Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.
Digital Object Identifier 10.1109/ACCESS.2017.DOI

A Traffic Type-based Differentiated Reputation Algorithm Radio Resource Allocation for Multi-service Content Delivery in 5G Heterogeneous Scenario

CRISTINA DESOGUS[1], (Student Member, IEEE), MATTEO ANEDDA[1], (MEMBER, IEEE), MAURIZIO MURRONI[1],(SENIOR MEMBER, IEEE), GABRIEL-MIRO MUNTEAN[2],(Senior Member, IEEE)
[1] Dept. Electrical and Electronic Engineering - DIEE/UdR CNIT, University of Cagliari, Piazza d'Armi, 09123, Cagliari, Italy
cristina.desogus@diee.unica.it, matteo.anedda@diee.unica.it, murroni@diee.unica.it
[2]Performance Engineering Laboratory, School of Electronic Engineering, Dublin City University Dublin, Ireland
gabriel.muntean@dcu.ie
Corresponding author: Cristina Desogus (e-mail: author@ cristina.desogus@diee.unica.it).

ABSTRACT The next generation network environment is expected to include networks of diverse types, associated with heterogeneous performance. At the same time, this heterogeneous network environment will be used to deliver various services with different requirements. In order to support high quality of experience for the users availing from these services, there is a need for a solution to the complex problem of selecting the appropriate network support for each user service type. This paper introduces the network Traffic tYpe-based DifferEntiated Reputation (TYDER) solution, which differentiates the data delivery process according to its type. TYDER considers network reputation in the context of traffic type requirements in order to increase the delivery performance for the data exchanged. Comparative testing involving four traffic categories which include video, gaming, browsing and Internet of Things(IoT) showed how TYDER outperforms a classic solution in terms of major performance metrics.

INDEX TERMS 5G, Device to Device, Internet of Things, Multimedia Delivery Content, Performance evaluation, Quality of Service, Radio Access Network, Traffic and Performance monitoring.

I. INTRODUCTION

RADIO resource allocation and scheduling algorithms are proposed in order to increase users' satisfaction levels when served with diverse services via heterogeneous wireless networks. These algorithms are employed in several areas of concern, including cognitive radio networks, satellite-terrestrial coexistence-communications based on radio maps, and radio sensor networks [1]. In particular the fast growth of Machine-to-Machine (M2M) communications introduces additional challenges when satisfying diverse Quality of Service (QoS) requirements of massive number of Machine Type Communications (MTC) with limited radio resources [2]. In this context, Device-to-Device (D2D) communications refer to technologies that enable devices to communicate directly with each other, avoiding data-path routing through a network infrastructure. D2D communications technologies are the most common solutions employed for M2M communications in 5G network environments [3]. The devices considered in D2D scenarios are pieces of User Equipment (UE) able to transfer data using the IP protocol and perform networking according to 3GPP specifications.

Current scenarios have a great variety, both regarding to the type of networks and services users require. As far as the networks are concerned, in addition to today's classic new generation broadband wireless networks, such as 802.11ac/ad/af [4] and cellular networks such as LTE-A [5], we are beginning to see the emergence of new 5G standards and networks such as the NarrowBand IoT (NB - IoT) [6].

This network environment is supposed to address the new demands of mobile users, which are highly diverse and increasingly rich. The trend that is expected in the coming years includes a growth in demand for services that once attracted

FIGURE 1: Example of Heterogeneous Wireless Networks

small groups of users. In particular, there will be a remarkable growth in gaming services and high resolution video requests will reach very high demand levels. These services will be accompanied by mobile IoT traffic, such as that generated for example by mobile health applications. This is as the number of mobile phones are expected to be surpassed by that of IoT devices at end of 2018 [7].

The presence of so many types of available services and a range of heterogeneous and constantly expanding networks, leads to the need to understand what is the best network for any particular service. In the literature, the concept of network reputation has come a long way. Reputation allows us to determine which network is able to offer a better QoS. The limit of reputation, as currently employed, is that for each type of service, the same evaluation parameter is used. This approach does not take into account that different services have different needs and issues and may require different types of protocols and different solutions to meet their requirements. Differentiated traffic delivery [8] and network reputation [9] have already been considered independently in the literature, but differentiated traffic-based reputation in the network selection process has never been considered.

In this work we will consider four different types of traffic: *video*, *games*, *document access and navigation* and *IoT*. Each type of traffic has associated a different type of communication protocol mostly used for its delivery and is affected particularly by certain network parameters. These will be specified in more details in the paragraph dedicated to network protocols.

In a scenario, such as the one described, characterized by heterogeneous networks and diverse rich services, the problem of choosing the best network that meets the user's QoS needs becomes extremely important and is increasingly more complex than in the past.

This paper introduces a network Traffic tYpe-based DifferEntiated Reputation Algorithm (TYDER), which differentiates the treatment of data delivery in heterogeneous multi-network environments according to its type. This work extends the research on reputation-based network selection by diversifying the reputation concept to consider different types of services. The proposed TYDER solution evaluates reputation based on feedback received automatically during data delivery. The feedback is assessed using different score functions, appropriate for each type of service considered. In order to perform evaluation, QoS parameters such as delay, packet loss ratio, and throughput are used. TYDER has been compared with a classic Multiplicative Exponential Weighting (MEW) approach [10]. MEW combines several inputs such as power of the received signal, throughput, packet delay, cost-per-user, the requested type of traffic, and type of device without considering network reputation concept. Furthermore, TYDER has also been compared to E-PoFANS [11] which proposed a reputation-based network selection scheme, but only for video traffic in 3.5G (i.e., UMTS/HSDPA) and WLAN heterogeneous access networks.

The rest of the paper is structured as follows. Section II discusses related works in the areas of network selection, network reputation and D2D communications. The proposed TYDER solution is detailed in section III. Section IV describes the simulation-based testbed and scenarios for performance evaluation, whereas section V presents and discusses the results. The paper is concluded in section VI.

