Joint HAP Access and LEO Satellite Backhaul in 6G: Matching Game-Based Approaches

Ziye Jia, Min Sheng, *Senior Member, IEEE*, Jiandong Li, *Senior Member, IEEE*, Di Zhou, *Member, IEEE*, and Zhu Han, *Fellow, IEEE*

***Abstract*— Space-air-ground networks play important roles in both fifth generation (5G) and sixth generation (6G) techniques. Low earth orbit (LEO) satellites and high altitude platforms (HAPs) are key components in space-air-ground networks to provide access services for the massive mobile and Internet of Things (IoT) users, especially in remote areas short of ground base station coverage. LEO satellite networks provide global coverage, while HAPs provide terrestrial users with closer, stable massive access service. In this work, we consider the cooperation of LEO satellites and HAPs for the massive access and data backhaul of remote area users. The problem is formulated to maximize the revenue in LEO satellites, which is in the form of mixed integer nonlinear programming. Since finding the optimal solution by exhaustive search is extremely complicated with a large scale of network, we propose a satellite-oriented restricted three-sided matching algorithm to deal with the matching among users, HAPs, and satellites. Furthermore, to tackle the dynamic connections between satellites and HAPs caused by the periodic motion of satellites, we present a two-tier matching algorithm, composed of the Gale-Shapley-based matching algorithm between users and HAPs, and the random path to pairwise-stable matching algorithm between HAPs and satellites. Numerical results show the effectiveness of the proposed algorithms.**

***Index Terms*— High altitude platform, low earth orbit satellite, space-air-ground network, Internet of Things, matching game.**

I. Introduction

The gradually matured fifth generation (5G) of communication system is finding more commercial applications and the research on the sixth generation (6G) technology is on the way [1]–[3]. Space-air-ground networks play important roles in both 5G and 6G, especially in remote areas, maritime applications, and emergency circumstances, where there are about three billion people in the world still have no regular connections to the Internet [4]–[6]. In addition, one goal in 6G is to realize the ubiquitous access services [7] for the massive mobile users and Internet of Things (IoT) users in the world [8], [9]. Either satellites or high-altitude platforms (HAPs) can extend network coverage to mobile users and IoT users in remote areas, hotspot areas and emergency areas who are uncovered or less covered by ground base stations (GBSs) [10]. Specifically, satellites involving geostationary earth orbit (GEO), medium earth orbit (MEO), and low earth orbit (LEO) satellites are basic components in space [11]. Compared with GEO and MEO satellites, LEO satellites have the advantages of lower cost of development and launch, lower delay, and are apt to form the LEO satellite networks leveraging the inter-satellite links (ISLs) [12]. HAPs are unmanned aircrafts including airplanes, airships and balloons, positioned above 20km altitude in the stratosphere, playing key roles in the air [13], [14]. Compared with LEO satellites, HAPs are more realistic for the massive terrestrial users in remote areas to access with stability and lower transmitting energy cost, since the direct connections between terrestrial users and satellites are intermittent and not stable due to the high-speed movement of satellites, and it also require the terrestrial users equipped with strong transmission power due to the long distance to satellites.

Recently, there is increasing enthusiasm on the research of both LEO satellites and HAPs from industry to academia. Companies such as SpaceX, Oneweb, O3B and Iridium Next are all devoted to develop LEO satellite networks [15], and SpaceX plans to launch more than 40,000 small satellites to provide global network service [16]. There are also several companies who are concentrating on the research and development of HAPs, for instance, Zephyr Platform of Airbus [17], Stratobus Platform of Thales [18], and Solar HAPS of HAPSMobile [19]. It is costly to build sufficient GBSs in remote areas, and there is no possibility to build GBSs in the ocean, while HAPs can provide stable massive access for mobile and IoT users in these areas. In addition, HAPs have the advantages of powerful payload ability and high altitude, wide coverage, and thus are cast as the stable base stations in the air. As supplements to the traditional satellite-ground networks, HAPs help form the space-air-ground networks. However, the received data in HAPs still need to be back to ground, but building specialized ground stations for HAPs is prohibitive or even impossible due to the geographical factors. The ideal channels between HAPs and LEO satellites are

Manuscript received January 31, 2020; revised June 8, 2020; accepted July 17, 2020. Date of publication August 24, 2020; date of current version March 17, 2021. This work was supported in part by the Natural Science Foundation of China under Grant U19B2025, Grant 61725103, Grant 61701363, and Grant 61931005; in part by the Doctoral Students' Short-Term Study Abroad Scholarship Fund of Xidian University; and in part by the U.S. NSF under Grant EARS-1839818, Grant CNS1717454, Grant CNS-1731424, and Grant CNS-1702850. *(Corresponding author: Min Sheng.)*

Ziye Jia, Min Sheng, Jiandong Li, and Di Zhou are with the State Key Laboratory of ISN, Xidian University, Xi'an 710071, China (e-mail: ziyejia@stu.xidian.edu.cn; msheng@mail.xidian.edu.cn; jdli@mail.xidian.edu.cn; zhoudi@xidian.edu.cn).

Zhu Han is with the University of Houston, Houston, TX 77004 USA, and also with the Department of Computer Science and Engineering, Kyung Hee University, Seoul 446-701, South Korea (e-mail: zhan2@uh.edu).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSAC.2020.3018824

good choices to serve as backhauls for HAPs. Further, LEO satellite networks are worldwide connected and can guarantee the collected data back to earth in an acceptable time.

In this article, we jointly consider the HAP-based massive access and LEO satellite-based backhaul to serve the mobile and IoT users in remote areas which lack GBSs and HAP-oriented ground stations. The problem is formulated to maximize the revenue (total data priorities) received by LEO satellites, since the users[1] with high priorities tend to provide more important information and pay more money for the service of HAPs and LEO satellites.[2] The problem is in the form of mixed integer non-linear programming (MINLP) and is NP-hard to obtain the optimal solution [20]. In addition, there is another challenge that the connections between HAPs and satellites are changing with time. We consider to divide the time horizon into several time slots and solve the problem in each independent time slot, because the relationships between HAPs and satellites can be deemed as quasi-static in each time slot. Additionally, compared with the high-dynamic connections between HAPs and satellites, the relative variation between terrestrial users and HAPs is slight and can be neglected. In addition, to directly solve the problem in the form of MINLP in each time slot is still intractable due to the NP-hardness [20]. Motivated by game theory, we consider to leverage the matching game based algorithms to tackle the problem to get the distributed near optimal solution with lower complexity.

As a Nobel Prize winning framework in Economic Science, matching game is an efficient distributed mechanism to deal with the social and marketing problem [21]. Recently, matching theory also finds its applications in network resource allocation, for example in the channel assignment in wireless scheduling [22], the unlicensed frequency allocation [23], and the resource assignment in wireless virtualized network [24]. The major advantage of matching is that it provides distributed solutions and considers the preference of each involved entity. There exist classical matching mechanism between two-sided entities [25], as well as the three-sided matching mechanism among three agents [26].

Intuitively inspired by the matching game framework, the problem for the relationships among the three agents (users, HAPs, and satellites) can be reformulated as a three-sided matching game with size and cyclic preference lists (TMSC) among the agents. The objective of TMSC is to find a stable matching among users, HAPs and satellites with the maximum cardinality. To efficiently obtain a stable matching result for the TMSC problem, we further propose the algorithm of satellite-oriented restricted three-sided matching with size and cyclic preference lists (R-TMSC) by adding two additional conditions for the preference lists of satellite and HAP. However, the stable matching result of the satellite-oriented R-TMSC algorithm from the previous time slot may lose the stability in the current time slot with the network topology changing, so the satellite-oriented R-TMSC algorithm needs to be executed repeatedly in each time slot. If the network topology is updated frequently and the time slot has a small length, the computing overhead and complexity of satellite-oriented R-TMSC algorithm in the whole time horizon will increase, especially in the case of a large amount of users. Note that although the network topology may change in two adjacent time slots, the change may be slight for the connections between HAPs and satellites, for example, one previous connected satellite leaving and one new connected satellite coming. As a result, the stable matching from the previous time slot will not change much and starting the satellite-oriented R-TMSC algorithm all over again seems redundant. To alleviate the redundancy of repeatedly executing satellite-oriented R-TMSC algorithm in the whole time horizon, we put forward a two-tier algorithm, involving the matching between users and HAPs, and the matching between HAPs and satellites. A Gale-Shapley (GS) based algorithm is designed for the matching between users and HAPs, and the matching result can be used for several time slots since the relationships between users and HAPs will maintain for a long period time due to the wide coverage of HAPs until the demands of users vary. With regard to the matching between HAPs and satellites, we present a random path to pairwise-stable matching (RPPS) based algorithm by leveraging the matching result in the previous time slot as the initialization in the current time slot. Then, the RPPS-based algorithm coverages to a new pairwise-stable matching rapidly especially when the network changes little. At this point, the satellite-oriented R-TMSC algorithm can be used to ensure the stable matching in the first time slot. Finally, we conduct simulations and experiments in real space-air-ground scenarios to check the performance of the proposed matching-based algorithms.

The main contributions of this work are summarized as follows:

- We propose the optimal system model of cooperation of HAP-based massive access and satellite-based backhaul for the data collection of mobile and IoT users in remote areas, which is an important framework in the space-air-ground integrated network in 6G.
- We reformulate the optimal problem into the TMSC framework and design the satellite-oriented R-TMSC algorithm to obtain the stable matching with maximum cardinality in each time slot, which is an efficient near-optimal solution for the original problem.
- To further alleviate the time complexity, we present a two-tier matching algorithm, including the GS-based matching algorithm for users and HAPs, and the RPPS-based algorithm for HAPs and satellites, leveraging the relative stability between users and HAPs and the slight change between HAPs and satellites in continuous time slots. The RPPS-based algorithm will converge to a pairwise-stable matching without redundancy by being initialized with the pairwise-stable matching in the previous time slot.
- We evaluate the effectiveness and efficiency for the proposed satellite-oriented R-TMSC and GS+RPPS algorithms under real satellite-HAP-ground networks by extensive simulations. The complexity of each proposed algorithm is also analyzed.

[1]In the rest of the paper, users in remote area include both mobile users and IoT users.

[2]Since we only consider LEO satellites in this article, "satellites" and "LEO satellites" are used interchangeably in the remaining of the paper.

The rest of the paper is organized as follows. Section II introduces the recent related works. Afterwards, the system model of cooperative LEO satellites and HAPs service for the mobile and IoT users in remote areas is elaborated and it is formulated as an optimization problem in Section III. Algorithms based on matching game to deal with the optimization problem are proposed in Section IV. Then in Section V, extensive simulation results and corresponding analyses are provided. Finally, conclusions are draw in Section VI.

II. Related Work

The satellite network related issues have been studied in some recent works. For example, [27] proposes an ultra-dense integrated LEO satellite-terrestrial network and the additional backhaul of LEO satellites extends the traditional network, aiming at maximizing the served users while satisfying the backhaul capacity constraint. In [12], the authors investigate the dynamic data scheduling problem in the small satellite networks, to maximize the total network rewards. A transmission beamforming mechanism is designed in [28], through the cooperative adaptive beamforming of satellite and terrestrial station. In [10], the authors explore the cooperation of unmanned aerial vehicles (UAVs) and satellites for space-air-ground IoT network.

