Efficient User Equipment Distribution in DUDe 5G Network using Linear Assignment Algorithms

Konstantinos Tsachrelias
Computer Engineering and Informatics Department
University of Patras
Patras, Greece
Email: up1096511@upatras.gr

Chrysostomos-Athanasios Katsigiannis
Computer Engineering and Informatics Department
University of Patras
Patras, Greece
Email: up1072490@upnet.gr

Vasileios Kokkinos
Computer Engineering and Informatics Department
University of Patras
Patras, Greece
Email: kokkinos@cti.gr

Apostolos Gkamas
Department of Chemistry
University of Ioannina
Ioannina, Greece
Email: gkamas@uoi.gr

Christos Bouras
Computer Engineering and Informatics Department
University of Patras
Patras, Greece
Email: bouras@upatras.gr

Philippos Pouyioutas
Computer Science Department
University of Nicosia
Nicosia, Cyprus
Email: pouyioutas.p@unic.ac.cy

***Abstract*—In the context of 5G networks, efficient resource allocation plays a crucial role in meeting the diverse and demanding requirements of modern applications. This paper presents a novel approach for optimizing User Equipment (UE) distribution in Downlink/Uplink Decoupling (DUDe) 5G networks through the application of Linear Assignment Algorithms (LAAs). The proposed multi-objective framework intelligently allocates UEs to DUDe Base Stations, aiming to maximize network performance, achieve load balancing, enhance Signal-to-Noise Ratio (SNR), and optimize energy efficiency. The findings hold significant implications for network operators, facilitating the advancement of 5G infrastructures to meet the ever-increasing demand for seamless and high-quality connectivity.**

Keywords—Downlink/Uplink Decoupling (DUDe), 5G Networks, Resource Allocation, Heterogeneous Networks* (HetNets), *Hungarian algorithm, Minimum cost flow algorithm

I. Introduction

The advent of Fifth Generation (5G) technology has ushered in a new era of ultra-fast, high-capacity communication networks that cater to the unprecedented demands of modern data-driven applications and devices. As the proliferation of mobile devices and data-hungry services continues unabated, optimizing User Equipment (UE) allocation in 5G networks has become a critical challenge to ensure efficient resource utilization and network performance. Downlink/Uplink Decoupling (DUDe) has emerged as a promising approach to address this challenge, enabling separate allocation of resources for downlink (Base Station (BS) to UE) and uplink (UE to BS) transmissions. To achieve effective UE distribution and subsequently enhance network performance, this paper focuses on comparing two powerful linear assignment algorithms: the Hungarian Algorithm and the Minimum Cost Flow Algorithm, in the context of DUDe technology.

The primary objective of this research is to investigate and identify the most efficient algorithm for UE allocation in DUDe 5G networks. The Hungarian Algorithm [1], a widely used optimization technique, has demonstrated considerable success in various allocation problems, making it a relevant benchmark for comparison. Meanwhile, the Minimum Cost Flow Algorithm [2], known for its capacity to solve network flow problems optimally, presents an appealing alternative that merits exploration in the context of UE allocation in 5G networks.

To conduct a comprehensive comparison, the paper leverages extensive simulations and performance evaluations based on realistic 5G network scenarios. The proposed evaluation framework meticulously examines essential performance metrics, with a particular focus on the Signal-to-Noise Ratio (SNR) improvement. SNR is a crucial indicator of signal quality, directly impacting the reliability and data transmission rates in a 5G network. By analysing and comparing SNR improvements achieved by the two algorithms, the study seeks to determine which approach achieves better UE allocation and subsequently delivers superior network performance in 5G environments. The implications of this research are significant to network operators, 5G infrastructure developers, and researchers seeking to enhance the quality of service in DUDe 5G networks. Ultimately, this investigation contributes valuable insights into the optimization of UE allocation strategies, bringing us one step closer to unlocking the full potential of 5G technology in the face of ever-increasing data demand [3], [4], [5], [6]. There is a significant amount of scientific research being conducted in the field of optimizing 5G networks via DUDe technology, though, it is worth mentioning that to the best of our knowledge this is the first study that compare these two algorithms (Hungarian, Minimum Cost Flow Algorithm) in resource allocation scenarios in DUDe 5G Heterogeneous Networks (HetNets). The authors in paper [7], focus on leveraging DUDe technology to enhance the allocation of network resources based on UE distribution. The research demonstrates that incorporating the capacity constraints of each type of BS results in a more balanced distribution of UEs across the network. In paper [8], the focus lies on addressing the limitations of the conventional Downlink-Uplink (DL-UL) coupled cell association scheme in HetNet. The existing approach often results in suboptimal UE association, where a majority of UEs tend to connect to high-powered Macro BSs (MBS) rather than utilizing the low-powered Small BSs (SBS). The authors in paper [9] tackle the issue of interference in device-to-device (D2D) communication within cellular networks, which is crucial for the advancement of 5G technology. Their proposed algorithm focuses on optimizing resource allocation to minimize interference while ensuring the network's target sum rate is met. Furthermore, in paper [10] review the advancements in solving maximum flow and minimum-cost flow problems in network flows, which are critical combinatorial optimization problems with many practical applications. The survey focuses on the progress made in developing exact algorithms, particularly highlighting their worst-case running times.