II. RELATED WORKS

In this section the authors survey the state of the art research related to the heterogeneous networks with particular focus on solutions involving access network selection, network reputation, traffic differentiation with emphasis on their main limitations. The main problem in the network selection algorithms is to identify the selection parameters and define the mechanism to combine them.

In [12] the authors have studied and compared systematically the most important mathematical theories used for modeling the network selection problem in the literature. Multiple Attribute Decision Making (MADM) is proposed to enable making a preference-based decision over the available alternatives that are characterized by multiple (usually conflicting) attributes.

In [13], the researchers have introduced a novel MADM based on critical parameters, such as the speed of the mobile device, network load and cost of the service, weighted through a fuzzy logic scheme, to obtain a candidate network suitable for the user. A QoS factor is attributed to each network. This factor is calculated for each network by processing the weighted decisional matrix using the analysis of data rate, delay, jitter and packet loss ratio. The focus of the proposed solution is on getting a candidate network with a low computational burden.

Desogus et al. [10] have proposed a MEW approach to MADM for network selection in a LTE-A/WLAN heterogeneous scenario. MEW combines several inputs such as power of the received signal, throughput, packet delay, cost-per-user, the requested type of traffic, and type of device in order to improve the real-time balance of available radio resources.

In paper [14], the authors have proposed a MADM solution for network selection in ultra dense scenarios, such as the 5G system, with the aim of eliminating unnecessary handovers. To minimize handovers, authors take into account different classes of traffic related to different user requirements. The proposed scheme is designed to minimize the handover only and not to improve QoS.

Trestian et al. [11] have proposed an algorithm for network selection which increases the energy efficiency of content delivery and prolongs the mobile device battery lifetime. This is achieved by selecting the network that offers the best energy-quality trade off. Much importance is given to battery power saving of the device, which becomes the central element of network selection. The selected network is the one that allows for the highest energy savings.

In [15], the authors have proposed and implemented an algorithm for network selection based on a newly defined network reputation metric which emphasizes the mobility of users within the network. The focus of the paper is to keep the level of QoS high while managing the user's mobility. Network reputation is calculated based on device profiles, user reputation reports, and network conditions. It is used in the selection decision of the network in order to allow the user to connect to the most appropriate one.

In [16], the authors have proposed a technique to order preferences by similarity to an ideal solution, called Technique for Order Preference by Similarity to Ideal Solution (TOPSIS), when solving the problem of offering the final user the highest QoS. TOPSIS combines utility function, reputation theory and MADM for selecting the best network alternative. The basic concept behind this method is that the selected alternative must have the shortest distance from the ideal solution and is furthest from the ideal negative solution. The Euclidean distance has been proposed to evaluate the relative proximity of the alternatives to the ideal solution. The reputation of the network is based on utility functions and is used to give greater and lesser advantage to a given utility within the method TOPSIS. In TOPSIS, the network's reputation is a static value that is not updated over time.

In these works, the reputation is attributed to the network without considering the type of traffic that is requested by users. In particular, in most cases video is the only type of traffic focused on. Moreover, none of the above methods have been tested on new generation networks, such as NB-IoT.

In paper [17], the authors try to remedy the problem of latency management in communication between user and cloud. The rapid growth of online games and especially the Massively Multiplayer Online Games (MMOG) leads to the growth of problems related to cloud management. This is because graphical rendering is downloaded to the cloud, so data transmission between end users and the cloud increases significantly response latency and limits user coverage, thus preventing cloud games from achieving high QoS levels. The proposal is to create a Fog-Assisted Cloud Gaming Infrastructure (CloudFog). CloudFog consists of supernodes that are responsible for rendering gaming videos and streaming them to nearby players. Fog allows the cloud to be responsible only for the intense calculation of the state of the game and for sending of supernode update information. This significantly reduces traffic and use of resources, therefore decreasing latency and bandwidth utilization. Each supernode is assigned a reputation, in order to assign to each player the suitable supernode able to provide a satisfying video streaming service. This type of strategy is specific to gaming.

In paper [18], the authors have proposed a solution to tackle the problem of Quality of Computing (QoC), in IoT systems. They introduced a dynamic network selection mechanism based on Software Defined Networks (SDN) designed to provide QoC in urban IoT scenarios in which heterogeneous network resources are shared. The proposed mechanism dynamically assigns portions of data from IoT flows on licensed and unlicensed bands to ensure QoC while minimizing operating costs and occupation of the licensed band. Proposing a solution that works on band portions instead of at the application level makes this method less flexible.

In paper [19], the authors have taken into consideration the problem of energy consumption in applications and devices that deal with transmission and reception of video content. This is because video content requires high-energy consumption. The algorithm is proposed to provide high Quality of Experience (QoE) levels to the user during multimedia delivery while maintaining a balanced trade-off between quality and energy. Among the various factors that are taken into consideration are the average loads and geographical positions. The algorithm, then selects future segments in two steps, in the first step considers the previous throughput and energy consumption of the user device to choose an appropriate quality level, and in the second step identifies the most suitable host peer based on their previously shared load and location. It foresees the storage of the situation that has verified the neighbor, so as to be able to predict with better approximation the consumption of the i_th device. In this case, the algorithm does not have a real reputation of the network but an evaluation of the same based on what has occurred or is occurring close to the user. Moreover, the focus is exclusively on video type traffic.

In paper [20], the authors have proposed a solution for a smart vertical handover framework to simplify network selection and reduce latency and handover frequency. By integrating the Media-Independent Handover (MIH) and Software-defined Network (SDN) technologies, it is possible to ensure that the handover takes place between only two potential networks regardless of the types of available technologies. The purpose is to avoid multiple handovers. The network selection takes place through a selective algorithm, and is based on the maximum possible QoS value. The developed model makes the most of the computational capacity on the network side, which increases the efficiency of calculation and at the same time reduces the energy consumption of the mobile terminal. It also improves the accuracy of the handover decision, as decisions are made at the three-level pre-selection. The amount of information exchanged during the decision-making process of handover and in the global network has considerably decreased. The final effects produce the optimization of the signaling load between the networks and the back-haul requirements. Although based on 5G technology, the proposed solution does not differentiate the various available networks and neither considers various types of traffic.