In contrast with the dynamic motion of satellites, HAPs have the advantage of staying at one point for a long time, which can provide with a stable access station for terrestrial users. There are some recent researches with respect to HAPs. The authors in [29] investigate how to extend the joint transmission coordinated multipoint into the HAP system to enhance the capacity by exploiting the phased array antennas. Reference [30] presents a constellation design methodology for the broad network composed of HAPs that guarantee quality of service (QoS) and demand of users. A HAP-ground architecture is proposed in [31], where HAPs are used for caching and pushing contents to vehicles by the large area broadcasting of HAPs, to realize efficient resource allocation with guaranteed QoS. In [32], the millimeter wave used for HAPs are investigated to provide coverage to unserved areas.

Moreover, efforts have been devoted to deal with the cooperation and interplay between satellites and HAPs. For instance, in [33], the authors investigate the transmission strategies in integrated HAP-satellite networks for energy efficiency in emergency communications. Reference [34] proposes a HAP-relayed satellite security network for vehicles, which mostly focuses on the downlink services. In [35], a HAP-aided downlink satellite communication system is designed, aiming at minimizing the total energy cost.

There is also a wide usage of matching game in communication systems, particularly for the distributed resource optimization. For instance, in [23], the authors propose a Roth-Vande Vate (RVV) based matching framework to tackle the unlicensed resource allocation problem with the requirements from both LTE and Wi-Fi. In [36], a caching approach is proposed in the wireless small cell networks using the many-to-many matching game with the limited capacity of backhaul links into consideration, to obtain a pairwise-stable matching in limited steps. A many-to-many matching game based algorithm is designed in [37] to deal with the pairing problem between data service operators and fog nodes in IoT network system. Reference [24] adopts a three-sided matching with size and cyclic preference distributed suboptimal algorithm to obtain a practical solution among the cooperation and matching of contents, computation nodes, mobile virtual network operators and users.

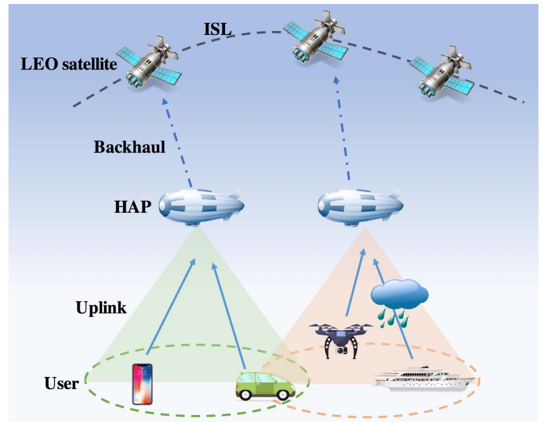


Fig. 1. A space-air-ground network scenario.

The above mentioned works have not considered the stable connections among users, HAPs and satellites, and the influence of satellite periodically dynamic property. Since the partner of HAPs in the related works are mostly the GEO satellites who are relatively static with earth and it is unnecessary to consider the effect of dynamic property of satellites. In addition, These works have not considered the data collection from remote area users. Even though there have been some previous works investigating the HAPs and LEO satellites, as far as the author's knowledge, distributed solutions based on matching game for revenue optimization in service for remote area users with HAPs and LEO satellites, has not been investigated before.

III. System Model and Formulation

We consider a cooperative space-air-ground network in Fig. 1, composed of LEO satellites in the space, HAPs in the air, and IoT users such as UAVs and vehicles, and mobile users such as individuals and maritime vessels in remote areas. When the data from users relayed by HAPs arrives on LEO satellites, it can be guaranteed back to earth through the ISLs among LEO satellites or through the GEO satellites by the store-carry-and-forward mechanism [12]. The key notations used in this article are listed in Table I.

A. Network Model

Fig. 1 is a scenario of space-air-ground network integrated of LEO satellites, HAPs and users in one-shot. Such users are connected directly to HAPs and HAPs are connected directly to LEO satellites. LEO satellites are moving periodically and may depart from the connections with HAPs with time variation. On the other hand, the HAP-LEO satellite connections will continue for a period of time and we regard such a period

TABLE I
KEY NOTATIONS

Symbol	Description
t, T, τ	Time slot, total number of time slot, and time slot length, $t \in T$.
$\mathcal{U}^t$, $\mathcal{H}^t$, $\mathcal{S}^t$	The set of users, HAPs, and satellites in time slot t.
u^t,h^t,s^t	The specific user $u^t \in \mathcal{U}^t$, HAP $h^t \in \mathcal{H}^t$, and LEO satellite $s^t \in \mathcal{S}^t$.
$w(h^t, s^t)$	Parameter to indicate the visibility of HAP h^t and satellite s^t.
α_{u^t}	Priority of user $u^t \in \mathcal{U}^t$.
γ_{uh}^t	SNR of U2H channel.
c_{uh}^t	Capacity of U2H channel.
$x_{u,h}^t$	Binary variable indicating whether user u^t is connected with HAP h^t.
$y_{h,s}^t$	Binary variable indicating whether HAP h^t is connected with LEO satellite s^t.
δ_{uh}^t	Required data rate from user u^t to HAP h^t.
N_{h^t}, N'_{h^t}, N_{s^t}	The capacity of HAP h^t for users, the capacity of HAP h^t for satellites, and the capacity of satellite s^t for HAPs.
$\mathscr{M}_1^t$, $\mathscr{M}_2^t$, $\mathscr{M}_3^t$	The matching among users, HAPs and satellites in satellite-oriented R-TMSC algorithm, the matching between users and HAPs in the GS-based algorithm, and the matching between HAPs and satellites in the RPPS-based algorithm.
$\mathscr{M}_1^t(z)$, $\mathscr{M}_2^t(z)$, $\mathscr{M}_3^t(z)$	The matched agents for z in the matching $\mathscr{M}_1^t$, $\mathscr{M}_2^t$, $\mathscr{M}_3^t$, respectively.
P_z^1, P_z^2, P_z^3	Preference lists of agent z in the matching $\mathscr{M}_1^t$, $\mathscr{M}_2^t$, $\mathscr{M}_3^t$, respectively.
$\mathcal{N}(\mathscr{M}_1^t, z)$	Number of triples including agent z in matching $\mathscr{M}_1^t$.
L_{h^t}, L_{s^t}	The maximum number of triples that HAP h^t takes part in $\mathscr{M}_1^t$, the maximum number of triples that satellite s^t takes part in $\mathscr{M}_1^t$.

as a time slot t and τ is the length of a time slot. In each time slot, the space-air-ground network topology is approximately deemed as quasi-static due to that both the connections of U2H and H2S have a duration time larger than τ. Denote by $\mathcal{U}^t = \{1^t, 2^t, \cdots, U^t\}$ the set of users in time slot t, $u^t \in \mathcal{U}^t$. The set of HAPs are given by $\mathcal{H}^t = \{1^t, 2^t, \cdots, H^t\}$, $h^t \in \mathcal{H}^t$. $\mathcal{S}^t = \{1^t, 2^t, \cdots, S^t\}$ indicates the set of LEO satellites, $s^t \in \mathcal{S}^t$. Parameter $w(h^t, s^t)$ indicates whether there exists an active link (light-of-sight transmission) between HAPS h^t and satellite s^t, 1 if existing, and 0 otherwise.

B. Channel Model

As the space-air-ground network topology in each time slot is quasi-static, the channel state information in each time slot can be regarded as changeless and it is not necessary to estimate the channel instantaneously. When the time slot changes, the corresponding channel state information will be accurately and timely updated according to the new parameters.

Basically, the channel capacity from user $u^t \in \mathcal{U}^t$ to HAP $h^t \in \mathcal{H}^t$ (U2H) stems from Shannon formula:

$$c_{uh}^t = B_{uh}^t \log_2(1 + \gamma_{uh}^t), \forall u^t \in \mathcal{U}^t, h^t \in \mathcal{H}^t, \tag{1}$$

in which B_{uh}^t is the available bandwidth and γ_{uh}^t is the signal noise ratio (SNR) at HAP h^t from user u^t. Since we only consider the applications of remote area users in this work, and compared with the urban environment where there are a great many of blocks, there exist few obstacles in the remote areas and we take no account of the small-scale fading caused by the multi-path effect. According to [33] and [38], taking the free space loss and rain attenuation into consideration as shown in the Fig. 1, γ_{uh}^t is calculated as:

$$\gamma_{uh}^t = \frac{P_{uh}^{tr} G_u^{tr} G_h^{re} L_r}{k_B T_s B_{uh}^t} \left(\frac{c}{4\pi d_{uh}^t \nu} \right)^2, \tag{2}$$

where P_{uh}^{tr} is the transmitting power of user $u^t \in \mathcal{U}^t$. G_u^{tr} and G_h^{re} are the antenna power gains of user transmitter and HAP receiver, respectively. L_r is the attenuation mostly stemmed from rain because the influence from rain attenuation is obvious when the centering frequency is greater than 10GHz and the detailed parameter can be acquired from [39]. k_B is the Boltzmann's constant and T_s is the system noise temperature. c is the speed of light and ν is the centering frequency. d_{uh}^t is the slant range in time slot t.

In accordance with [33], the channel between HAPs and satellites (H2S) is quasi-vacuum and it is almost an ideal channel with Gaussian Noise. Thus, the H2S channel has a much larger communication capacity than the U2H channel and we take no account of the H2S channel capacity restriction in this article.

C. Problem Formulation

Let binary variable $x_{u,h}^t$ denote whether user u^t is connected with HAP h^t, and variable $y_{h,s}^t$ denotes whether HAP h^t is connected with LEO satellite s^t, and the detailed definitions are as follows,

$$x_{u,h}^t = \begin{cases} 1, & \text{if user } u^t \text{ is conneted with HAP } h^t, \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

and

$$y_{h,s}^t = \begin{cases} 1, & \text{if HAP } h^t \text{ is conneted with LEO satellite } s^t, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

In $\mathbf{P0}$, the objective is to maximize the total revenue (priorities of successfully connected users) on satellites relayed by HAPs in the whole time slots T. Since users with high priorities are likely to provide more important data and pay more money to HAPs and satellites, both HAPs and satellites prefer to serve such users.

$$\mathbf{P0}: \max_{\mathbf{x},\mathbf{y}} \sum_{t \in T} \sum_{u^t \in \mathcal{U}^t} \sum_{h^t \in \mathcal{H}^t} \sum_{s^t \in \mathcal{S}^t} \alpha_{u^t} x_{u,h}^t y_{h,s}^t$$

$$\sum_{u^t \in \mathcal{U}^t} x_{u,h}^t \leq N_{h^t}, \forall h^t \in \mathcal{H}^t, \quad t \in T, \tag{5}$$

$$\sum_{h^t \in \mathcal{H}^t} x_{u,h}^t \leq 1, \quad \forall u^t \in \mathcal{U}^t, t \in T, \tag{6}$$

$$\delta_{uh}^t x_{u,h}^t \leq c_{uh}^t, \quad \forall u^t \in \mathcal{U}^t, h^t \in \mathcal{H}^t, t \in T, \tag{7}$$

$$\sum_{h^t\in\mathcal{H}^t} y_{h,s}^t \le N_{s^t}, \forall s^t \in \mathcal{S}^t, \quad t\in T, \tag{8}$$

$$\sum_{s^t\in\mathcal{S}^t} y_{h,s}^t \le N'_{h^t}, \forall h^t \in \mathcal{H}^t, \quad t\in T, \tag{9}$$

$$y_{h,s}^t \le w(h^t,s^t), \quad \forall h^t\in\mathcal{H}^t, s^t\in\mathcal{S}^t, t\in T, \tag{10}$$

$$x_{u,h}^t \in \{0,1\}, \quad \forall u^t\in\mathcal{U}^t, h^t\in\mathcal{H}^t, \ t\in T, \tag{11}$$

$$y_{h,s}^t \in \{0,1\}, \quad \forall h^t\in\mathcal{H}^t, s^t\in\mathcal{S}^t, t\in T, \tag{12}$$

in which $\mathbf{x} = \left\{x_{u,h}^t, \forall u^t\in\mathcal{U}^t, h^t\in\mathcal{H}^t, t\in T\right\}$ and $\mathbf{y} = \left\{y_{h,s}^t, \forall h^t\in\mathcal{H}^t, s^t\in\mathcal{S}^t, t\in T\right\}$.

