [8], which establishes a mathematically rigorous framework for decomposing the joint advantage function into constituent individual advantage components. For any given permutation of agents $\{1 : N_s\} = \{1, 2, \cdots, N_s\}$, considering a joint observation Ω and corresponding joint action $\mathbf{a}^{1:N_s}$, the following fundamental decomposition holds universally, independent of auxiliary assumptions:

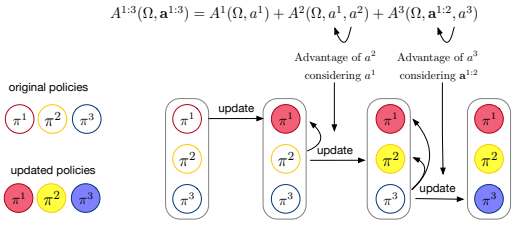


Fig. 2 Illustration of the multi-agent advantage decomposition theorem.

$$A^{1:N_s}(\Omega, \mathbf{a}^{1:N_s}) = \sum_{i=1}^{N_s} A^i(\Omega, \mathbf{a}^{1:i-1}, a^i). \tag{2}$$

Building upon this theoretical foundation, MAT implements a sophisticated mapping mechanism that transforms the agents' observation sequence $(\Omega^1, \ldots, \Omega^{N_s})$ into a corresponding action sequence $(a^1, \ldots, a^{N_s})$. The architecture incorporates sequential dependencies by ensuring that each action a^i is conditioned on all preceding agents' decisions $\mathbf{a}^{1:i-1}$. To achieve this, MAT utilizes an encoder-decoder architecture (illustrated in Fig. 3), where the encoder systematically learns comprehensive representations of joint observations, while the decoder generates actions for individual agents, ensuring coherent and coordinated decision-making across the satellite constellation.

At the heart of the encoder and decoder are attention mechanisms [9], which capture the interrelationship of input sequences. The attention function can be defined as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3}$$

where Q, K, and V are the query, key, and value matrices, respectively, and d_k is the dimension of the key vectors. Self-attention refers to the case where $Q = K = V$.

The encoder processes observations from multiple agents to generate latent representations $\hat{\Omega}^{1:N_s}$. In the training phase, the encoder approximates the value functions to minimize the empirical Bellman error:

$$L_{\text{Encoder}}(\phi) = \frac{1}{TN_s} \sum_{i=1}^{N_s} \sum_{t=0}^{T-1} \left(V_\phi(\hat{\Omega}_t^i) - \hat{R}_t\right)^2 \tag{4}$$

where $V_\phi(\hat{\Omega}_t^i)$ is the estimated value function for agent i at time t, and $\hat{R}_t$ is the estimated reward at time t.

The decoder outputs action representations that generate policy distributions $\pi_\theta^i(a_t^i|\hat{\Omega}_t^{1:N_s}, a_t^{1:i-1})$. To train the decoder, we use a clipped PPO objective:

$$L_{\text{Decoder}}(\theta) = -\frac{1}{TN_s} \sum_{i=1}^{N_s} \sum_{t=0}^{T-1} J(\theta) \tag{5}$$

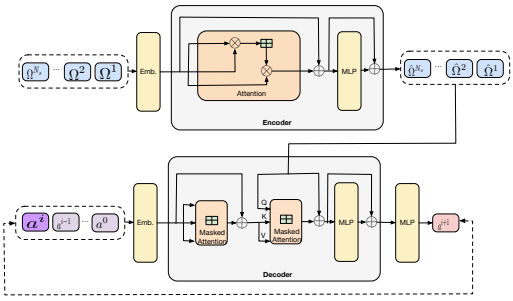


Fig. 3 Multi-Agent Transformer Architecture. MARL problems are transformed into a sequential decision-making process and leverages attention mechanisms to capture dependencies between input sequences.