In [9], the authors have focused on network selection that allows the best connectivity based on the characteristics of the network, considering their variation over time, and based on the user's position within each network. They proposed a network selection solution that can detect the user's location that aims to improve the distribution of content in a heterogeneous wireless network environment by selecting the best network. Based on the network performance information and the mobile user's location and speed, the algorithm selects the best available network to ensure the delivery of high quality data into the heterogeneous wireless network environment. Even in this case, the network does not have a reputation, but the selection is based on distance and user mobility. Moreover, traffic differentiation is not considered, either.

In paper [21], the authors have addressed the problem created by the heterogeneity of mobile devices (e.g., screen resolution, battery life and hardware performance) that create a serious impact on the end user's QoE. They proposed Evolved QoE-aware Energy-saving Device-Oriented Adaptive Scheme (E^3DOAS) for mobile multimedia delivery over future wireless networks. E^3DOAS uses a strategy of allocation of rates based on coalition play within the heterogeneous multi-device environment and optimizes the trade-off between the quality perceived by the end user of multimedia delivery and the energy saving of the mobile device. The focus of the algorithm is the balancing of networks based on the energy saving of the device. In this case, the networks do not have a reputation and a ranking that takes into account the progress of the QoE over time. Finally, the algorithm does not take into account the type of service used by the user.

In paper [22], the authors have introduced an algorithm that balances the LTE small-cells networks. The purpose of the algorithm is to perform balancing between the networks avoiding the collapse of one or more of them which would create a significant worsening of the throughput. To carry out this balancing, they are based on the progress of the overloaded cells and adjacent cells, adapting the state of the network load and considering the load estimate. The use of resources depends on the quality of the signal and the traffic requests of the User (UE) connected in LTE. The proposed solution does not fit heterogeneous networks and can only be used on networks that rely on the resource blocks (which are the basis of LTE). Furthermore, the reputation of the network is not calculated. The main purpose is in fact to have a set of balanced LTE networks.

In paper [23], the authors have looked for a solution to the problems due to Dense heterogeneous Networks (DenseNets), in which mobile users make the choice in terms of the network to connect to, in order to balance energy savings and delivery performance. The proposed solution is a Hybrid Unicast-Multicast utility-based Network Selection algorithm (HUMANS), which offers the additional option of selecting multicast transmissions in the network selection process during video delivery. This allows to outperform other solutions in terms of percentage of interruption and average quality of transmission, both in low and high density scenarios. Neither reputation or traffic differentiation is considered.

In this context, TYDER proposes a network selection solution based on network reputation and traffic differentiation in a 5G heterogeneous networks scenario, including NB-IoT networks. TYDER's goal is to improve QoS with respect to other state-of-the art algorithms. Taking into account the feedback from users, TYDER monitors the various networks QoS levels over time, allowing selection of the best network for user needs at any time.

III. PROPOSED SOLUTION

TYDER associates a reputation to each network available to the user and within each network for eachof the four types of service considered. The reputation is based on feedback from users who use the network and considers in its calculation feedback variation during the day and week, respectively. The feedback the users send to the sever is numeric and is calculated through a multi-criteria method, differentiated according to the service used. This approach puts particular emphasis on the fundamental factors for the QoS of each particular service type. The risk or sensitivity factors of each service are converted into utility functions that are used in conjunction to determine a score associated with the network that is being used at that very moment in time. This score is sent to the server that stores it in its own database of networksand can be processed alongside other such scores. This database is queried every time a user enters the network, moves within it or changes its service.

A. PROTOCOLS

Specifically, the differentiation of service type is performed based on the protocols employed. The protocol list includes (and is not limited to) the following major ones:

- For the Video service type (VI), one of the following protocols is likely to be employed:
  - Dynamic Adaptive Streaming over HTTP (MPEG DASH) [24]; it is based on dynamic adaptive streaming media technology. It allows the customer to choose the bitrate based on download speed, network status, and buffer change [25]. This is very useful because if the network is particularly slow, smaller blocks are required to be transmitted in order to maintain satisfactory QoS levels.
  - HTTP Live Streaming (HLS) [26]; like MPEG DASH, HLS is an adaptive protocol. At the start of the streaming session an extended M3U (M3U8) playlist is downloaded. This contains the metadata for the various sub-streams that are provided. According to network conditions one or another of these sub-streams are played.

- Real Time Streaming Protocol (RTSP) [27] defines control sequences useful in controlling multimedia playback. While HTTP is stateless, RTSP has state; an identifier is used when needed to track concurrent sessions. Like HTTP, RTSP uses TCP to maintain an end-to-end connection and, while most RTSP control messages are sent by the client to the server, some commands travel in the other direction (i.e., from server to client). RTSP is not an adaptive protocol and controls data delivery only. The transmission is performed using other protocols such as Realtime Transport protocol (RTP) [28].

The first two protocols are proprietary protocols, present respectively in Microsoft Windows and Apple products, whereas RTP and RTSP are not. Video content transmissions performed using these protocols are sensitive to jitter, throughput, and delay.

- Gaming service type (GM)uses protocols such as:
  - Open Game Protocol (OGP) [29]. OGP was developed and designed to provide specific real-time information about games running at any given server. Most effort was made to meet all the needs of a flexible game protocol which is supposed to support every kind of game;
  - Transport Control Protocol (TCP) [30] and User Datagram Protocol (UDP) [31]. TCP and UDP are transport layer protocols and perform data delivery using reliable or unreliable solutions, respectively. Depending on the type of online game, one, the other or both transport protocols are preferred.