When there is a large amount of users and the total required resources by users are greater than the resources provided by HAPs and satellites, HAPs and satellites should make choices according to the user priority α_{u^t}. Constraint (5) denotes the total number of users connected with HAP h^t should not exceed the capacity N_{h^t} of HAP h^t for users. In other words, the maximum number of users connecting to HAP h^t in time slot t is N_{h^t}. Constraint (6) indicates that each user can only connect to one HAP in time slot t. Constraint (7) represents the data rate δ_{uh}^t between user u^t and HAP h^t cannot exceed the U2H channel capacity. Constraint (8) denotes the total number of HAPs connected with satellite s^t cannot exceed the capacity N_{s^t} of satellite s^t for HAPs, i.e., the maximum number of HAPs connected to satellites^t in time slot t is N_{s^t}. Constraint (9) indicates that the total number of satellites connected with HAP h^t cannot exceed the capacity N'_{h^t} of HAP h^t for satellites, i.e., the maximum number of satellites connected with HAP h^t in a time slot t is N'_{h^t}. Constraint (10) represents that the connection of H2S is limited by the visibility of H2S.

It is noted that $\mathbf{P0}$ is a MINLP problem which is NP-hard to solve [20], and the solution is a tricky issue especially in the case of large scale space-air-ground networks with multiple time slots, since the exhaustive search is highly complex and cannot even guarantee a solution in an acceptable period of time. Hence, we consider the distributed methods depending on the matching game in Section IV.

IV. Algorithm Design and Analysis

In this section, we strive to solve $\mathbf{P0}$ using the matching game and we start by a general overview of matching game in Subsection IV-A. Then, combined with our problem, we propose a satellite-oriented R-TMSC algorithm to tackle the matching among users, HAPs and satellites in each time slot in Subsection IV-B. Further, to efficiently solve the dynamic connections between HAPs and satellites, we present a two-tier matching algorithm in Subsection IV-C, including the GS-based one-to-many matching between HAPs and users, and the algorithm of RPPS-based many-to-many matching between HAPs and satellites.

A. An Overview of Matching Game

Matching game, born in economics [40], has a widespread applications in marketing and social life. The common thread is that each agent in the matching game has a preference over another agent set, and the goal is to find a stable matching with respect to these preferences [21]. Basically, according to the number of participants, matching game can be classified into the two-sided matching and three-sided matching. In the two-sided matching game, it can be further divided into one-to-one matching, one-to-many matching, and many-to-many matching [25]. The typical matching game is used for the static framework, while the dynamic effect cannot be neglected due to the dynamic characteristics of real world and correspondingly, the dynamic game theory is developed [41].

B. Satellite-Oriented R-TMSC Algorithm

1) Basics of TMSC: Three-sided matching is very common in social domains, for example the supplier-firm-buyer problem [26]. Three-sided matching with size and cyclic preference is developed from the three-sided matching, and there are three types of matching agents considered as men (M), women (W) and dogs (D), respectively [42]. In TMSC, each type of agent has a preference list only including one type of other agents, i.e., men only rank women, women only rank dogs, and dogs only rank men, according to their own preferences P_m, P_w, P_d. Each entity can be matched to a limited number of another type of entity, due to the capacity constraint. Consider a matching $(m,w,d) \in \mathscr{M}$ among all the possible triples $\mathcal{T} = M\times W\times D$ of men, women and dogs, $\mathscr{M}(m) = w$, $\mathscr{M}(w) = d$, and $\mathscr{M}(d) = m$ indicate that woman $w\in W$ is acceptable to man $m\in M$, dog $d\in D$ is acceptable to woman $w\in W$, and man $m\in M$ is acceptable to dog $d\in D$. Intuitively, the optimization problems in multiple time slots can be tackled by regarding each individual time slot as a three-sided matching problem. Correspondingly, in our space-air-ground system, satellites $\mathcal{S}^t$, users $\mathcal{U}^t$, and HAPs $\mathcal{H}^t$ are cast as men, women, and dogs in TMSC, $\mathscr{M}_1^t \subseteq \mathcal{U}^t\times\mathcal{H}^t\times\mathcal{S}^t$. The main point in TMSC is to find a stable matching, and the blocking triple indicating whether TMSC achieves stability is defined as follows.

Definition 1 (Blocking Triple in TMSC): A triple $(s^t,u^t,h^t) \notin \mathscr{M}_1^t$ but $(s^t,u^t,h^t) \in \mathcal{T}$ is a blocking triple if there exists such a set:

$$\begin{aligned}&\left\{\mathscr{M}_1^t(s^t)=\emptyset \vee u^t \succ_{s^t} \mathscr{M}_1^t(s^t)\right\}\wedge\\ &\quad\left\{\mathscr{M}_1^t(u^t)=\emptyset \vee h^t \succ_{u^t} \mathscr{M}_1^t(u^t)\right\}\wedge\left\{\mathcal{N}(\mathscr{M}_1^t,h^t)\le L_{h^t}\right\},\end{aligned} \tag{13}$$

in which $u^t \succ_{s^t} \mathscr{M}_1^t(s^t)$ indicates that satellite s^t prefers user u^t to its current matched user $\mathscr{M}_1^t(s^t)$. Similarly, $h^t \succ_{u^t} \mathscr{M}_1^t(u^t)$ represents that user u^t prefers HAP h^t to its current matched HAP $\mathscr{M}_1^t(u^t)$. $\mathcal{N}(\mathscr{M}_1^t,h^t) \le L_{h^t}$ denotes that the total matching amount of HAP h^t should not exceed its capacity $L_{h^t} = N_{h^t}$. A matching $\mathscr{M}_1^t$ is said to be *stable* if there exists no blocking triple for $\mathscr{M}_1^t$.

In light of [43], in TMSC problem, it is aimed to find a matching $\mathscr{M}_1^t = \{(s^t,h^t,u^t)\}$ such that $s^t\in P_{h^t}^1$, $h^t\in P_{u^t}^1$, and $u^t\in P_{s^t}^1$ with the maximum cardinality,

$$\begin{aligned}\mathbf{P1}:\ &\max\ |\mathscr{M}_1^t|\\ &\text{s.t. } \mathcal{N}(\mathscr{M}_1^t,s^t)\le L_{s^t}, \forall s^t\in\mathcal{S}^t, u^t\in\mathcal{U}^t, h^t\in\mathcal{H}^t,\end{aligned} \tag{14}$$

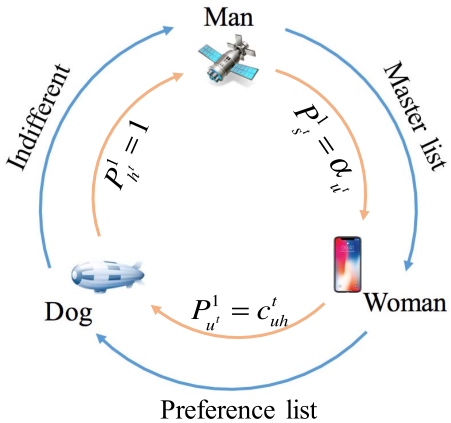


Fig. 2. Satellite-oriented R-TMSC game model for space-air-ground network.

$$\mathcal{N}(\mathscr{M}_1^t, h^t) \le L_{h^t}, \forall s^t \in \mathcal{S}^t, u^t \in \mathcal{U}^t, h^t \in \mathcal{H}^t, \tag{15}$$

$$\mathcal{N}(\mathscr{M}_1^t, u^t) \le 1, \forall s^t \in \mathcal{S}^t, u^t \in \mathcal{U}^t, h^t \in \mathcal{H}^t, \tag{16}$$

$$Bl(s^t, u^t, h^t) = 0, \forall s^t \in \mathcal{S}^t, u^t \in \mathcal{U}^t, h^t \in \mathcal{H}^t, \tag{17}$$

where $P^1_{h^t}$, $P^1_{u^t}$, and $P^1_{s^t}$ are the preference lists of HAP h^t, user u^t and satellite s^t in TMSC, respectively. $|\mathscr{M}_1^t|$ is the cardinality of a TMSC stable matching, and $\mathcal{N}(\mathscr{M}_1^t, z)$ denotes the number of triples including agent z in matching $\mathscr{M}_1^t$. Constraints (14-16) indicate the maximum number of triples that satellite s^t, HAP h^t, and user u^t participating in $\mathscr{M}_1^t$, respectively being $L_{s^t}(= N_{h^t} \cdot N_{s^t})$, L_{h^t}, and 1. Constraint (17) denotes that there exists no blocking triple in the final matching. Constraint (7) in problem **P0** is implied in user's preference list $P^1_{u^t}$. If the channel U2H from user u^t to HAP h^t is available, HAP h^t is in the user's preference list $P^1_{u^t}$, otherwise h^t is not in $P^1_{u^t}$.

According to [42], the determination for the existence of a stable matching in TMSC is NP-complete. To tackle such a difficulty, two restrictions are added in the preference lists in TMSC to form the R-TMSC, which is elaborated in the following.

2) R-TMSC Algorithm: From TMSC to R-TMSC, two restrictions are added. First, each satellite's preference list $P^1_{s^t}$ on users is from a master preference list, which means that all the users are sorted in a strict order and $P^1_{s^t}$ is obtained all or part from the master list. Furthermore, HAPs are indifferent with satellites, meaning that all satellites form a tie in the preference list of HAP $P^1_{h^t}$. Specifically, the preference lists $P^1_{s^t}$, $P^1_{u^t}$, and $P^1_{h^t}$ are created as follows. The satellite's preference list $P^1_{s^t}$ from a master list depends on the user's priority α_{u^t} in a descending order: $P^1_{s^t} = \alpha_{u^t}, \forall s^t \in \mathcal{S}^t$. The user's preference $P^1_{u^t}$ on HAPs is related to the capacity of U2H channel: $P^1_{u^t} = c^t_{uh}, \forall u^t \in \mathcal{U}^t$. It means that users prefer the HAPs with which the U2H channel has a greater capacity. Following the second restriction that HAPs are indifferent with satellites in R-TMSC, the preference list of HAPs on the accessible satellites is $P^1_{h^t} = 1, \forall h^t \in \mathcal{H}^t$. A vivid satellite-oriented R-TMSC game model for space-air-ground network is showed in Fig. 2, in which the cyclic relationships among satellites, users and HAPs are clearly illustrated.