The rest of the paper is organized as follows. Section II presents the DUDe technology and its key features. Section III presents the analysis of the algorithms that we will compare in our scenarios. Section IV presents the mathematical model which we used in our simulation environment. Section V presents the results of the simulation and provides a detailed analysis of the findings. Finally, Section VI concludes the paper and provides suggestions for future research.

II. DUDE REVIEW AND FEATURES

DUDe technology is founded on the principle of separately allocating resources for DL (BS to UE) and UL (UE to BS) transmissions. This decoupling of downlink and uplink resources enables network operators to tailor resource allocation strategies more effectively, considering the distinct traffic patterns and requirements of each direction. By uncoupling downlink and uplink allocations, DUDe technology can mitigate interference, improve resource utilization, and enhance the overall quality of service experienced by 5G UEs. Central to DUDe technology's effectiveness is the optimization of UE allocation within the 5G network. Efficient UE allocation plays a critical role in maintaining load balancing across BSs, maximizing network throughput, and minimizing latency. Also to better understand the difference between traditional Downlink/Uplink Coupling (DUCo) and DUDe, in DUCo the assignment of UEs to the BSs is based on the DL SNR of the UE (and both DL-UL are connected to the same BS), while in the case of DUDe the assignment of UEs UL and DL to the BS is based both on the UL SNR and DL SNR of the UE (and UL and DL can be connected in difference BSs). DUDe technology offers several distinct advantages that contribute to its appeal in 5G network optimization. By allocating downlink and uplink resources independently, DUDe efficiently manages varying UE demands and traffic patterns, resulting in reduced congestion and improved spectral efficiency. Furthermore, the decoupling approach enhances network flexibility and adaptability, allowing for dynamic resource allocation and seamless support for diverse services. The review also emphasizes the potential for DUDe technology to support massive Machine-Type Communications (mMTC) and Internet of Things (IoT) devices, as it efficiently caters to the diverse connectivity needs of different UE types [11], [12], [13].

III. ALGORITHM ANALYSIS

This section provides a comprehensive analysis of Algorithm 1 and Algorithm 2 that we employ in the DUDe 5G network. Our goal is to implement these algorithms to observe and compare their impact on UE distribution and overall network performance. By examining their effects, we aim to determine which algorithm leads to a more efficient and optimized network performance. Firstly, we have the description of the original Munkres' Assignment Algorithm, also known as the Hungarian Algorithm, which was published in 1957 when access to computers was limited. As a result, a modified version of the algorithm was developed to allow manual computation using a two-dimensional matrix. This manual approach involved steps like staring and priming zeros, as well as covering and uncovering rows and columns. The Hungarian Algorithm's main goal is to efficiently solve the assignment problem, which involves optimally assigning agents to tasks in a bipartite graph while minimizing costs or maximizing profits. By representing the graph as a matrix, the algorithm can be executed manually by following a series of steps:

Algorithm 1 Analysis of Hungarian Algorithm.

Step 0: Create a nxm matrix called the cost matrix in which each element represents the cost of assigning one of n workers to one of m jobs. Rotate the matrix so that there are at least as many columns as rows and let k=min(n, m).

Step 1: For each row of the matrix, find the smallest element and subtract it from every element in its row. Go to Step 2.

Step 2: Find a zero (Z) in the resulting matrix. If there is no starred zero in its row or column, star Z. Repeat for each element in the matrix. Go to Step 3.

Step 3: Cover each column containing a starred zero. If K columns are covered, the starred zeros describe a complete set of unique assignments. In this case, Go to DONE, otherwise, Go to Step 4.