  This type of service is particularly sensitive to delay, packet loss rate, jitter, and throughput.
- The IoT service type (IoT) uses a number of new generation protocols such as:
  - Message Queuing Telemetry Transport (MQTT) [32], it has been designed as an extremely light publication/subscription message transport. It is useful for connections with remote locations where a small code is required and/or network bandwidth is a priority. The version currently in use is MQTT-SN [33], acronym of MQTT for Sensor Networks. It is aimed at embedded devices on non-TCP/IP networks, whereas MQTT itself explicitly expects a TCP/IP stack.
  - Simple/Streaming Text Oriented Messaging Protocol (STOMP) [34] is a text-based protocol, making it more analogous to HTTP in terms of how it looks under the covers. It is a very simple and easy to implement protocol, coming from the HTTP school of design; the server side may be hard to implement well, but it is very easy to write a client to get yourself connected. For example you can use Telnet to login to any STOMP broker and interact with it.
  - Advanced Message Queuing Protocol (AMQP) [35], this protocol was designed as an open replacement for existing proprietary messaging middleware. Its greatest strengths are reliability and interoperability. It also provides a wide range of features related to messaging, including reliable queuing, topic-based publish-and-subscribe messaging, flexible routing, transactions, and security. AMQP exchanges route messages directly in fan-out form, by topic, and also based on headers.

  In general, these protocols are designed to manage a large number of very small size packets. The performance of IoT services is influenced by:
  - Energy consumption;
  - System lifetime: a measure of the longevity of the nodes;
  - Latency: the time delay experienced in a system;
  - Delay and delay variation: refer to delay and delay variation in data collection from nodes;
  - Bandwidth, capacity and throughput: indicate the capacity of a sensor network to send data over a link within a given time.
- Document Access and Navigation service type is associated mostly with web browsing and file access. Therefore this service type is also referred to as *browsing* (BR). It is associated with protocols like HTTP/TCP and is more sensitive to:
  - packet loss rate, as loss causes retransmissions, which are then translated in jitter and delays.
  - delay;
  - jitter;
  - throughput.

B. TYDER ARCHITECTURE

TYDER architecture mainly relies on two macro modules, the Client Side and the Server Side, both connected to MNOs from which they receive information regarding wireless networks, such as Network Operator Type, Network ID, Network Position and Traffic Load Container. The two modules are connected to each other and exchange information. The client module sends feedback to the server side and the server module sends the list of networks with their respective reputations to the Client Side. This architectural solution is illustrated in Figure 2.

A possible software implementation and deployment of TYDER can involve two major apps. The Client Side consists of a special app installed in the user's device. The Server Side requires an app in the cloud connected to a database, so that all users can at any time query the application and get the information necessary for the operation of the proposed system. The two apps would work at the application layer and are able to communicate with the lower layers regardless of what communication protocol is employed. This is to be able to identify the various types of services employed.

Fig. 3 shows the stack of the two modules taken into consideration. As one can see, both client and server solutions work at the application layer of the TCP/IP network

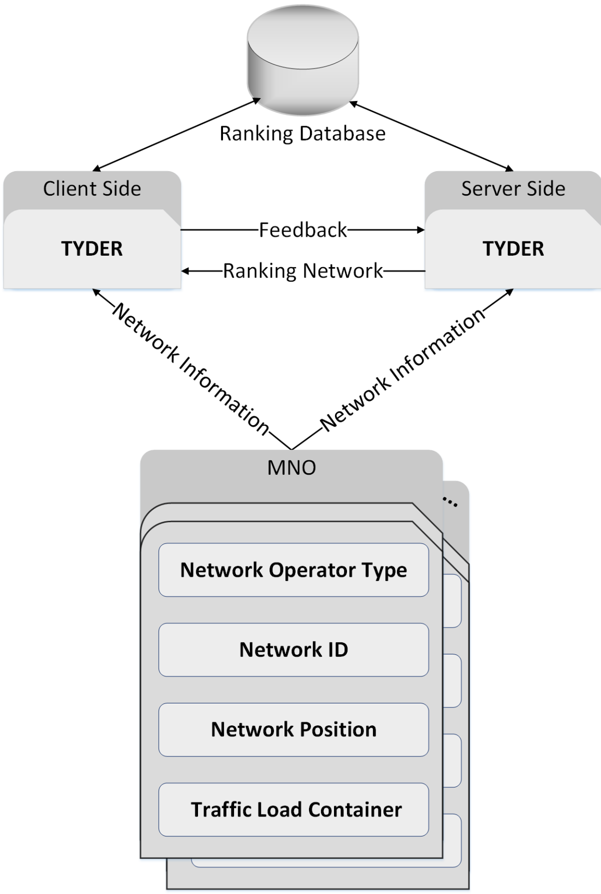


FIGURE 2: TYDER Architecture

stack. This allows for total independence of the transport level protocols used by the services and the physical layer on which the information will travel (eg LTE, NB-IoT or WiFi).

C. CLIENT SIDE MODULE

The Client Side takes care of managing two very important parts of the system:

- Ranking received from the server side in order to decide which is the best candidate network for its interest;
- Creation and delivery of feedback to the server-side.

The Fig. 4 shows how these two parts are integrated into the client-side module.

1) Candidate Network Selection

The first function of the client is to select the candidate network. When the user accesses the system of heterogeneous networks for the first time, when they move within it or when they switch from one application to another, they must query the server to obtain the ranking of the network. During the query, the client side will send information about the user profile and service type to the server. The server side will send the ranking of the network containing the reputation of the available networks, associated to the specific service requested.

The Data Collector block will collect this information together with the information contained in the User Profile and the Service Profile and pass it to the Network Filter. The Network Filter block eliminates all networks that do not meet the minimum/maximum criteria. For example, if the device speed exceeds the maximum quota supported by the network standard, it will be eliminated from the possible choices. It will then get the network that has the best reputation for the service requested at that time. In this way the device can connect to the network able to offer the best QoS.