Before the specific description of satellite-oriented R-TMSC algorithm, the following sets are defined:

$$A^{+1}(\mathscr{M}_1^t, s^t) = \{u^t | u^t \succ_{s^t} \mathscr{M}_1^t(s^t), u^t \in P^1_{s^t}\} \tag{18}$$

contains the set of users that satellite s^t prefers to its existing matching $\mathscr{M}_1^t(s^t)$.

$$A^{+1}(\mathscr{M}_1^t, u^t) = \{h^t | h^t \succ_{u^t} \mathscr{M}_1^t(u^t), h^t \in P^1_{u^t}\} \tag{19}$$

indicates the set of HAPs that user u^t prefers to its current matching $\mathscr{M}_1^t(u^t)$.

$$A^{-1}(\mathscr{M}_1^t, s^t) = \{h^t | h^t \in \mathcal{H}^t, s^t \in P^1_{h^t}, N(\mathscr{M}_1^t, h^t) \le L_{h^t}\} \tag{20}$$

denotes the set of HAPs that still can accommodate satellites.

$$A^{-2}(\mathscr{M}_1^t, s^t) = \{u^t | u^t \in \mathcal{U}^t, A^{+1}(\mathscr{M}_1^t, u^t) \cap A^{-1}(\mathscr{M}_1^t, s^t) \neq \emptyset\} \tag{21}$$

indicates that for user u^t, there is a HAP h^t that user u^t prefers to its current matching $\mathscr{M}_1^t(u^t)$ and h^t is still able to accommodate satellite s^t.

The specific satellite-oriented R-TMSC algorithm is listed in Algorithm 1.

In general, a satellite s^t is chosen to start and it selects the first user u^t in the preference list $P^1_{s^t}$, and then the selected user u^t chooses the first HAP in the preference list $P^1_{u^t}$. The indicator flag is used to control the algorithm execution, and $\mathrm{Head}(\mathrm{X}, z)$ indicates the element in X which has the highest priority in the preference list of z. To begin with, Algorithm 1 is initialized with $\mathscr{M}_1^t = \emptyset$. A better user u'^t for satellite s^t is searched in steps 5-7 of Algorithm 1. If such a user u'^t exists, a better HAP h'^t is being found for u'^t in steps 8-9. Next, the size of satellite s^t is checked and if $\mathcal{N}(\mathscr{M}_1^t, s^t) == L_{s^t}$, the worst matching triple $(s^t, \mathscr{M}_1^t(s^t), \mathscr{M}_1^t(\mathscr{M}_1^t(s^t)))$ in $\mathscr{M}_1^t$ is deleted. Analogously, the size of user u'^t is judged and processed in steps 15-18 of Algorithm 1. Then, the new formed triple (s^t, u'^t, h'^t) is added in $\mathscr{M}_1^t$. Steps 22-26 check if there still exists unmatched users when both satellites and HAPs are able to accommodate more users. Such a process is looped until the maximum cardinality $|\mathscr{M}_1^t|$ is achieved. As already noted, Algorithm 1 is applied to the stable matching of satellites, users, and HAPs in one time slot t. To get the stable matching $\mathscr{M}_1$ in all time slots T, Algorithm 1 is repeated in each time slot and $\mathscr{M}_1 = \bigcup_{t \in T} \mathscr{M}_1^t$. Algorithm 1 will output a stable matching in limited steps and the detailed proof is as follows in Theorem 1.

Theorem 1: The satellite-oriented R-TMSC algorithm will obtain a stable matching in finite iterations.

Proof: In Algorithm 1, the *while* loop stops when flag drops to 0. In each iteration of steps 8-19, user u'^t is matched to a better HAP in its preference list. When a matched user is deleted in step 16, a prioritized user is matched to a better HAP in step 19. Since the user amount and the number of HAPs in user's preference list are limited, the iteration will terminate in finite steps. Next, the final stable matching of Algorithm 1 is proved by contradiction. If the final matching

Algorithm 1 Satellite-Oriented R-TMSC Algorithm

Input: $\mathcal{S}^t$, $\mathcal{U}^t$, $\mathcal{H}^t$.
Output: $\mathscr{M}_1^t$.

Initialization: Construct the preference lists $P_{s^t}^1$, $P_{u^t}^1$, $P_{h^t}^1$. Set $\mathscr{M}_1^t = \emptyset$ and flag $= 1$.
while flag $== 1$ **do**
  Set flag $= 0$.
  for each satellite s^t **do**
    $\mathcal{U}'^t = A^{+1}(\mathscr{M}_1^t, s^t) \cap A^{-2}(\mathscr{M}_1^t, s^t)$.
    if $\mathcal{U}'^t \neq \emptyset$ **then**
      $u'^t = \text{Head}(\mathcal{U}'^t, s^t)$.
      $\mathcal{H}'^t = A^{+1}(\mathscr{M}_1^t, u'^t) \cup A^{-1}(\mathscr{M}_1^t, s^t)$.
      $h'^t = \text{Head}(\mathcal{H}'^t, u'^t)$.
      if $\mathcal{N}(\mathscr{M}_1^t, s^t) == L_{s^t}$ **then**
        Choose the worst matching for s^t in set $\{(s^t, \mathscr{M}_1^t(s^t), \mathscr{M}_1^t(\mathscr{M}_1^t(s^t)))\}$.
        $\mathscr{M}_1^t = \mathscr{M}_1^t \backslash \text{worst}\{(s^t, \mathscr{M}_1^t(s^t), \mathscr{M}_1^t(\mathscr{M}_1^t(s^t)))\}$.
        Set flag $= 1$.
      end if
      if $\mathcal{N}(\mathscr{M}_1^t, u'^t) == 1$ **then**
        $\mathscr{M}_1^t = \mathscr{M}_1^t \backslash (*, u'^t, \mathscr{M}_1^t(u'^t))$.
        Set flag $= 1$.
      end if
      $\mathscr{M}_1^t = \mathscr{M}_1^t \cup (s^t, u'^t, h'^t)$.
    end if
  end for
  if the unmatched users $\neq \emptyset$ **then**
    if $\mathcal{N}(\mathscr{M}_1^t, s^t) < L_{s^t}$ & $\mathcal{N}(\mathscr{M}_1^t, h^t) \leq L_{h^t}$ **then**
      Set flag $= 1$.
    end if
  end if
end while

is not stable, there still exists a blocking triple. It means that for a satellite s^t, there is a user u^{*t} in $P_{s^t}^1$ that $u^{*t} \succ_{s^t} \mathscr{M}_1^t(s^t)$, and there is a HAP h^{*t} in $P_{h^t}^1$ that $h^{*t} \succ_{u^{*t}} \mathscr{M}_1^t(u^{*t})$ and h^{*t} can still accommodate s^t. In other words, $A^{+1}(\mathscr{M}_1^t, s^t) \cap A^{-2}(\mathscr{M}_1^t, s^t) \neq \emptyset$ and $A^{+1}(\mathscr{M}_1^t, u^{*t}) \cup A^{-1}(\mathscr{M}_1^t, s^t) \neq \emptyset$, and Algorithm 1 will not terminate in such cases. Therefore, the matching result of Algorithm 1 is stable. ■

In light of [42], Algorithm 1 has a time complexity of $\mathcal{O}(|\mathcal{S}^t| \cdot |\mathcal{U}^t| \cdot \sum_{u^t \in \mathcal{U}^t} |P_{u^t}|)$, which is subjected to satellite amount, user amount and the length of user's preference list. In the worst case that all preference lists of users have a length of $|\mathcal{H}^t|$, the time complexity is $\mathcal{O}(|\mathcal{S}^t| \cdot |\mathcal{U}^t|^2 \cdot |\mathcal{H}^t|)$. Moreover, in view of the satellite-oriented R-TMSC algorithm is executed in each time slot, the total time complexity is $\mathcal{O}(T \cdot |\mathcal{S}^t| \cdot |\mathcal{H}^t| \cdot |\mathcal{U}^t|^2)$. Since the network topology may vary slightly in two adjacent time slots and the stable matching triples may change mildly, repeatedly executing Algorithm 1 in each time slot is redundant. It is not computationally efficient when T is large, and it is better to take advantage of the matching result in the previous time slot. We will tackle this issue in Subsection IV-C.

Algorithm 2 Two-Tier Matching Algorithm.

1: **for** $1 \leq t \leq T$ **do**
2:   **if** $t = 1$ **then**
3:     Get the stable matching among users, HAPs, and satellites by Algorithm 1.
4:   **else**
5:     **if** user's demand has no change in t compared with the demand in $t-1$ **then**
6:       Go to step 10.
7:     **else**
8:       Obtain the stable matching between users and HAPs by Algorithm 3.
9:     **end if**
10:    Using Algorithm 4 to get the pairwise-stable matching between HAPs and satellites.
11: **end if**
12: **end for**

C. Two-Tier Matching Algorithm

As already noted, to take advantage of the stable matching result in the previous time slot, we further explore the characteristics of three entities (users, HAPs, satellites). Practically, the connections between HAPs and satellites are changing with the periodic movement of satellites. In addition, due to the high altitude and stable position of HAPs and relative lower speed of users, the relative positions between HAPs and users can be approximately treated as static in a long period, which spans multiple time slots. Therefore, we consider a two-tier matching algorithm in Algorithm 2, including the GS-based many-to-one matching between users and HAPs in Algorithm 3 and the RPPS-based dynamic many-to-many matching between HAPs and satellites in Algorithm 4. Moreover, it is not necessary to perform Algorithm 3 in each time slot until the user demand changes. Afterwards, Algorithm 4 is executed dynamically with time slot varying. Additionally, in the first time slot, the initialized stable matching among users, HAPs and satellites is obtained by the satellite-oriented R-TMSC in Algorithm 1.

1) GS-Based Matching Algorithm for Users and HAPs: Instead of using the R-TMSC algorithm in each time slot, we can utilize the relationships between two adjacent time slots to simplify the complexity due to the small variation of network in adjacent time slots. In detail, the corresponding locations between terrestrial users and HAPs are approximately quasi-static. Hence, the matching between users and HAPs can deemed as no change in different time slots until user's demands vary. The stable matching $\mathscr{M}_2^t$ between users and HAPs can be obtained by Algorithm 3.

To begin with, the preference lists of users on HAPs are defined according to the U2H channel condition, $P_{u^t}^2 = c_{uh}^t, \forall u^t \in \mathcal{U}^t$, the same as in the R-TMSC algorithm. The preference list of HAPs on users are defined in accordance with the user's priority, $P_{h^t}^2 = \alpha_{u^t}, \forall h^t \in \mathcal{H}^t$. The blocking pair is defined as:

Definition 2 (Blocking Pair): in many-to-one matching: A pair $(u^t, h^t) \notin \mathscr{M}_2^t$ is a blocking pair for matching $\mathscr{M}_2^t$ if:

Algorithm 3 GS-Based Matching Between Users and HAPs.

Input: $\mathcal{U}^t$, $\mathcal{H}^t$.
Output: $\mathscr{M}_2^t$.