Step 4: Find a noncovered zero and prime it. If there is no starred zero in the row containing this primed zero, Go to Step 5. Otherwise, cover this row and uncover the column containing the starred zero. Continue in this manner until there are no uncovered zeros left. Save the smallest uncovered value and Go to Step 6.

Step 5: Construct a series of alternating primed and starred zeros as follows. Let Z0 represent the uncovered primed zero found in Step 4. Let Z1 denote the starred zero in the column of Z0 (if any). Let Z2 denote the primed zero in the row of Z1 (there will always be one). Continue until the series terminates at a primed zero that has no starred zero in its column. Unstart each starred zero of the series, star each primed zero of the series, erase all primes and uncover every line in the matrix. Return to Step 3.

Step 6: Add the value found in Step 4 to every element of each covered row and subtract it from every element of each uncovered column. Return to Step 4 without altering any stars, primes, or covered lines.

DONE: Assignment pairs are indicated by the positions of the starred zeros in the cost matrix. If C(i, j) is a starred zero, then the element associated with row i is assigned to the element associated with column j.

Next Algorithm 2 is used to find the maximum flow with minimum cost in a flow network. It initializes variables and matrices to store information about the network, extracts data from the input 'da', and sets up the network based on the provided edges. The main part of Algorithm 2 begins by initializing flow-related variables. It then enters a loop that runs until all vertices have been processed. Within each iteration of the loop, the algorithm updates the reduced cost matrix based on the current flow. It then finds the shortest path from the source vertex to all other vertices using the Bellman-Ford algorithm. If no augmenting path with a negative reduced cost is found, the loop terminates. Otherwise, the algorithm determines the minimum flow that can be pushed along the shortest path and updates the flow on each edge accordingly.

After updating the flow, the algorithm checks if the flow increase resulted in a non-improvement (no change in minimum cost). If so, it adjusts the minimum flow to terminate the algorithm. The algorithm continues this process until it finds the maximum flow with the minimum cost. Finally, it calculates the minimum cost and outputs the flow matrix, maximum flow, and minimum cost.

Algorithm 2 Analysis of Minimum Cost Flow Algorithm.

Input:
2D array 'da' representing the network edges with columns: (source, target, capacity, cost)
Output:
- Maximum flow 'mf' with minimum cost 'mmf'
- 2D array 'f' representing the flow on each edge

Step1. **Initialize variables and matrices**:
- e = number of edges
- v = number of vertices (max value from source and target vertices)
- d1, d2, cc, bb = arrays to store source, target, capacity, and cost respectively
- c = 2D array to store edge capacities between vertices (initialize with zeros)
- b = 2D array to store edge costs between vertices (initialize with zeros)
- a = 2D array to store adjusted edge costs during the algorithm (initialize with zeros)
- p, s = arrays to store predecessor and successor vertices during shortest path calculations

Step 2. **Extract data from input 'da'**:
- Set v = max(max(d1), max(d2))
- Initialize c, b, p, and s based on the edges in 'da'

Step 3. **Network Simplex Algorithm**:
a. Initialize 'mf' to 0 and 'mf0' to positive infinity.
b. Initialize 'f' as a 2D array with zeros to represent the flow on each edge.
c. While 'v' is greater than 0:
i. Initialize 'a' as a 2D array with all elements set to positive infinity except for the diagonal elements (set to zero).
ii. Find the reduced costs and update 'a' matrix:
- For each vertex 'i':
  - For each vertex 'j':
    - If 'j' is not equal to 'i', set 'a(i, j)' to positive infinity.
    - If the edge between 'i' and 'j' exists and the flow on that edge is not zero:
      - Set 'a(i, j)' to the cost of the edge ('b(i, j)') if 'f(i, j)' is 0 (forward edge).
      - Set 'a(j, i)' to the negative cost of the edge ('-b(i, j)') if 'f(i, j)' is equal to the capacity of the edge (backward edge).
      - Otherwise, set 'a(i, j)' to the capacity of the edge ('b(i, j)') and 'a(j, i)' to the negative capacity ('-b(i, j)').

iii. Initialize arrays 'p' and 's' to store predecessor and successor vertices.
- For each vertex 'i' (except the first vertex):
  - Set 'p(i)' to positive infinity.
  - Set 's(i)' to 'i'.