The User Profile block will contain all user preferences and will also be useful for managing information on the position of the device in order to store the user's mobility models. The Device Profile block contains the specific properties of the device, among them very important is the location service that allows us to identify the position of the device and its speed of movement. In terms of speed, we can have three different types of speed: high speed (more than 15 Km/h), the speed of a user on a vehicle; low speed (less than 15 Km/h), the speed of a user on foot or a slow vehicle; stationary users when the user does not need mobility support.

The Service Profile contains all the information regarding the type of service that the user is currently using. The services are grouped into four macro areas that correspond to the most used services and with greater scope for development: VI, GM, BR, and IoT, respectively. Each of them will be associated with an identifier that will enable the use of a specific function, most appropriate to that service type.

The Network Profile contains all the information concerning the network, such as the ID, and type of network. The Network Ranking contains the ranking network scores that are received from the server database.

The diagram in Fig. 5(a) shows the steps necessary to perform the candidate network selection, the first function of the client side:

1) The Client Side sends a query (A) containing the type of service used, to the database that contains the network ranking.
2) The database sends to the Client Side the answer to the query (B), the ranking of the networks available to the user at that precise moment. The ranking contains a list of tuples that contains:
   - ID of the network;
   - Type of network;
   - Value of its reputation.

   This information is sent to the client via TCP/IP.
3) The Client Side, through the Data Collector module, groups together the necessary informations to select the candidate network. This informations are:
   - Ranking Network;
   - Service Profile;
   - User Profile;
   - Device Profile.
4) The data collector sends this information to the network filter module. The network filter module makes a

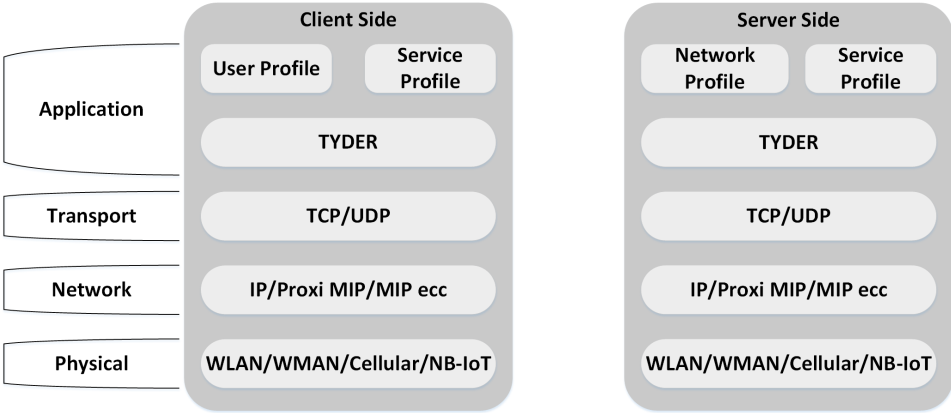


FIGURE 3: Client/Server Side Protocol Stack

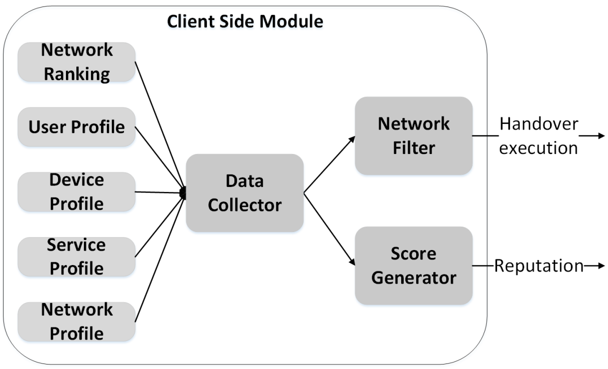


FIGURE 4: Client Module

selection on available networks, eliminating networks that do not be included in the minimum/maximum criteria.
5) The network filter module returns the IDâĂŹs candidate network (E).
6) The Client Side sends the ID of the candidate network (F) to the Mobile Call Handoff (MCH).

2) Network Evaluation

The second function of the client side is the evaluation of the network. Regularly (e.g. every minute) the device sends network evaluation info regarding its service use to the server side. The Data Collector block will contain relevant information about: Service Profile, Device Profile, and Network Profile. This information will be sent to the Score Generator block. The function of this block is to calculate the value to be assigned to the network and send it to the server side, in form of feedback. Depending on the type of service used, a specific score function will be activated. However, the server will always receive a value between 0 and 1.

Diagram 5(b) shows the sequence of actions of network evaluation, the second function of client side. The client side calculates and sends the reputation of the network to the server side.

1) The Client Side requires reputation on the Server Side. The Server Side sends the information (A) in the form of a network ranking for that specific type of traffic.
2) The Client Side groups the information necessary for the calculation of the reputation through the data collector module (B).
3) This information are service profile, device profile and network profile (C).
4) Through the use of the utility functions, the score generator module calculates the feedback of the used network and sends it to the server side (D).

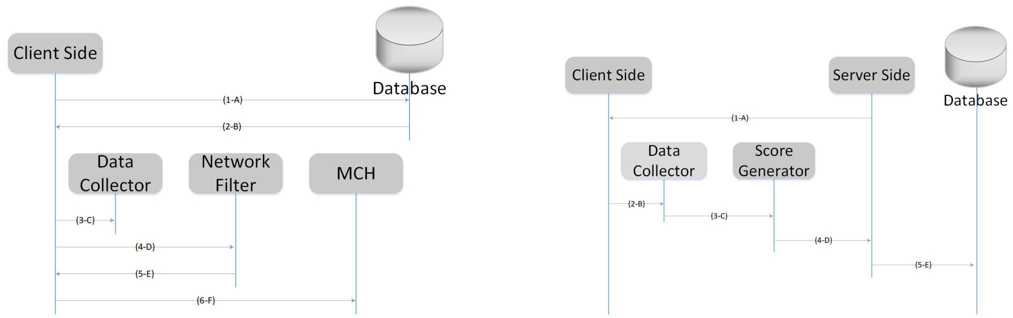
D. CLIENT SIDE ALGORITHM

The purpose is to provide feedback about the network that user was using or is using. The information sent to the server includes:

- Feedback value;
- Type of service;
- ID network;
- Time stamp.