Initialization: Construct the preference lists $P_{u^t}^2$ and $P_{h^t}^2$. Set $\mathscr{M}_2^t = \emptyset$ and flag $= 1$.
while flag $== 1$ **do**
  flag $= 0$.
  for each unmatched user u^t **do**
    if $P_{u^t}^2 \neq \emptyset$ **then**
      Choose the best HAP $h^t \in P_{u^t}^2$ as $\mathscr{M}_2^t(u^t)$.
      if $|\mathscr{M}_2^t(h^t)| == N_{h^t}$ **then**
        Choose the worst matched user u'^t in $\mathscr{M}_2^t(h^t)$.
        if $u^t \succ_{h^t} u'^t$ **then**
          Swap u^t and u'^t in $\mathscr{M}_2^t(h^t)$.
        else
          Delete h^t in $P_{u^t}^2$.
        end if
        Set flag $= 1$.
      end if
      Add pair (u^t, h^t) in $\mathscr{M}_2^t$.
    end if
  end for
end while

1) u^t is unserved or prefers h^t to its current matching $\mathscr{M}_2^t(u^t)$; 2) h^t is underutilized or prefers u^t to at least one matching in $\mathscr{M}_2^t(h^t)$. $\mathscr{M}_2^t$ is thought as stable if there exist no blocking pair for $\mathscr{M}_2^t$.

Algorithm 3 starts from a user u^t (user-oriented) and it will select the most preferred HAP h^t in the preference list $P_{u^t}^2$. If the selected HAP h^t is undersubscribed, pair (u^t, h^t) is directly added to $\mathscr{M}_2^t$. On the other hand, if HAP h^t has no vacancy ($|\mathscr{M}_2^t(h^t)| == N_{h^t}$) and user u^t will be compared with the matched users $\mathscr{M}_2^t(h^t)$ of h^t. If user u^t is better than the worst matched user u'^t, u'^t and u^t are swapped. Algorithm 3 will be terminated when there is no blocking pairs or the preference list of all users are empty, and a stable matching $\mathscr{M}_2^t$ is acquired.

In terms of [44], the time complexity of Algorithm 3 is relevant with the number of possible user-HAP pairs, which equals to $|\mathcal{U}^t| \cdot |\mathcal{H}^t|$. In the worst case that user's demand changes in each time slot and Algorithm 3 is executed in time slots $[2, T]$, the time complexity is $\mathcal{O}((T-1) \cdot |\mathcal{U}^t| \cdot |\mathcal{H}^t|) = \mathcal{O}(T \cdot |\mathcal{U}^t| \cdot |\mathcal{H}^t|)$. In practice, it is unnecessary to perform Algorithm 3 in each time slot because the user's demand spans multiple time slots, especially when τ is small.

2) RPPS-Based Matching Algorithm for HAPs and Satellites: After obtaining the matching results between users and HAPs, we should deal with the matching between HAPs and satellites, which is a many-to-many matching problem. However, different with the matching between users and HAPs, the stable matching between HAPs and satellites in the previous time slot may be invalid due to the dynamic accessibility of H2S channel caused by satellite periodically movement. Therefore, a stable matching between HAPs and satellites should be reassigned to be stable in a new time slot. However, no matter the traditional many-to-many matching methods like [37] or the classical RVV algorithm [21], [41] for the dynamic one-to-one matching cannot be directly applied to this dynamic many-to-many problem.

We consider a random path to pairwise-stable matching method to tackle the foregoing challenges, inspired by [45] and [46]. To make the algorithm clear, a couple of new concepts for the matching of HAPs and satellites are introduced as follows.

Definition 3 (Substitutable Preference): Let $\mathcal{R}_s$ be the set of potential satellites (partners) for HAP h^t, $R_s \subseteq \mathcal{R}_s$. The preference list of HAP h^t is *substitutable* if for any satellites $s^t, s'^t \in F_{h^t}(R_s)$, $s^t \in F_{h^t}(R_s \backslash \{s'^t\})$. $F_{h^t}(R_s)$ denotes the subset of R_s that HAP h^t wishes to match to.
It essentially means that for HAP h^t, a satellite chosen from a larger potential satellite set is always chosen from a smaller potential satellite set [45].

Definition 4 (Responsive Preference): For each satellite s^t with maximum size N_{s^t}, its preference is *responsive* with quota N_{s^t} if: 1) for all $h^t \in \mathcal{J}_h$, and all $J_h \subseteq \mathcal{J}_h \backslash h^t$ with $|J_h| < N_{s^t}$, $h^t \cup J_h \succeq_{s^t} J_h \Leftrightarrow h^t \succeq_{s^t} \emptyset$; 2) for all $h^t, h'^t \in \mathcal{J}_h$, and all $J_h \subseteq \mathcal{J}_h \backslash \{h^t, h'^t\}$, with $|J_h| < N_{s^t}$, $h^t \cup J_h \succeq_{s^t} h'^t \cup J_h \Leftrightarrow h^t \succeq_{s^t} h'^t$; 3) for all $J_h \subseteq \mathcal{J}_h$ with $|J_h| > N_{s^t}$, $\emptyset \succ_{s^t} J_h$. $\mathcal{J}_h$ is the set of HAPs (partners) for satellite s^t, $J_h \subseteq \mathcal{J}_h$.

In other words, satellite s^t prefers adding an acceptable HAP unless reaching the quota (N_{s^t}) and it prefers replacing a HAP with a better one when the quota (N_{s^t}) is exhausted. Besides, satellite s^t prefers to being unmatched if any subset J_h exceeds the quota (N_{s^t}).

Definition 5 (Pairwise-Stable): A matching $\mathscr{M}_3^t$ is pairwise-stable if: 1) there is no *blocking individuals* h^t or s^t that $\mathscr{M}_3^t(h^t) \neq F_{h^t}(\mathscr{M}_3^t(h^t))$ or $\mathscr{M}_3^t(s^t) \neq F_{s^t}(\mathscr{M}_3^t(s^t))$; 2) there exists no *blocking pair* (h^t, s^t) with $h^t \notin \mathscr{M}_3^t(s^t)$ and $s^t \notin \mathscr{M}_3^t(h^t)$ such that $\mathcal{R}_s \in F_{h^t}\left(\mu(h^t, \mathscr{M}_3^t) \cup \{s^t\}\right)$ and $\mathcal{J}_h \in F_{s^t}\left(\mu(s^t, \mathscr{M}_3^t) \cup \{h^t\}\right)$, then $\mathcal{R}_s \succ_{h^t} \mu(h^t, \mathscr{M}_3^t)$ and $\mathcal{J}_h \succ_{s^t} \mu(s^t, \mathscr{M}_3^t)$. $\mu(i, \mathscr{M}_3^t)$ is the set of $j \in \mathcal{H}^t \cup \mathcal{S}^t$ such that $i \in \mathscr{M}_3^t(j)$ and $j \in \mathscr{M}_3^t(i)$.

It means that a matching $\mathscr{M}_3^t$ for HAPs and satellites is pairwise-stable if $\mathscr{M}_3^t$ is neither blocked individually nor blocked in pairs.

We continue to use the preference lists of HAPs on satellites in Algorithm 1, i.e., HAPs are indifferent with satellites $P_{h^t}^3 = 1, \forall h^t \in \mathcal{H}^t$, which satisfies the substitutable preference in Definition 3. The preference lists of satellites on HAPs are related with the matching result $\mathscr{M}_2^t$ between users and HAPs from Algorithm 3, i.e., $P_{s^t}^3 = |\mathscr{M}_2^t(h^t)|, \forall s^t \in \mathcal{S}^t$, in which $|\mathscr{M}_2^t(h^t)|$ is the total number of matched users for HAP h^t. In particular, $P_{s^t}^3$ reflects the coupling between Algorithm 3 and Algorithm 4. In other words, the matching result between HAPs and satellites is related with the matching result between users and HAPs. $P_{s^t}^3$ meets the conditions of responsive preference in Definition 4. When each HAP has a substitutable preference, and each satellite has a responsive preference with quota N_{s^t}, there exists a path of matching $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_K$ (blocking path) from a pairwise-unstable matching $\mathcal{M}_0$ to a pairwise-stable matching $\mathcal{M}_K$, in which $\mathcal{M}_{k+1}$ is turned from $\mathcal{M}_k$ by satisfying a blocking individual or a blocking pair.

Algorithm 4 RPPS-Based Matching between HAPs and Satellites.

Input: Matching result $\mathscr{M}_3^{t-1}$ between HAPs and satellites in time slot $t-1$. Updated preference lists $P_{h^t}^3$ and $P_{s^t}^3$.
Output: A pairwise-stable matching $\mathscr{M}_3^t$ in time slot t.
1: *Initialization:* $i=0$, $\mathcal{M}_i = \mathscr{M}_3^{t-1} \backslash M_{s^{t-1}}$, and $I=\emptyset$.
2: **for** $I \neq \mathcal{Q}^t$ **do**
3: Add an agent ι to I: $\bar{I} = I \cup \iota$.
4: **if** there are ι related blocking pairs $\{(\iota, b)\}$ **then**
5: **if** the quota of b is exhausted **then**
6: $\mathcal{M}_i = \mathcal{M}_i \backslash \text{Worst}\{(\mathcal{M}_i(b), b)\}$.
7: **end if**
8: $\mathcal{M}_i = \mathcal{M}_i \cup \text{Best}\{(\iota, b)\}$
9: $i=i+1$.
10: **else**
11: Go to Step 27.
12: **end if**
13: **while** there are any blocking pairs $\{(h^t, s^t)\}$ for HAP h^t in $\bar{I}$ **do**
14: **if** the quota of h^t is exhausted **then**
15: $\mathcal{M}_i = \mathcal{M}_i \backslash \text{Worst}\{(h^t, \mathcal{M}_i(h^t))\}$.
16: $\mathcal{M}_i = \mathcal{M}_i \cup \text{Best}\{(h^t, s^t)\}$.
17: $i=i+1$.
18: **end if**
19: **end while**
20: **while** there are any blocking pairs $\{(h^t, s^t)\}$ for satellite s^t in $\bar{I}$ **do**
21: **if** the quota of s^t is exhausted **then**
22: $\mathcal{M}_i = \mathcal{M}_i \backslash \text{Worst}\{(s^t, \mathcal{M}_i(s^t))\}$.
23: $\mathcal{M}_i = \mathcal{M}_i \cup \text{Best}\{(h^t, s^t)\}$
24: $i=i+1$.
25: **end if**
26: **end while**
27: Set $I=\bar{I}$.
28: **end for**
29: $\mathscr{M}_t^3 = \mathcal{M}_i$.

Such a blocking path to pairwise-stability is a convergent path and avid readers are referred to [45] for the proof in detail.