iv. Find the shortest path from the first vertex to all other vertices using the Bellman-Ford algorithm:
- For each vertex 'k':
  - For each vertex 'i':
    - For each vertex 'j':
      - If 'p(i) > p(j) + a(j, i)', update 'p(i)' and 's(i)'.

v. If the shortest path cost to the last vertex ('v') is equal to or greater than positive infinity, break the loop.
vi. Initialize variables 'dv' to positive infinity and 'm' to 'v'.
vii. Find the minimum flow that can be pushed along the shortest path:
- While 'v' is greater than 0:
  - If the flow on the edge from 's(m)' to 'm' is positive:
    - Set 'dv' to the minimum of ('c(s(m), m) - f(s(m), m)') and 'dv'.
  - If the flow on the edge from 's(m)' to 'm' is negative:
    - Set 'dv' to the minimum of ('f(m, s(m))') and 'dv'.
  - Set 'm' to 's(m)'.

viii.Update flow 'f' along the shortest path and adjust the residual capacities accordingly:
- While 'v' is greater than 0:
  - If the flow on the edge from 's(m)' to 'm' is positive:
    - Increment the flow on the edge from 's(m)' to 'm' by 'dv'.
  - If the flow on the edge from 's(m)' to 'm' is negative:
    - Decrement the flow on the edge from 'm' to 's(m)' by 'dv'.
  - Set 'm' to 's(m)'.

ix. If the current flow ('mf') plus the minimum flow ('dv') is greater than or equal to 'mf0':
- Set 'dv' to the difference between 'mf0' and 'mf'.
- Set 'd' to 1.

x. Update the total flow ('mf') by adding 'dv'.
xi. If 'd' is 1, break the loop.
d. End of the while loop.

Step 4. Calculate the minimum cost 'mmf':
- Set 'mmf' to 0.
- For each vertex 'i':
  - For each vertex 'j':
    - Increment 'mmf' by the product of flow on the edge ('f(i, j)') and the cost of the edge ('b(i, j)').

Step 5. **Output the results**:
- Print the flow matrix 'f'.
- Print the maximum flow with minimum cost ('mf').
- Print the minimum cost ('mmf').

The Hungarian Algorithm stands as a valuable tool renowned for its effectiveness in solving assignment problems, particularly when dealing with scenarios where the objective is to optimally match elements from two sets while minimizing the total cost. However, when confronted with more intricate and multifaceted challenges, especially those encompassing network optimization, multiple constraints, and various cost considerations, the Minimum Cost Flow Algorithm emerges as the preferred choice. The broader spectrum of capabilities of the Minimum Cost Flow Algorithm makes it an indispensable resource in the arsenal of optimization techniques. This algorithm not only excels at addressing traditional assignment problems but also highlights its versatility in handling complex and multifaceted scenarios. Its efficiency in navigating intricate networks and accommodating various constraints positions it as the go-to option for tasks demanding a more comprehensive approach to optimization. Consequently, when tackling the intricacies of challenging optimization tasks, the Minimum Cost Flow Algorithm often proves superior due to its adaptability and proficiency in managing the numerous aspects of complex problems.

IV. MATHEMATICAL MODEL

This section gives a detailed explanation of the mathematical model used to set up and carry out the simulations in subsequent scenarios. The determination of the minimum distance between UEs and BS relies on a mathematical model defined in TR 38.901 Section 7.4.1 [14]. However, the detailed analysis of this model is beyond the scope of this paper. The model evaluates the Path Loss (PL) in various scenarios, considering both Line-Of-Sight (LOS) and Non-Line-Of-Sight (NLOS) conditions.

$$PL_{\mathrm{RMa-LOS}} = \begin{cases} PL_1 & 10m \leq d_{\mathrm{2D}} \leq d_{\mathrm{BP}} \\ PL_2 & d_{\mathrm{BP}} \leq d_{\mathrm{2D}} \leq 10\mathrm{km} \end{cases} \tag{1}$$

$$\begin{gathered} PL_1 = 20\,log_{10}(\,40\pi d_{\mathrm{3D}} f_c 3) + \\ min(\,0.03h^{1.72}, 10)\,log_{10}(\,d_{\mathrm{3D}}) - \\ min(\,0.044h^{1.72}, 14.77) + 0.002\,log_{10}(\,h) d_{\mathrm{3D}} \end{gathered} \tag{2}$$