The first problem is to calculate the feedback and based on which attributes to calculate it. In fact, we know that different types of services have different needs. So, we differentiate the type of service. For each service type, there is a Score Function. The evaluation of this Score Function results in a score which is in fact the feedback value.

Contributing to the calculation of feedback will be a very different and often conflicting set of attributes. For this reason, we use a MADM, extremely used in literature in situations where there are some conflicting attributes. There are numerous utility functions in the literature, many of which are specific to the video service. We can use one of these or modify the single part of this and weights, e.g., multiplicative exponent weighting, additive logarithm weighted, etc. In fig. 6 it is shown how utility functions affect the performance of the score for the various types of traffic.



(a) (b)

FIGURE 5: Client Procedures

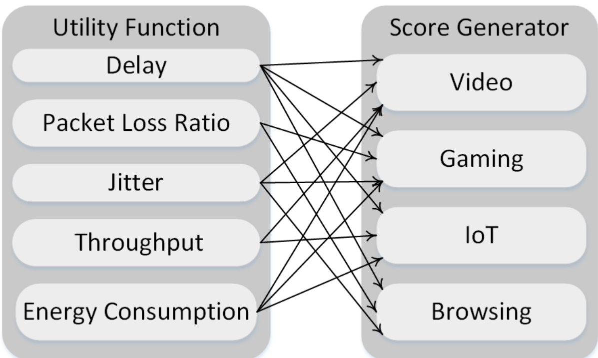


FIGURE 6: Utility Function and Score Generator

1) Client Algorithm

The purpose of TYDER algorithm, described in Alg. 1, is to calculate the network feedback based on the type of service used. The algorithm receives incoming data on the network and the type of traffic used. As already indicated, the four types of traffic considered are VI, GM, BR, and IoT.

Once the network has been identified by the transmitted ID, check the type of traffic and, depending on it, the corresponding function is applied. Feedback is calculated based on QoS values. These are used to derive the values of the individual Utility Functions (UF), which are added together using an additive logarithm weighted model. In the algorithm presented, the weights of the individual UF are attributed according to their importance within the requested service. In this way the UFs are differentiated within the Score Function (SF). SF, UF and the other aspects are explained in more details in the following paragraph.

2) UFs

Different UFs are used by TYDER. They are described next.

- UF for Delay - see eq. (1):

$$u_D = \begin{cases} 1 & \text{if } 0 \leq t \leq T_{min} \\ \frac{-1}{(T_{max}-T_{min})}(t - T_{max}) & \text{if } T_{min} \leq t \leq T_{max} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Algorithm 1: Feedback Computation

Result: Return the value of feedback for the used network.

input: typeOfService = type of service used by the user;
ID = identifier of the network to which the user is connected;
WoJ = weight of Jitter;
WoT = weight of Throughput;
WoD = weight of Delay;
WoPLR = weight of Packet Loss Ratio;
WoEC = weight of Energy Consumption.

begin
  for $i \leftarrow 0$ **to** $listOFNetwork$ **do**
    if *ID = i* **then**
      if *typeOfService == VI* **then**
        feedback = $WoJ * ln(u_J) + WoT * ln(u_T) + WoD * ln(u_D)$
      end
      if *typeOfService == GM* **then**
        feedback = $WoD * ln(u_{D_i}) + WoPLR * ln(u_{PLR_i}) + WoJ * ln(u_{J_i})$
      end
      if *typeOfService == BR* **then**
        feedback = $WoPLR * ln(u_{PLR_i}) + WoD * ln(u_{D_i}) + WoJ * ln(u_{J_i})$
      end
      if *typeOfService == IoT* **then**
        feedback = $WoEC * ln(u_{EC_i}) + WoD * ln(u_{D_i}) + WoT * ln(u_{T_i})$
      end
    end
  end
end

where: u_D is the UF who estimated delay; T_{min} minimum delay of traffic; T_{max} maximum delay of traffic; t real delay of traffic.

- UF for Packet Loss Ratio (PLS) - see eq. (2):

$$u_{PLR} = 100 * (\frac{MissedPackets}{TotalPackets}) \tag{2}$$

where: u_{PLR} is the UF who estimated the packet loss ratio.

- UF for Jitter - see eq. (3):

$$u_J = D_{peak-to-peak} + 2 * n * R_{rms} \tag{3}$$

where: u_j is the UF who estimated total jitter; $D_{peak-to-peak}$ is deterministic jitter; n is based on the bit error rate (BER) required of the link; R_{rms} is random jitter.

- UF for Throughput - see eq. (4):

$$u_T = \begin{cases} 0 & \text{if } Th < Th_{min} \\ 1 - e^{\frac{-\alpha * Th^2}{\beta + Th}} & \text{if } Th_{min} \le Th \le Th_{max} \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

where u_T is the UF who estimated Throughput, Th is the predicted average throughput for each of the candidate networks (Mbps), Th_{min} is the minimum throughput necessary to obtain the requested service (Mbps), Th_{max} is maximum achievable throughput, α and β are two positive parameters which determine the shape of the UF (no unit).

- UF for Energy Consumption - see eq. (5):

$$u_{EC} = (r_t + Th_i * r_d) * t + c \tag{5}$$

where: u_{EC} is the UF who estimated energy consumption (Joule); t represents the transaction time (seconds); r_t is the mobile deviceâĂŹs energy consumption per unit of time (W); Th_i is the available throughput (kbps) provided by RAN i; r_d is the energy consumption rate for data/received stream (Joule/Kbyte), c is a constant (no unit) [11].