The RPPS-based matching algorithm between HAPs and satellites are specified in Algorithm 4. The input for Algorithm 4 is the pairwise-stable matching $\mathscr{M}_3^{t-1}$ from the previous time slot $t-1$ when $t \geq 2$, which is no longer pairwise-stable in the current time slot t. In addition, since the accessible satellites for each HAP may change, the preference lists $P_{h^t}^3$ and $P_{s^t}^3$ are updated in line with the current connections between HAPs and satellites. Algorithm 4 is initialized with matching $\mathcal{M}_i = \mathscr{M}_3^{t-1} \backslash M_{s^{t-1}}$, and an *internally stable* set $I=\emptyset$, in which $M_{s^{t-1}} \subseteq \mathscr{M}_3^{t-1}$ is the partial matching related with satellites $\{s^{t-1}\}$ that $\{s^t\}$ are no longer in $P_{h^t}^3$ due to satellite movement. *Internally stable* means that the agents in set I are not matched with the agents out of I. Apparently, $\emptyset$ is an internally stable set. Let $\mathcal{Q}^t = \mathcal{H}^t \cup \mathcal{S}^t$ denote the set of all agents of HAPs and satellites. As long as $I \neq \mathcal{Q}^t$, an new agent (HAP or satellite) ι is added to I, i.e., $\bar{I} = I \cup \iota$. When there are blocking pairs $\{(\iota, b)\}$ in $\bar{I}$ involving ι, then ι is matched with its most preferred partner Best $\{(\iota, b)\}$ in $\bar{I}$, as steps 4-12 of Algorithm 4. Then, check if there are any HAP-pointed blocking pairs $\{(h^t, s^t)\}$ in $\bar{I}$ and let HAP h^t match with its most preferred satellite in $\{(h^t, s^t)\}$, as listed in lines 13-19 of Algorithm 4. Next, if there are any satellite-pointed blocking pairs $\{(h^t, s^t)\}$ in $\bar{I}$ and let satellite s^t match with its most preferred HAP in $\{(h^t, s^t)\}$, as shown in lines 20-26 of Algorithm 4. After each iteration, I is updated as $I=\bar{I}$, and I is internally stable under matching $\mathcal{M}_i$. Finally, the pairwise stable matching in time slot t is updated as $\mathscr{M}_t^3 = \mathcal{M}_i$.

To analyze the time complexity, we focus on the three loops in Algorithm 4. The outer loop is related with the number of the whole agent $\mathcal{Q}^t$. For the loop from step 13 to step 19, the worst case is that each HAP searches the satellites from the bottom of the preference list and the complexity is based on $|\mathcal{H}^t| \cdot |\mathcal{S}^t|$. Similarly, the complexity of loop from step 20 to step 26 depends on $|\mathcal{S}^t| \cdot |\mathcal{H}^t|$. Hence, the time complexity of Algorithm 4 is $\mathcal{O}\left(\mathcal{Q}^t \cdot (|\mathcal{H}^t| \cdot |\mathcal{S}^t| + |\mathcal{S}^t| \cdot |\mathcal{H}^t|)\right) = \mathcal{O}\left(|\mathcal{H}^t|^2 \cdot |\mathcal{S}^t| + |\mathcal{H}^t| \cdot |\mathcal{S}^t|^2\right)$. To obtain the pairwise-stable matching of HAPs and satellites in all time slots, Algorithm 4 is executed in each time slot $t > 2$, and the total time complexity is $\mathcal{O}\left((T-1) \cdot \left(|\mathcal{H}^t|^2 \cdot |\mathcal{S}^t| + |\mathcal{H}^t| \cdot |\mathcal{S}^t|^2\right)\right) = \mathcal{O}\left(T \cdot \left(|\mathcal{H}^t|^2 \cdot |\mathcal{S}^t| + |\mathcal{H}^t| \cdot |\mathcal{S}^t|^2\right)\right)$. The total time complexity of cooperation of Algorithm 3 and Algorithm 4 in all time slots is related to the time complexity $\mathcal{O}(T \cdot |\mathcal{U}^t| \cdot |\mathcal{H}^t|)$ and $\mathcal{O}\left(T \cdot \left(|\mathcal{H}^t|^2 \cdot |\mathcal{S}^t| + |\mathcal{H}^t| \cdot |\mathcal{S}^t|^2\right)\right)$, equal to $\mathcal{O}(T \cdot \left(|\mathcal{U}^t| \cdot |\mathcal{H}^t| + |\mathcal{H}^t|^2 \cdot |\mathcal{S}^t| + |\mathcal{H}^t| \cdot |\mathcal{S}^t|^2\right))$. Pay attention that in a real scenario, the number of HAPs and satellites are fixed, much less than the number of users, and therefore the time complexity is mostly decided by the number of users $|\mathcal{U}^t|$. Distinguished from the complexity $\mathcal{O}(T \cdot |\mathcal{S}^t| \cdot |\mathcal{H}^t| \cdot |\mathcal{U}^t|^2)$ of Algorithm 1 which is mostly decided by $|\mathcal{U}^t|^2$, $\mathcal{O}(T \cdot \left(|\mathcal{U}^t| \cdot |\mathcal{H}^t| + |\mathcal{H}^t|^2 \cdot |\mathcal{S}^t| + |\mathcal{H}^t| \cdot |\mathcal{S}^t|^2\right))$ has a linear complexity with $|\mathcal{U}^t|$. From this point, algorithm GS+RPPS performs better than the satellite-oriented R-TMSC algorithm in the case of large number of users.

Additionally, the hidden reason on the complexity decreasing of the matching game based distributed algorithms lies in that only control signals of preference list and and the resource state information related with the preference lists need to be sent to a certain computing server. Compared with the global information of the system for the centralized mechanism, the signaling is limited and the the computing complexity is greatly decreased.

V. Performance Results and Analysis

In this section, We conduct a series of simulations in a practical scenario composed of users, HAPs, and satellites. The satellite orbit parameters, positions of HAPs, and the connection relationships among users, HAPs and satellites are generated from Satellite Tool Kit (STK) and then put into MATLAB. MATLAB is employed to implement the proposed algorithms and output corresponding results.

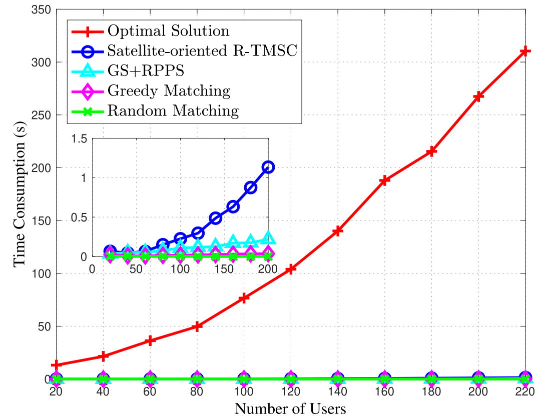
A. Parameter Setting

The main simulation parameters are listed in Table II. We use the Iridium-liked LEO satellite constellation in space.

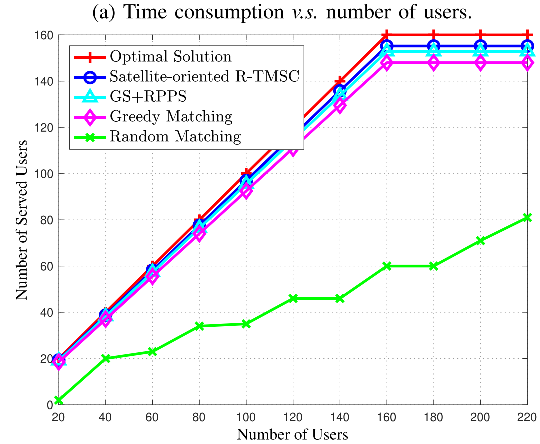
TABLE II
PRIMARY SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
$\lvert\mathcal{S}^t\rvert$	66	N'_{h^t}	2
Orbit number	6	N_{s^t}	2
Orbit height	780km	P^{tr}_{uh}	1W
Orbit period	100min	$G^{tr}_u \cdot G^{re}_h$	41dB
Constellation type	Polar-obit	L_r	-23dB
$\lvert\mathcal{H}^t\rvert$	4	T_s	300K
HAP altitude	20km	ν	28GHz
$\lvert\mathcal{U}^t\rvert$	[20,220]	B^t_{uh}	20MHz
τ	200s	d^t_{uh}	[20, 50] km
T	30	α_{u^t}	$[1, \lvert\mathcal{U}^t\rvert]/\lvert\mathcal{U}^t\rvert$
N_{h^t}	40	δ^t_{uh}	2Mbps

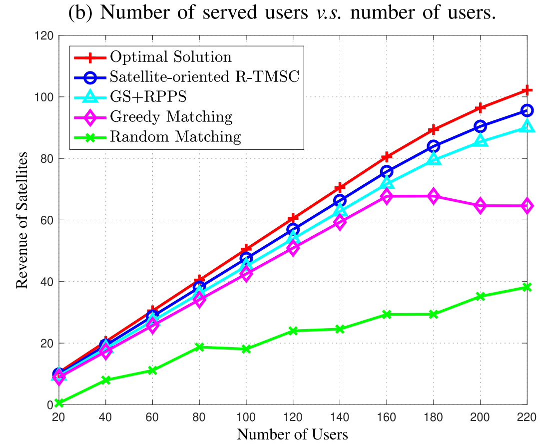
In detail, 66 satellites are uniformly distributed in 6 polar-obit planes, with the orbit height 780km and the orbit period 6028s. Such a satellite constellation can guarantee that any HAP can be connected with at least one satellite in any time. There are 4 HAPs are fixed at the altitude of 20km in the air, uniformly distributed in the area of latitude $[25°N, 50°N]$ and longitude $[73°E, 93°E]$, mostly covering the depopulated area of China [30]. The radius of HAP's coverage is set as 46km, and the possible distance range between users and HAPs is calculated as $d^t_{uh} = [20, 50]$ km. Referring to [47] and [39], the U2H channel related parameters are set as $P^{tr}_{uh} = 1$W, $G^{tr}_u \cdot G^{re}_h = 41$dB, $L_r = -23$dB, $T_s = 300$K, $B^t_{uh} = 20$MHz, and $\nu = 28$GHz. We consider a time horizon of 100min from Oct.15 2019 06:00:00 UTCG to Oct.15 2019 07:40:00 UTCG, with time slot length $\tau = 200s$, and in each time slot the user number $|\mathcal{U}^t|$ changes in the range of $[20, 220]$ and each user has a normalized priority from $[1, |\mathcal{U}^t|]/|\mathcal{U}^t|$. In addition, in each time slot t, the movement distance of users is far less than the diameter 92km of HAP's coverage, and thus these users are still in the coverage of HAPs, and the distances of U2H at the start of the time slot t and at the end of the time slot t are approximately equal. Also, due to the high altitude of HAPs, d^t_{uh} is approximately deemed as a constant in the same time slot, the U2H channel is also regarded as unchangeable within one time slot. When the time slot changes, the channel model will be recalculated and updated according to the parameters in the new time slot. The capacity of HAP for users and satellites are respectively $N_{h^t} = 40$ and $N'_{h^t} = 2$. The capacity of satellite for HAPs is $N_{s^t} = 2$.



(a) Time consumption *v.s.* number of users.



(b) Number of served users *v.s.* number of users.



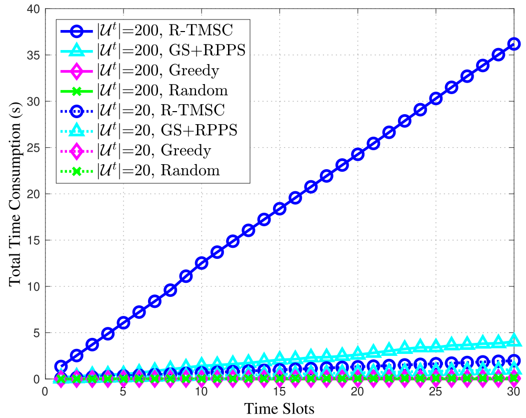
(c) Revenue of satellites *v.s.* number of users.