Algorithm 1 Multi-Agent Transformer

Input: Stepsize α, batch size B, number of agents N_s, episodes K, steps per episode T.
Initialize: Encoder $\{\phi_0\}$, Decoder $\{\theta_0\}$, Replay buffer $\mathcal{B}$.
for $k = 0, 1, \cdots, K-1$ **do**
  for $t = 0, 1, \cdots, T-1$ **do**
    Collect observations $\Omega_t^1, \cdots, \Omega_t^{N_s}$.
    Encode observations to get $\hat{\Omega}_t^1, \cdots, \hat{\Omega}_t^{N_s}$.
    Input $\hat{\Omega}_t^1, \cdots, \hat{\Omega}_t^{N_s}$ to the decoder.
    for $i = 0, 1, \cdots, N_s - 1$ **do**
      Input $\hat{a}_t^0, \cdots, \hat{a}_t^i$ and infer $\hat{a}_t^{i+1}$ with the auto-regressive decoder.
    end for
    Execute joint actions $a_t^1, \cdots, a_t^{N_s}$ in environments and collect the reward $R(\Omega_t, a_t)$.
    Insert $(\Omega_t, a_t, R(\Omega_t, a_t))$ into $\mathcal{B}$.
  end for
  Sample a random minibatch of B steps from $\mathcal{B}$.
  Generate $V_\phi(\hat{\Omega}_t^1), \cdots, V_\phi(\hat{\Omega}_t^{N_s})$ with the output layer of the encoder.
  Calculate $L_{\text{Encoder}}(\phi)$ with Eq. 4.
  Calculate the joint advantage function $\hat{A}$ using Generalized Advantage Estimation (GAE) based on value estimates $V_\phi(\hat{\Omega}_t^1), \cdots, V_\phi(\hat{\Omega}_t^{N_s})$.
  Input $\hat{\Omega}_t^1, \cdots, \hat{\Omega}_t^{N_s}$ and $\hat{a}_t^1, \ldots, \hat{a}_t^{N_s-1}$, generate $\pi_\theta^1, \cdots, \pi_\theta^{N_s}$ at once with the decoder.
  Calculate $L_{\text{Decoder}}(\theta)$ with Eq. 5.
  Update the encoder and decoder by minimizing $L_{\text{Encoder}}(\phi) + L_{\text{Decoder}}(\theta)$ with gradient descent.
end for

$$J(\theta) = \min\left(r_t^i(\theta)\hat{A}_t, \text{clip}(r_t^i(\theta), 1 \pm \epsilon)\hat{A}_t\right) \tag{6}$$

where the probability ratio is:

$$r_t^i(\theta) = \frac{\pi_\theta^i(a_t^i|\hat{\Omega}_t^{1:N_s}, \hat{a}_t^{1:i-1})}{\pi_{\theta_{\text{old}}}^i(a_t^i|\hat{\Omega}_t^{1:N_s}, \hat{a}_t^{1:i-1})}. \tag{7}$$

where $\hat{A}_t$ is the estimated advantage function at time t.

Training details are summarized in Algorithm 1.

4. Experiments

To comprehensively evaluate the effectiveness of MAT, we conduct extensive experiments on a series of benchmark scenarios. The experimental setup consists of N_s satellites operating in a 500km walk delta orbit, accompanied by N_t ground targets distributed along the ground track with stochastic offsets, as illustrated in Fig. 4. The simulation timespan is configured to span one complete orbital period.

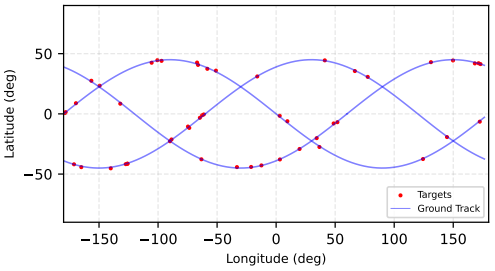


Fig. 4 Simulation setup. N_s satellites operating in a 500km walk-delta orbit, with N_t targets distributed along the ground track with random offsets.

For the algorithm parameter settings, MAT uses a transformer architecture with embedding dimension 512, one attention head. The learning rate is $lr = 0.003$, Adam's $\epsilon = 10^{-5}$, the discount factor $\gamma = 0.99$, and priority weights $p_j = 1$ for simplicity.