$$PL_2 = PL_1(d_{\mathrm{BP}}) + 40\, log_{10}(\, d_{\mathrm{3D}}/d_{\mathrm{BP}}) \quad (3)$$

$$\mathrm{SNR} = \mathrm{Psignal/Pnoise} \quad (4)$$

The path loss is computed using equations (1), (2), and (3). Equation (1) accounts for the path loss in both LOS and NLOS conditions, considering the distance between the UE and the BS antennas. Equation (2) factors in the three-dimensional distance, carrier frequency, UE height, and other relevant parameters for path loss calculation. Equation (3) further adjusts the path loss, considering the breakpoint distance and the three-dimensional distance. Once the minimum distance is determined for each UE from different types of antennas, the next step involves computing the SNR to identify the antenna type that offers the optimal connection. Following the determination of SNR values using Equation (4), the Matlab "MatchPairs" function and other relevant functions are utilized to optimize the distribution of these SNR values efficiently across the network. By employing these functions, the SNR values can be adjusted and organized in tables to better suit the algorithms' requirements. The objective here is to find an optimal distribution of UEs based on their SNR values, which will enable the algorithms to make well-informed decisions in allocating resources and managing connections within the network,

V. SIMULATIONS PARAMETERS AND RESULTS

In this section, three comparative scenarios are explored. The first scenario provides a comparison between DUCo and DUDe. The second scenario compares the resource allocation methods of basic DUDe and DUDe employing the Hungarian algorithm, while the third scenario compares the Hungarian algorithm to the Minimum Cost Flow algorithm for resource allocation within DUDe. These scenarios are evaluated for cases involving 100, 500, and 1000 UEs. It is also noteworthy that the Hungarian algorithm is implemented using Matlab's "MatchPairs" function. Firstly, we established a 5G heterogeneous network for our simulations within a 2 x 2 km urban square area, as we can see in Fig.1. This network comprises a combination of different antenna types, including 2 Macro Cell antennas, 4 Micro Cell antennas, and 8 Pico Cell antennas. These antennas are strategically positioned, with the Macro Cell antennas placed at a height of 30 meters and the Micro Cell antennas at 10 meters, while the Pico Cell antennas are positioned at a height of 5 meters. It's important to note that the Macro Cell antennas are in the centre of the area, and the remaining 12 antennas are strategically distributed around them. Each type of antenna in our network operates with varying transmission power levels. The Macro Cell antennas are the most potent, transmitting at a power level of 45 dBm, followed by the Micro Cell antennas at 33 dBm, and the Pico Cell antennas with the lowest power output at 24 dBm. Additionally, these antennas exhibit different gains, with the Macro Cell antennas having a gain of 21 dBi, the Micro Cell antennas with a gain of 10 dBi, and the Pico Cell antennas featuring a gain of 5 dBi. The summarized details of the aforementioned parameters can be found in Table I. Before analysing the simulations, to prevent any misconceptions, it is important to note that the algorithms (Hungarian, Minimum Cost Flow) produce identical distributions. The variations lie in the complexity and speed of their respective implementations, details of which will be examined further below. Fig. 2 to Fig. 4, show that when DUDe technology is applied, it results in a significantly more even distribution compared to the previous DUCo technology where UEs are mainly connected with the first two BSs. However, as we can see from Fig. 5. to Fig. 7., DUDe technology still falls short of reaching the level of distribution efficiency achieved by the Hungarian approach. This difference is particularly important because the implementation of the Hungarian algorithm takes an integrated approach, considering not only the UE proximity to the antennas but also their spatial distribution, with the primary goal of improving the overall network performance. To prevent any potential confusion with similar studies, it is important to clarify that we adopted a similar model structure and parameters for our simulation but applied them to distinct scenarios resulting in different conclusions. This contribution offers valuable insights to the broader scientific community.

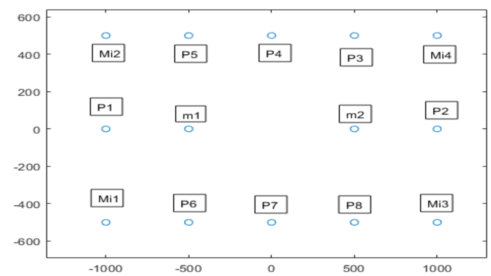

Fig. 1. Topology of our network. (M) for Macro (Mi) for Micro and (P) for Pico.