Fig. 3. Time complexity and output matching results in one time slot by different algorithms.

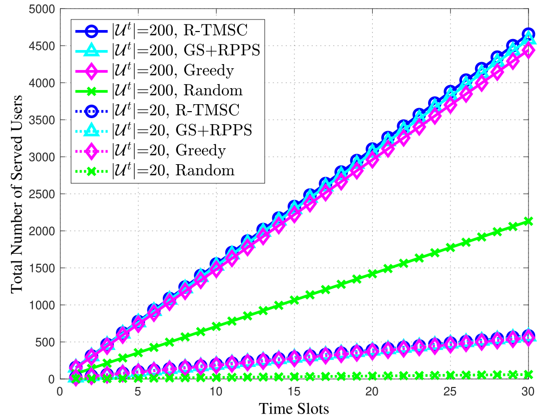
B. Performance Evaluation

To evaluate the performance of satellite-oriented R-TMSC algorithm and GS+RPPS algorithm, we compare the algorithms with the results of exhaustive search for problem **P0**, which serves as an optimal bench mark. Additionally, a greedy matching and a random matching among users, HAPs and satellites are also provided as reference comparisons.

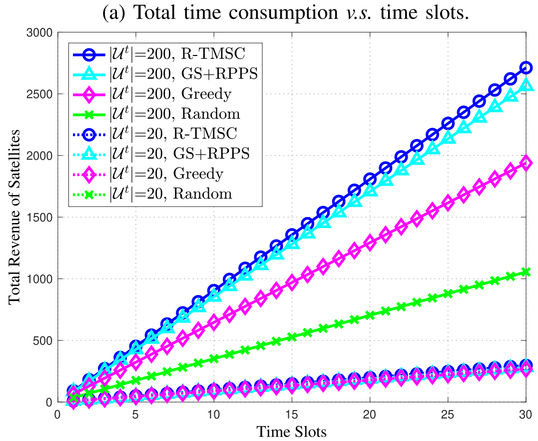
In Fig. 3, we first analyze the effectiveness and time complexity of the proposed algorithms in one time slot, caused by the number of users. In addition, the initial matching for RPPS is set empty in one time slot. The optimal solution is obtained by brute force searching, and apparently in Fig. 3a, it has high computing time consumption with exponential complexity with the increment of number of users, while the time consumption of other methods are in an acceptable range. It can be further noticed that when the number of users is more than 170, the execution time of optimal solution is more than 200s, which is completely unworkable due to that it is greater that the length of a time slot $\tau = 200$s. Both satellite-oriented R-TMSC and GS+RPPS algorithms have a low time complexity and GS+RPPS is better than R-TMSC especially in the case of a great number of users. Meanwhile, Fig. 3b shows that with respect to the total number of served users, satellite-oriented R-TMSC and GS+RPPS algorithms can achieve



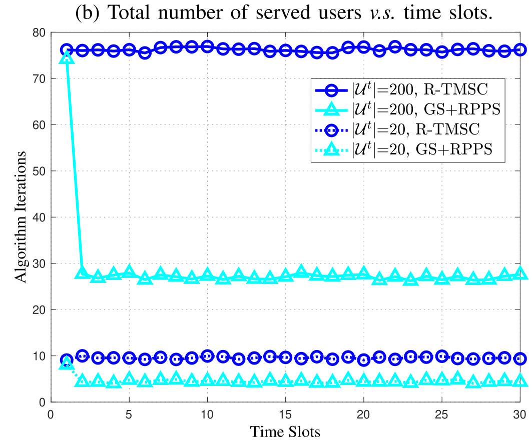
(a) Total time consumption *v.s.* time slots.



(b) Total number of served users *v.s.* time slots.



(c) Total revenue of satellites *v.s.* time slots.



(d) Algorithm iterations in different time slots.

Fig. 4. Time complexity and matching results in multi time slots by different algorithms.

near-optimal effects and R-TMSC performs a little better than GS+RPPS. It is also observed in Fig. 3b that the number of served users levels off no matter in optimal solution, or the satellite-oriented R-TMSC algorithm, GC+RPPS, and greedy matching when the number of users is larger than 160, and this is accounted by the fact that the HAP's capacity for accessible users becomes bottleneck. In addition, the results from random matching performs poorly compared with other mechanisms. The reason of the continually increasing number of served users within random matching in Fig. 3b lies in that both the capacity of HAPs and satellites are far from saturation. With regards to the revenue of satellites in Fig. 3c, there is a gradually increasing trend for all algorithms. However, as compared with Fig. 3b, the revenue still increases except for the greedy matching even the total number of served users remains unchanged in Fig. 3b. The reason lies in that the total number of users are increasing and new users may have higher priorities which promote the consecutive growth of total revenues. Furthermore, we can observe that the upward trend of total revenue stops for the greedy matching, because it stops checking priorities of the unmatched users when the capacity of HAPs is exhausted in greedy matching. From Fig. 3, it is confirmed that the algorithms of satellite-oriented R-TMSC and GC+RPPS can obtain a satisfied solution with an acceptable time cost in one time slot in which the network topology is quasi-static.

We further study the performance of the proposed algorithms for the dynamic network with satellites periodic motion in Fig. 4 and we evaluate the effects of dynamic by the varying of time slots. To make it clear, we consider two sets of different number of users ($|\mathcal{U}^t| = 20$, $|\mathcal{U}^t| = 200$) and assume in each time slot the demands of users have no change for fair comparisons. In Fig. 4a, for the total time consumption variation with the time slots increment, there is an obvious growing trend of satellite-oriented R-TMSC algorithm especially for a large number of users, while the GS+RPPS has a much better performance than R-TMSC. We also study the total number of served users and the total revenue of satellites versus time slots increasing in Fig. 4b and Fig. 4c. The growing trends are expected due to the accumulated successful matched users are increasing, and the performance of GS+RPPS algorithm is extremely close to the efficacy of satellite-oriented R-TMSC algorithm. Although the performance of greedy algorithm seems acceptable for the number of served users in Fig. 4b, it is unacceptable because the total revenue of satellites under greedy algorithm in Fig. 4c has great gaps compared with both satellite-oriented R-TMSC and GS+RPPS algorithms. Furthermore, to clearly show the efficiency of GS+RPPS algorithm, Fig. 4d depicts the algorithm iterations in each time slot for algorithms R-TMSC and GS+RPPS. As expected, the GS+RPPS algorithm has a smaller size of iteration compared with the R-TMSC algorithm, and this is the underlying rationale that GS+RPPS has a lower time consumption. Another observation is that the algorithm iteration of GS+RPPS changes dramatically from the first time slot to the second time slot and subsequently, the algorithm iteration stays almost the same level in the following time slots. Such an appearance can be accounted by the fact that the pairwise-stable matching result from the previous time slot is well leveraged by GS+RPPS algorithm. Specifically, due to that the real connections between HAPs and satellites change slightly in adjacent time slots and the GS+RPPS algorithm just needs to slightly revise the pairwise-stable matching result from the previous time slot to obtain an updated pairwise-stable matching in the current time slot. Such a simulation result attributes to the algorithm analysis as in Subsection IV-C. In addition, the algorithm iteration of GS+RPPS in the first time slot is as much as the satellite-oriented R-TMSC algorithm, and this is due to that the pairwise-stable matching in the first time slot is obtained from the initialization of empty set, while the the initialization of GS+RPPS algorithm in time slots $t > 2$ is initialized by the pairwise-stable matching of the previous time slot $t - 1$.

In terms of the simulation results and corresponding analyzes, the satellite-oriented R-TMSC algorithm and GS+RPPS algorithm have analogous performance in small size of time slots. GS+RPPS algorithm is superior to R-TMSC algorithm with lower time complexity in large size of time slots but with a certain performance loss, while satellite-oriented R-TMSC algorithm keeps the satisfied matching results but with higher time complexity with the increment of time slots. For practice usage, it can be chosen from the proposed algorithms based on the specific scenario and performance metrics.

VI. Conclusion

In this article, we have dealt with the cooperation of HAP-based massive access and satellite-based backhaul in the space-air-ground networks. Specifically, we first present a total revenue maximization problem and due to the high complexity of exhausting search, we further introduce the matching game to solve the problem. Intuitively, the three-sided matching for three types of agent is suitable for the problem among satellites, HAPs and users, and therefore we propose the corresponding satellite-oriented R-TMSC algorithm to obtain the stable matching. Furthermore, to tackle the complexity brought by dynamic connections between HAPs and satellites, we present the GS+RPPS algorithm which can efficiently leverage the matching result from the previous time slot. Finally, extensive simulations are conducted and results show the effectiveness and efficiency of the proposed algorithms. The satellite-oriented R-TMSC algorithm guarantees the revenue of satellites as well as the number of served users while consume acceptable execution time for several continuous time slots. The GS+RPPS algorithm can obtain satisfied revenue for satellites and number of served users with faster execution time even within a large amount of continuous time slots. According to the different performance of the proposed algorithms under different scenario setup, we can select the best algorithm for different practice usage.

References

[1] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Netw.*, vol. 34, no. 3, pp. 134–142, May 2020.

[2] Z. Zhang *et al.*, "6G wireless networks: Vision, requirements, architecture, and key technologies," *IEEE Veh. Technol. Mag.*, vol. 14, no. 3, pp. 28–41, Sep. 2019.

[3] Q. Qi, X. Chen, L. Lei, C. Zhong, and Z. Zhang, "Outage-constrained robust design for sustainable B5G cellular Internet of Things," *IEEE Trans. Wireless Commun.*, vol. 18, no. 12, pp. 5780–5790, Dec. 2019.

[4] J. Liu, Y. Shi, Z. M. Fadlullah, and N. Kato, "Space-air-ground integrated network: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2714–2741, May 2018.

[5] M. Sheng, Y. Wang, J. Li, R. Liu, D. Zhou, and L. He, "Toward a flexible and reconfigurable broadband satellite network: Resource management architecture and strategies," *IEEE Wireless Commun.*, vol. 24, no. 4, pp. 127–133, Aug. 2017.

[6] J. Qiu, D. Grace, G. Ding, M. D. Zakaria, and Q. Wu, "Air-ground heterogeneous networks for 5G and beyond via integrating high and low altitude platforms," *IEEE Wireless Commun.*, vol. 26, no. 6, pp. 140–148, Dec. 2019.

[7] L. Liu, E. G. Larsson, W. Yu, P. Popovski, C. Stefanovic, and E. de Carvalho, "Sparse signal processing for grant-free massive connectivity: A future paradigm for random access protocols in the Internet of Things," *IEEE Signal Process. Mag.*, vol. 35, no. 5, pp. 88–99, Sep. 2018.

[8] G. Yu, X. Chen, and D. W. K. Ng, "Low-cost design of massive access for cellular Internet of Things," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 8008–8020, Nov. 2019.

[9] D. W. K. Ng, T. Q. Duong, C. Zhong, and R. Schober, *Spectral and Energy-Efficient Wireless-Powered IoT Networks*. Hoboken, NJ, USA: Wiley, 2019, pp. 233–251.