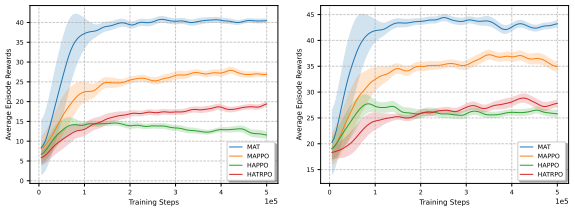
4.1 Analysis of Episode Reward

Episode return is the sum of rewards obtained over an episode. As shown in Fig. 5 and Table 1, MAT demonstrates superior performance across all experimental scenarios compared to state-of-the-art MARL methods, including MAPPO[10], HAPPO[11], and HATRPO[12]. In the scenario with 3 satellites and 50 targets ($N_s = 3, N_t = 50$), MAT demonstrates substantial performance gains, achieving improvements of 50.3%, 246.9%, and 110.7% compared to MAPPO, HAPPO, and HATRPO, respectively. When increasing the number of targets to 100 ($N_s = 3, N_t = 100$), MAT maintains its superior performance with improvements of 22.3%, 66.3%, and 55.5%. Similarly, in configurations with 5 satellites, MAT exhibits remarkable advantages - for 50 targets ($N_s = 5, N_t = 50$), it surpasses the baseline methods by 91.1%, 155.2%, and 122.0%, while for 100 targets ($N_s = 5, N_t = 100$), it achieves improvements of 38.0%, 64.5%, and 68.0%.

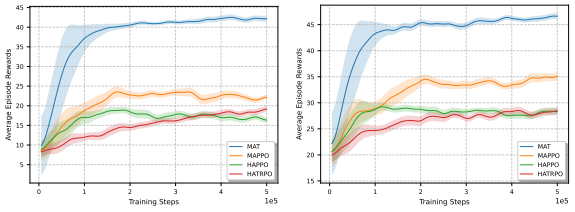
The exceptional performance of MAT can be attributed to two fundamental design principles. First, the sequential update scheme ensures robust stability and convergence properties, providing theoretical guarantees for monotonic improvement throughout the learning process. Second, the transformer architecture's sophisticated attention mechanisms enable MAT to effectively capture and process critical system-wide information, facilitating optimal decision-making across multiple agents.

4.2 Analysis of Stability

To rigorously evaluate the stability of MAT, we analyze the distribution entropy, a quantitative metric that characterizes the probability distribution across the action space. Higher values reflect broader exploration of the action space,



(a) $N_s = 3$ $N_t = 50$ (b) $N_s = 3$ $N_t = 100$

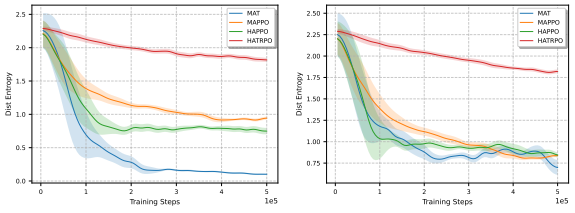


(c) $N_s = 5$ $N_t = 50$ (d) $N_s = 5$ $N_t = 100$

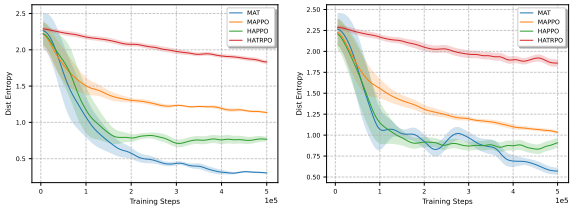
Fig. 5 Episode return comparison across different scenarios.

Table 1 Performance comparison of episode returns across different experimental scenarios.