TABLE I. SIMULATION PARAMETERS

Parameter	Value
Transmit power(dBm)	UE=20, Macro cell = 45,Micro cell = 33 Pico cell = 24
BS height (m)	Macro height = 30, Micro height =10, Pico height = 5
Antenna gain (dBi)	Macro cell = 21, Micro cell = 10, Pico cell = 5
Bandwidth (MHz)	20
Environmental parameters	UE1=100/UE2=500/UE3=1000, Position=random
Power Noise	Pnoise= -74+10log(Bandwidth(hz))

The bar charts, which visually represent these results, underscore the critical importance of addressing the linear assignment problem before implementing DUDe technology within a 5G network infrastructure. It is essential to recognize that the outcomes portrayed in the subsequent figures are the culmination of an extensive analysis based on data collected from 500 snapshots, providing a comprehensive perspective on the performance of these network distribution methods. Furthermore, the figures illustrate the mean UE distribution following the implementation of 500 snapshots. For instance, in the case of 100 UEs using the DUCo application, on average, 44 UEs are connected to Macro 1 and 2. Conversely, with the DUDe application, approximately 36 UEs connect to these same antennas.

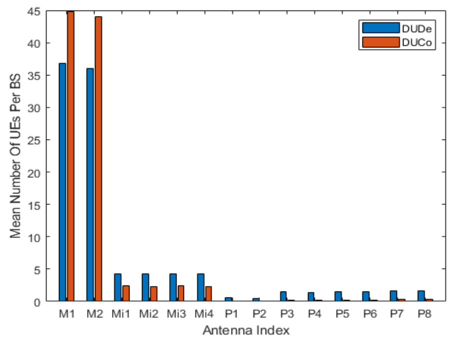

Fig. 2. Mean Comparison Distribution between DUCo/DUDe for 100UEs.

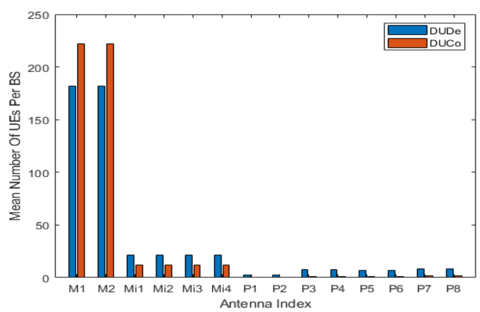

Fig. 3. Mean Comparison Distribution between DUCo/DUDe for 500UEs

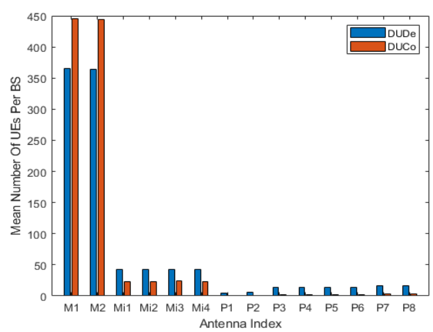

Fig. 4. Mean Comparison Distribution between DUCo/DUDe for 1000UEs.

To enhance the reader's comprehension of the bar charts in our study, it is essential to clarify the notation used. Specifically, in the initial two slots along the X-axis, we have Macro-1 followed by Macro-2. Moving along, the subsequent four positions are dedicated to Micro, denoted as Micro-1 through Micro-4. Lastly, the remaining eight antennas are arranged in a sequential fashion, representing Pico-1 through Pico-8. After conducting our simulations, we observed a significant difference in UE distribution between the DUDe approach without the application of Hungarian and DUDe with Hungarian. In the 1000 UEs case, the mean distribution of UEs on Macro BSs exceeded 350, while in the latter case, the UE count did not surpass 100. This observation directly translates into improved network performance and more efficient UE service. By avoiding the overload of any single BS with a large number of UEs, the network becomes better equipped to handle incoming UEs seamlessly. It is worth noting that some UEs do not connect to the BS nearest to them. For instance, in the case of UE 8, despite having the highest SNR of 36 dB, the DUDe algorithm without Hungarian directed them to Macro-1 BS, whereas with the application of Hungarian, they connected to Pico-4 BS with an SNR value of 9 dB. While this may not result in optimal communication quality for this specific UE, the overall network benefits from improved efficiency, zero delays, and reduced interference. This phenomenon also underscores the effectiveness of the Hungarian algorithm in solving the linear assignment problem. UEs do not always connect to the BS with the "best" signal quality, defined as the closest distance. Instead, the algorithm optimizes the network's performance, leading to a more efficient and interference-free operation.