[10] J. Wang, C. Jiang, Z. Wei, C. Pan, H. Zhang, and Y. Ren, "Joint UAV hovering altitude and power control for space-air-ground IoT networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1741–1753, Apr. 2019.

[11] I. Ali, P. G. Bonanni, N. AI-Dhahir, and J. E. Hershey, *Doppler Applications in LEO Satellite Communication Systems*. New York, NY, USA: Springer, 2002.

[12] D. Zhou, M. Sheng, J. Luo, R. Liu, J. Li, and Z. Han, "Collaborative data scheduling with joint forward and backward induction in small satellite networks," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3443–3456, May 2019.

[13] S. Karapantazis and F. Pavlidou, "Broadband communications via high-altitude platforms: A survey," *IEEE Commun. Surveys Tuts.*, vol. 7, no. 1, pp. 2–31, 1st Quart., 2005.

[14] J. Gonzalo, D. López, D. Domínguez, A. García, and A. Escapa, "On the capabilities and limitations of high altitude pseudo-satellites," *Prog. Aerosp. Sci.*, vol. 98, pp. 37–56, Apr. 2018.

[15] G. Giambene, S. Kota, and P. Pillai, "Satellite-5G integration: A network perspective," *IEEE Netw.*, vol. 32, no. 5, pp. 25–31, Sep. 2018.

[16] (Oct. 2019). [Online]. Available: https://www.newscientist.com/article/2220346-spacex-plans-to-put-more-than-400%00-satellites-in-space/

[17] (Aug. 2018). [Online]. Available: https://www.airbus.com/defence/uav/zephyr.html

[18] (Mar. 2017). [Online]. Available: https://www.thalesgroup.com/en/worldwide/space/news/whats-stratobus

[19] (Dec. 2017). [Online]. Available: https://www.hapsmobile.com/en/

[20] V. V. Vazirani, *Approximation Algorithms*. Springer, Mar. 2013.

[21] D. F. Manlove, *Algorithmics of Matching Under Preferences*. Singapore: World Scientific, Apr. 2013.

[22] A. Leshem, E. Zehavi, and Y. Yaffe, "Multichannel opportunistic carrier sensing for stable channel access control in cognitive radio systems," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 1, pp. 82–95, Jan. 2012.

[23] Y. Gu, C. Jiang, L. X. Cai, M. Pan, L. Song, and Z. Han, "Dynamic path to stability in LTE-unlicensed with user mobility: A matching framework," *IEEE Trans. Wireless Commun.*, vol. 16, no. 7, pp. 4547–4561, Jul. 2017.

[24] Z. Miao, Y. Wang, and Z. Han, "A supplier-firm-buyer framework for computation and content resource assignment in wireless virtual networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 8, pp. 4116–4128, Aug. 2019.

[25] D. Gale and L. S. Shapley, "College admissions and the stability of marriage," *Amer. Math. Monthly*, vol. 69, no. 1, pp. 9–15, Jan. 1962.

[26] C. Ng and D. Hirschberg, "Three-dimensional stable matching problems," *SIAM J. Discrete Math.*, vol. 4, no. 2, pp. 245–252, Aug. 1991.

[27] B. Di, H. Zhang, L. Song, Y. Li, and G. Y. Li, "Ultra-dense LEO: Integrating terrestrial-satellite networks into 5G and beyond for data offloading," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 47–62, Jan. 2019.

[28] J. Du, C. Jiang, H. Zhang, X. Wang, Y. Ren, and M. Debbah, "Secure satellite-terrestrial transmission over incumbent terrestrial networks via cooperative beamforming," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 7, pp. 1367–1382, Jul. 2018.

[29] M. D. Zakaria, D. Grace, P. D. Mitchell, T. M. Shami, and N. Morozs, "Exploiting user-centric joint transmission—Coordinated multipoint with a high altitude platform system architecture," *IEEE Access*, vol. 7, pp. 38957–38972, 2019.
[30] F. Dong, H. Han, X. Gong, J. Wang, and H. Li, "A constellation design methodology based on QoS and user demand in high-altitude platform broadband networks," *IEEE Trans. Multimedia*, vol. 18, no. 12, pp. 2384–2397, Dec. 2016.
[31] S. Zhang, W. Quan, J. Li, W. Shi, P. Yang, and X. Shen, "Air-ground integrated vehicular network slicing with content pushing and caching," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2114–2127, Sep. 2018.
[32] S. Dutta, F. Hsieh, and F. W. Vook, "HAPS based communication using mmWave bands," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.
[33] F. Dong, H. Li, X. Gong, Q. Liu, and J. Wang, "Energy-efficient transmissions for remote wireless sensor networks: An integrated HAP/satellite architecture for emergency scenarios," *Sensors*, vol. 15, no. 9, pp. 22266–22290, Sep. 2015.
[34] M. Q. Vu, N. T. Dang, and A. T. Pham, "HAP-aided relaying satellite FSO/QKD systems for secure vehicular networks," in *Proc. IEEE 89th Veh. Technol. Conf. (VTC-Spring)*, Apr. 2019, pp. 1–6.
[35] Z. Chen, T. Joon Lim, and M. Motani, "HAP-assisted LEO satellite downlink transmission: An energy harvesting perspective," in *Proc. IEEE 15th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jun. 2014, pp. 194–198.
[36] K. Hamidouche, W. Saad, and M. Debbah, "Many-to-many matching games for proactive social-caching in wireless small cell networks," in *Proc. 12th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, May 2014, pp. 569–574.
[37] H. Zhang, Y. Xiao, S. Bu, D. Niyato, F. R. Yu, and Z. Han, "Computing resource allocation in three-tier IoT fog networks: A joint optimization approach combining stackelberg game and matching," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1204–1215, Oct. 2017.
[38] Iskandart, S. Gratsia, and M. E. Ernawan, "LTE uplink cellular capacity analysis in a high altitude platforms (HAPS) communication," in *Proc. 11th Int. Conf. Telecommun. Syst. Services Appl. (TSSA)*, Oct. 2017, pp. 1–5.
[39] *Spectrum Needs of High-Altitude Platform Stations Broadband Links Operating in the Fixed Service*, document Rep. ITU-R F.2438-0, F. Series Nov. 2018.
[40] H. Xu and B. Li, "Seen as stable marriages," in *Proc. IEEE INFOCOM*, Apr. 2011, pp. 586–590.
[41] A. E. Roth and J. H. V. Vate, "Random paths to stability in two-sided matching," *Econometrica*, vol. 58, no. 6, pp. 1475–1480, Nov. 1990.
[42] L. Cui and W. Jia, "Cyclic stable matching for three-sided networking services," *Comput. Netw.*, vol. 57, no. 1, pp. 351–363, Jan. 2013.
[43] N. Raveendran *et al.*, "Cyclic three-sided matching game inspired wireless network virtualization," *IEEE Trans. Mobile Comput.*, early access, Oct. 15, 2019, doi: 10.1109/TMC.2019.2947522
[44] W. Xu, X. Li, Y. Wang, C.-H. Lee, and Z. Feng, "Joint dynamic spectrum access and multi-relay selection: A matching-theory-based approach," in *Proc. IEEE 85th Veh. Technol. Conf. (VTC Spring)*, Jun. 2017, pp. 1–7.
[45] F. Kojima and M. U. Ünver, "Random paths to pairwise stability in many-to-many matching problems: A study on market equilibration," *Int. J. Game Theory*, vol. 36, nos. 3–4, pp. 473–488, Mar. 2008.
[46] M. Sotomayor, "Three remarks on the many-to-many stable matching problem," *Math. Social Sci.*, vol. 38, no. 1, pp. 55–70, Jul. 1999.
[47] D. Grace and M. Mohorcic, *Broadband Communications via High Altitude Platforms*. Hoboken, NJ, USA: Wiley, 2010.

Ziye Jia received the B.E. degree in communications engineering and the M.S. degree in electronics and communications engineering from Xidian University, Xi'an, China, in 2012 and 2015, respectively, where she is currently pursuing the Ph.D. degree in communications and information systems. She was a Visiting Ph.D. Student with the Department of Electrical and Computer Engineering, University of Houston, from 2018 to 2020. Her current research interests include resource allocation and network function virtualization in space information networks.

Min Sheng (Senior Member, IEEE) joined Xidian University, in 2000, where she is currently a Full Professor and the Director of the State Key Laboratory of Integrated Services Networks. She has published over 200 refereed papers in international leading journals and key conferences in the area of wireless communications and networking. Her current research interests include space-terrestrial integration networks, intelligent wireless networks, and mobile ad hoc networks. She received the China National Funds for Distinguished Young Scientists in 2018. She is the Vice Chair of the IEEE Xi'an Section. She is an Editor of the IEEE COMMUNICATIONS LETTERS and the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.

Jiandong Li (Senior Member, IEEE) received the B.E., M.S., and Ph.D. degrees in communications engineering from Xidian University, Xi'an, China, in 1982, 1985, and 1991, respectively. He has been a Faculty Member with the School of Telecommunications Engineering, Xidian University, since 1985, where he is currently a Professor and the Vice Director of the academic committee of the State Key Laboratory of Integrated Service Networks. He was a Visiting Professor with the Department of Electrical and Computer Engineering, Cornell University, from 2002 to 2003. His major research interests include wireless mobile communication, and cognitive and self-organizing networks. He was awarded as the Distinguished Young Researcher from NSFC and the Changjiang Scholar from the Ministry of Education, China. He has served as the General Vice Chair of the ChinaCom 2009 and the TPC Chair of IEEE ICCC 2013.

Di Zhou (Member, IEEE) received the B.E. and Ph.D. degrees in communication and information systems from Xidian University, Xi'an, China, in 2013 and 2019, respectively. She was a Visiting Ph.D. Student with the Department of Electrical and Computer Engineering, University of Houston, from 2017 to 2018. Since 2019, she has been with the Broadband Wireless Communications Laboratory, School of Telecommunications Engineering, Xidian University, where she currently holds a faculty postdoctoral position. Her research interests include routing, resource allocation, and mission planning in space information networks.

Zhu Han (Fellow, IEEE) received the B.S. degree in electronic engineering from Tsinghua University in 1997 and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, MD, USA, in 1999 and 2003, respectively. From 2000 to 2002, he was a Research and Development Engineer with JDSU, Germantown, MD. From 2003 to 2006, he was a Research Associate with the University of Maryland. From 2006 to 2008, he was an Assistant Professor with Boise State University, ID. He is currently the John and Rebecca Moores Professor of the Electrical and Computer Engineering Department and the Computer Science Department, University of Houston, TX. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. He has been an AAAS fellow since 2019 and an ACM Distinguished Member since 2019. He received the NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the EURASIP Best Paper Award for the *Journal on Advances in Signal Processing* in 2015, the IEEE Leonard G. Abraham Prize in the field of communications systems (Best Paper Award in IEEE JSAC) in 2016, and several best paper awards in IEEE conferences. He is also the winner of the 2021 IEEE Kiyo Tomiyasu Award, for outstanding early to mid-career contributions to technologies holding the promise of innovative applications, with the following citation: for contributions to game theory and distributed management of autonomous communication networks. He was an IEEE Communications Society Distinguished Lecturer from 2015 to 2018. He is 1% highly cited researcher since 2017 according to Web of Science.