3) Additive Logarithm Weighted method

The UF that have been defined are grouped using the Additive Logarithm Weighted (ALoW) method [12]. The choice was made due to the mathematical properties of logarithms. In fact, the UFs are grouped together with a series of summaries unlike what happens in exponential multiplication methods. In this way, if a UF assumes a value of zero, it will affect in a less decisive way the final choice of the candidate network. Formula from eq. (6) shows ALoW method.

$$ln(U^i) = \sum_j w^j * ln(u_j) \tag{6}$$

where $ln(U^i)$ is the natural logarithm of U and i indicates the i-th network. Sum of the j-th weight multiplied by the natural logarithm of the UFs, where j indicates the j-th UF. The result obtained is the reputation value that we attribute to the network used.

In the following manner, the SFs for each type of service are computed.

- SF for VI - see eq. (7):

$$ln(U_i^V) = w_J * ln(u_{J_i}) + w_T * ln(u_{T_i}) + w_D * \\ * ln(u_{D_i}) \tag{7}$$

- SF for GM - see eq. (8):

$$ln(U_i^G) = w_D * ln(u_{D_i}) + w_{PLR} * ln(u_{PLR_i}) + \\ + w_J * ln(u_{J_i}) \tag{8}$$

- SF for IoT - see eq. (9):

$$ln(U_i^{IoT}) = w_{EC} * ln(u_{EC_i}) + w_D * ln(u_{D_i}) + \\ + w_T * ln(u_{T_i}) \tag{9}$$

- SF for BR - see eq. (10):

$$ln(U_i^B) = w_{PLR} * ln(u_{PLR_i}) + w_D * ln(u_{D_i}) + \\ + w_J * ln(u_{J_i}) \tag{10}$$

where U_i is the SF for i^th network, $u_{J_i}^{w_J}$, $u_{T_i}^{w_T}$, $u_{D_i}^{w_D}$, $u_{PLR_i}^{w_{PLR}}$, and $u_{EC_i}^{w_{EC}}$ are the UFs which will be defined for jitter, throughput, delay, packet loss rate, and energy consumption respectively.

w_J, w_T, w_D, w_{PLR}, and w_{EC} are the weights that are attributed to each UF within the specific score. The weight value will be given based on the importance of the single UF within the specific SF. In general, the rule from eq. (11) is applied:

$$\sum_j w_j = 1 \tag{11}$$

For defining the values of the weights, the Analytic Hierarchy Process (AHP) [12] is used, recommended for defining weight values in case of multi-criteria decisions. In this way, each weight is assigned a value based on its importance with respect to the other attributes that contribute to the formation of the SF.

The AHP method employs five basic phases, through which the weights are obtained, as follows:

1) Development of a hierarchy between the variables involved;
2) Construction of the matrix of pairs comparisons;
3) Determination of relative local weights;
4) Analysis of the consistency of the judgments;

5) Determination of global weights: the principle of hierarchical composition.

For example, in the case of video service, we know that the video is very sensitive to jitter and throughput and less to delay. Therefore, jitter and throughput must have a greater weight than the delay. Among them, however, have a similar importance, so they can have the same weight. The Saaty Scale [36] is based on these principles and provides a scale of values that allows you to give weight to each attribute based on the importance that each of them has compared to the other components. In table 1, this scale of values is presented.

TABLE 1: Example of assumed values by Saaty Scale

Value	Description
1	if i and j are equally important
3	if i is a little more important than j
5	if i is more important than j
7	if i is definitely more important than j
9	if i is absolutely more important than j
$\frac{1}{3}$	if i is a little less important than j
$\frac{1}{5}$	if i is fairly less important than j
$\frac{1}{7}$	if i is definitely less important than j
$\frac{1}{9}$	if i is absolutely less important than j

Through the values defined in the Saaty Scale it is possible to obtain the weights of the single UF. Following the steps necessary for AHP, we create a hierarchy among the variables used, we create a matrix in which we compare our variables to each other, two at a time, in this way we have a relative weight. An example of this process is shown in the table 2.

Once the feedback value is calculated, this is sent to the server to be able to compete in network reputation.

TABLE 2: Example of weight for Video SF

	Jitter	Throughput	Delay	Weight
Jitter	X	1	5	0.4
Throughput	1	X	5	0.4
Delay	$\frac{1}{5}$	$\frac{1}{5}$	X	0.2

E. SERVER SIDE MODULE

1) Network Ranking Provisioning

The server side is responsible for sending the ranking of available networks on the device that request it, and keeping the ranking updated by obtaining feedback from the devices, as shown in fig. 7. Whenever a device requests it, the server side must send the ranking of the networks available to the user. To do this, it must collect information regarding the type of service requested, the networks available to the user and the user's profile, through the Data Collector block. This information is sent to the Network Ranking block that queries the database, based on the type of service requested selects the available networks, and finally sends the user a list of networks with their reputations.

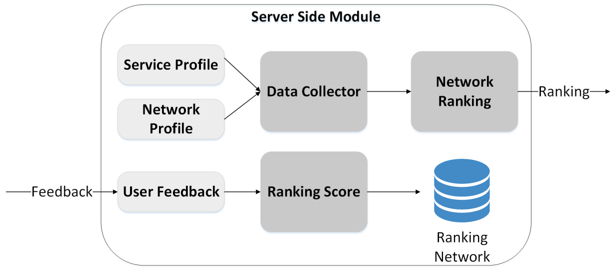


FIGURE 7: Server Module

2) Network Reputation Update

The second function of the server side module is to keep the network reputations database updated. To do this, the Ranking Score block collects the various feedback scores that are sent by the devices. As feedback arrives at this block, it stores them in the database and keeps track of the network's reputation. To keep the value updated, a ranking algorithm is used that takes into account the current reputation of the network and its trend during the day and during the week. This value is stored in a tuple $R_i(V, G, B, I)$ where R_i is the reputation tuple for i^th network, V is the reputation for video service, G is the value of reputation for gaming service, B is the reputation for browsing service, and I is the value of reputation for IoT service.