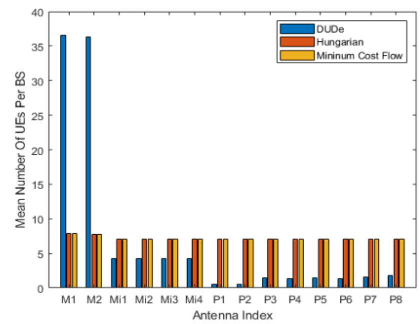

Fig. 5. Mean Comparison Distribution between DUDe/DUDe with Hungarian/ Minimum Cost Flow for 100UEs

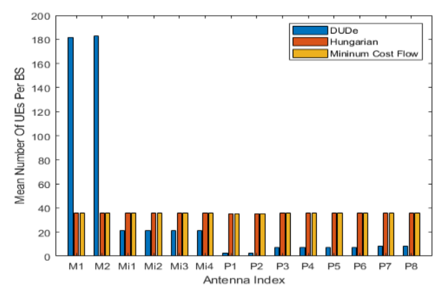

Fig. 6. Mean Comparison Distribution between DUDe/DUDe with Hungarian/ Minimum Cost Flow for 500UEs

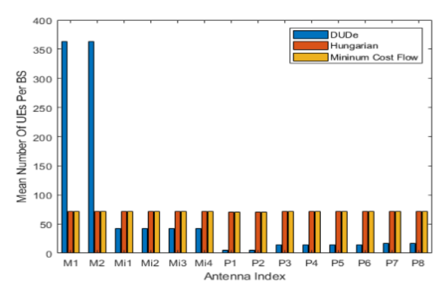

Fig. 7. Mean Comparison Distribution between DUDe/DUDe with Hungarian/ Minimum Cost Flow for 1000UEs

Following the demonstrated significance of addressing the Linear Assignment Problem by combining it with DUDe technology for optimal UE allocation in our network, our next phase involves exploring this problem using the Minimum Cost Flow Algorithm. This endeavour aims to discern the distinctions between this approach and the algorithm we previously employed via the Hungarian function. In our simulations, we found that the Minimum Cost Flow algorithm performs better than the Hungarian algorithm in terms of both execution time and computational requirements. As the number of UEs increases, the Hungarian algorithm's time complexity grows cubically, leading to longer execution times. In contrast, the Minimum Cost Flow algorithm scales more efficiently as the number of UEs and BSs (denoted as U and B, respectively) increases, resulting in reduced computational workload and faster UE allocation. The overall time complexity of the Minimum Cost Flow algorithm is primarily determined by finding the minimum cost flow in the directed graph, which can be expressed as O (U * (U + B + log(U + B))). This contrasts with the Hungarian algorithm, which has a time complexity of O(N^4), where N represents the maximum of U and B. In summary, our comparative analysis strongly supports the use of the Minimum Cost Flow algorithm for more efficient UE allocation when compared to the Hungarian algorithm. This adoption contributes to the enhancement of wireless communication systems and optimizes the utilization of MM technology in 5G networks. Table II presents the comparison between these two algorithms.

TABLE II. COMPARISON BETWEEEN ALGORITHMS

Algorithms	**Complexity types**	**Time Of Implematation**
Hungarian	O(N^4)	10 min for 100 UEs 12 min for 500 UEs 15 min for 1000 UEs
Minimum Cost Flow	O(U * (U + B + log(U + B)))	10 Seconds for all UEs scenarios
Difference		98%, 98.5%,99%

Furthermore, it is important to note that in our simulations, implementing the Hungarian algorithm through Hungarian required a substantial amount of time, typically ranging from 10 to 15 minutes. In contrast, the Minimum Cost Flow function completed the same tasks in just a matter of seconds. To assess the execution times for Hungarian, we utilized the 'timeit' command in Matlab, conducting measurements for each scenario individually.