Scenario	MAT	MAPPO	HAPPO	HATRPO
$N_s = 3, N_t = 50$	**40.46**	26.92	11.66	19.20
$N_s = 3, N_t = 100$	**43.08**	35.22	25.90	27.71
$N_s = 5, N_t = 50$	**42.16**	22.06	16.52	18.99
$N_s = 5, N_t = 100$	**46.54**	33.72	28.3	27.71



(a) $N_s = 3$ $N_t = 50$ (b) $N_s = 3$ $N_t = 100$



(c) $N_s = 5$ $N_t = 50$ (d) $N_s = 5$ $N_t = 100$

Fig. 6 Distribution entropy comparison across different scenarios.

whereas lower values signify the convergence toward deterministic policy decisions.

As illustrated in Fig. 6, while all methods exhibit similar entropy levels during the initial exploration phase, MAT achieves a significant reduction throughout the training process. This empirically validates MAT's stability in converging to stable, deterministic policies.

5. Conclusion

In this letter, we applied MAT to address the challenging AEOSCMP problem. Through innovative reformulation of joint policy optimization as a sequential decision process, MAT achieves remarkable improvements over existing MARL methods. Extensive experiments demonstrate that MAT not only significantly outperforms state-of-the-art baselines in observation effectiveness across different scenarios, but also exhibits superior stability during training and more reliable convergence properties. The compelling empirical results validate MAT as a powerful and practical solution for coordinating agile Earth observation satellite constellations, marking an important step forward in autonomous space mission planning.

Acknowledgments

This work is supported by the National Key Research and Development Program of China (2022YFB3902801).

References

[1] X. Wang, G. Wu, L. Xing, and W. Pedrycz, "Agile earth observation satellite scheduling over 20 years: Formulations, methods, and future directions," IEEE Systems Journal, vol.15, no.3, pp.3881–3892, 2020.

[2] M. Lemaıtre, 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.

[3] A. Herrmann, M. Stephenson, and H. Schaub, "Reinforcement learning for multi-satellite agile earth observing scheduling under various communication assumptions," AAS Rocky Mountain GN&C Conference, 2023.

[4] L. Dalin, W. Haijiao, Y. Zhen, G. Yanfeng, and S. Shi, "An online distributed satellite cooperative observation scheduling algorithm based on multiagent deep reinforcement learning," IEEE Geoscience and Remote Sensing Letters, vol.18, no.11, pp.1901–1905, 2020.

[5] J.G. Kuba, R. Chen, M. Wen, Y. Wen, F. Sun, J. Wang, and Y. Yang, "Trust region policy optimisation in multi-agent reinforcement learning," arXiv preprint arXiv:2109.11251, 2021.

[6] M. Wen, J. Kuba, R. Lin, W. Zhang, Y. Wen, J. Wang, and Y. Yang, "Multi-agent reinforcement learning is a sequence modeling problem," Advances in Neural Information Processing Systems, vol.35, pp.16509–16521, 2022.

[7] P.W. Kenneally, S. Piggott, and H. Schaub, "Basilisk: A flexible, scalable and modular astrodynamics simulation framework," Journal of aerospace information systems, vol.17, no.9, pp.496–507, 2020.

[8] J.G. Kuba, M. Wen, L. Meng, H. Zhang, D. Mguni, J. Wang, Y. Yang, *et al.*, "Settling the variance of multi-agent policy gradients," Advances in Neural Information Processing Systems, vol.34, pp.13458–13470, 2021.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," Advances in Neural Information Processing Systems, 2017.

[10] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu, "The surprising effectiveness of ppo in cooperative multi-agent games," Advances in Neural Information Processing Systems, vol.35, pp.24611–24624, 2022.

[11] J. Liu, Y. Zhong, S. Hu, H. Fu, Q. Fu, X. Chang, and Y. Yang, "Maximum entropy heterogeneous-agent reinforcement learning," The Twelfth International Conference on Learning Representations, 2024.

[12] Y. Zhong, J.G. Kuba, X. Feng, S. Hu, J. Ji, and Y. Yang, "Heterogeneous-agent reinforcement learning," Journal of Machine Learning Research, vol.25, no.1-67, p.1, 2024.