VI. CONCLUSION AND FUTURE WORK

Our simulations have demonstrated that both the Hungarian and Minimum Cost Flow algorithms perform comparably in terms of the resulting UE distribution, achieving similar quality of service for end-UEs. However, the key differentiator between the two algorithms lies in their computational efficiency and complexity. The Minimum Cost Flow algorithm outperforms the Hungarian algorithm in terms of speed and computational resources required for implementation. Future work in this area could explore numerous subjects. For example, further optimization techniques and heuristics for both the Hungarian and Minimum Cost Flow algorithms could be investigated, to potentially enhance their performance in specific network scenarios or under real world conditions, in order to uncover any potential challenges and obstacles during their implementation. Additionally, we will address scalability issues and performance under different network conditions, particularly focusing on varying numbers of UEs and BSs. Moreover, we will consider real-world constraints such as dynamic network conditions and user mobility to ensure our approach remains effective and adaptable in practical applications. Moreover, the integration of machine learning techniques could be investigated, in order to predict UE distribution patterns and assist in the decision-making process for allocating resources more intelligently. Lastly, field trials and simulations in actual 5G network deployments, can be conducted, to validate the findings and assess the practical implications of implementing the Minimum Cost Flow algorithm over the Hungarian algorithm.

ACKNOWLEDGMENT

The research project was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the “2nd Call for H.F.R.I. Research Projects to support Faculty Members & Researchers” (Project Number: 02440).

REFERENCES

[1] C. Huang, A. M. Graphcore, J. D. Graphcore, D. Mottin and I. Assent, “HUNIPU: Efficient Hungarian Algorithm on IPUs”, 2024 IEEE 40th International Conference on Data Engineering Workshops (ICDEW), Utrecht, Netherlands, 2024, pp. 388-394

[2] L. Chen, R. Kyng, Y. P. Liu, R. Peng, M. P. Gutenberg and S. Sachdeva, “Maximum Flow and Minimum-Cost Flow in Almost-Linear Time”, 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS), Denver, CO, USA, 2022, pp. 612-623

[3] M.Hasan, E.Hossain, “Distributed resource allocation in 5G cellular networks”,In Towards 5G Applications, Requirements and Candidate Technologies, pp. 245–269 (2015).

[4] H.Elshaer, F.Boccardi, M.Dohler, and R.Irmer “Downlink and Uplink Decoupling: A disruptive architectural design for 5G networks”, 2014 IEEE Global Communications Conference, Austin, TX, USA, 2014, pp. 1798-1803

[5] Y.Cao,“Hungarian Algorithm for Linear Assignment Problems” (V2.3),“https://www.mathworks.com/matlabcentral/fileexchange/20652-hungarian-algorithm-for-linear-assignment-problems-v2-3”,MATLAB Central File Exchange. Retrieved July 28, 2023.

[6] W.Zhang,“Finding minimum cost flow in the network A Matlab program and application” Selforganizology 4.2 (2017): 30-34.

[7] K. Tsachrelias, A. Gkamas, C. -A. Katsigiannis, C. Bouras, V. Kokkinos and P. Pouyioutas, "On the Optimization of UE Allocation in Heterogeneous 5G Networks Using DUDe Techniques," 2023 Fourteenth International Conference on Ubiquitous and Future Networks (ICUFN), Paris, France, 2023, pp. 521-526

[8] H. Z. Khan et al., “Resource Allocation and Throughput Maximization in Decoupled 5G”, 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Korea (South), 2020, pp.1-6

[9] F. Nizam et al., “Resource Allocation Algorithm for D2D Communication in Cellular Networks Based on Hungarian Algorithm”, 2019 IEEE 14th Malaysia International Conference on Communication (MICC), Selangor, Malaysia, 2019, pp. 106-111

[10] C. Oliverio, and A.Letchford, “A survey on exact algorithms for the maximum flow and minimum‐cost flow problems”, Networks 82.2 (2023): 167-176.

[11] A. J. Muhammed, Z. Ma, Z. Zhang, P. Fan and E. G. Larsson, “Energy-Efficient Resource Allocation for NOMA Based Small Cell Networks With Wireless Backhauls”, in IEEE Transactions on Communications, vol.68,no.6, pp. 3766-3781, June 2020,

[12] Y.Shi, E. Alsusa, and M. W. Baidas. “A survey on downlink–uplink decoupled access: Advances, challenges, and open problems.” In ScienceDirect, Computer Networks, Vol. 213 (2022): 109040.

[13] HZ Khan, Humayun Zubair, et al. “Resource allocation and throughput maximization in decoupled 5G. “ 2020 ieee wireless communications and networking conference (wcnc). IEEE, 2020.

[14] 3GPP, TR 38.901 Section 7.4.1"Study on channel model for frequencies from 0.5 to 100 GHz." (Release 17), March 2022