The server side deals with the management of mainly two system functions:

- Send ranking to the client side, which will manage it;
- Process the feedback received from the client side.

The function of sending the Ranking to the Client Side is the simplest of functions performed on the Server side. In fact, when a user enters the system, he makes a request to the server. This request specifies the type of traffic that you decide to use at that time. The Server queries the networks, based on the type of traffic requested and will send to the Client side, the network or networks that have a better Ranking value at that time.

The second function occurs when the user sends feedback on the network used. This happens every time the user leaves the network, because he has moved away or because the conditions of the network have changed and it is necessary to carry out a handover. The Client Side sends the network feedback to Server, calculated according to the QoS parameters that were obtained during use of the network. The Server Side processes the data sent and updates the database in a way that is consistent with the current situation of the network. In this way the reputation of the network is always updated with the latest info.

The diagram in fig. 8 shows the steps necessary to perform these important functions at the server side:

1) (1-A) Client sends a request to the Server for the best ranked network associated with the service indicated and Server replies with the network ID that matches the Client request.

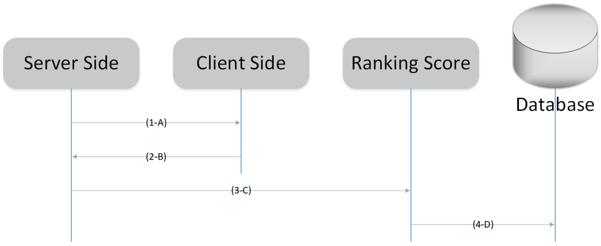


FIGURE 8: Server Procedures

2) (2-B) The Client uses the network and then it sends feedback to the Server regarding the network used.
3) (3-C) The server processes the feedback information according to the Ranking Score algorithm which will be presented later..
4) (4-D) The processed information is sent to the database where it is stored for future use.

3) Server Side Algorithm

On the server side, we have the problem of how to use the values that are sent from the client side. There is a 4-tuple including values that will be stored and processed in order to be returned as a reputation when a user wants to connect to the network. The individual feedback values sent by the user are maintained as they are received for a period of time (e.g. one hour), after which they will be aggregated and saved as an average value, whereas the individual values are deleted to give way to new values. The purpose of the algorithm is to rank the available networks, based on their reputation, attributed over time by clients to each individual network. The algorithm is designed to take into account reputation trend in both short and long term. By combining the trend over a day with that over a week, the overall network ranking value is obtained.

This function is shown in algorithm 2. Called for all available networks, it receives the type of service that the user intends to use, and calculates the ranking for all networks. The ranking takes into account the values stored over time, attributing to them an increasingly smaller value as you move away from the current moment.

The second function, performed according to the algorithm 3, focuses on saving the feedback sent by clients. It is simply attributed to the network used by the user and associated to the type of service requested.

F. COMPUTATIONAL COMPLEXITY

The computational complexity of TYDER is $O(nlogn)$, indeed, lower than the computational complexity of MEW and E-PoFANS, which both have $O(2^n)$.

G. SYSTEM TRAINING

The system needs a period of learning, in fact initially all networks have the same reputation. This changes over time by users entering the system. Users release their feedback that changes network reputations, thus allowing the algorithm

Algorithm 2: Computation of network reputation and update of network ranking

Result: Return the value of reputation for the network.
input: ToS = type of service required.
RN = Reputation Network
RD = Reputation Day
FN = Feedback Network
for *s in type of service available* **do**
  if *ToS = t* **then**
    foreach *day = d in a week* **do**
      $RD_d = FN_{d1} + \sum_{k=2}^{24} FN_{dk} * (1 - \frac{k}{25})$
    end
    $RN = RD_1 + \sum_{i=2}^{7} RD_i * (1 - \frac{i}{8})$
  end
end

Algorithm 3: List of Reputation Networks

Result: Ranking storage.
input: ToS = Type of Service
ID = identification of network
feedback = feedback value attributed by the user to the network
LoN = List of Networks
foreach *network in LoN* **do**
  if *ID = network* **then**
    network[ToS].add(feedback);
  end
end

to function properly. This training period is called system training and was considered in the simulation phase. Training is necessary to provide system consistency.

IV. PERFORMANCE EVALUATION

The evaluation of the TYDER algorithm, was carried out by integrating Python code in models built using the OMNeT++ [37] network simulator. The simulations were performed using the parameters presented in table 3. They respect the main characteristics of the three access technologies used: Wi-Fi, LTE and NB-IoT.

The simulations use the same network configuration, while considering different traffic situations. The traffic configuration involves the coexistence of three different types of access networks. Specifically, the presence of two Wi-Fi access points, an eNB for LTE and 4 eNBs for NB-IoT, this is because this technology is designed to exploit the existing LTE infrastructure.

The simulations were preceded by a training simulation. The training simulation involved the presence of only one user within the simulation system. For each network, it was positioned in the cell center and on the cell edge. For each position request of the 4 types of traffic was simulated, for a duration of 30 seconds each. This allowed to fill the database

TABLE 3: Simulation Networks Parameters

Attributes	Wi-Fi Network	LTE Network	NB-IoT
Technologies	8002.11ac	LTE cat 6	NB-IoT
Number of AP/eNB	2	1	4
Radius Cell	250 m	500 m	5000 m
Transmission Power	20 dBm	46 dBm	23 dBm
Bandwidth	40 MHz	20 MHz	180 KHz
Data rate Max	600 Mbps	300 Mbps	250 kbps
Propagation Model	Hybrid model (in/out)	Hybrid model (in/out)	Hybrid model (in/out)

TABLE 4: Data Rates Used

Service	Data Rate
Video	4 Mbps
Gaming	3 Mbps
Browsing	100 kbps
IoT	2 bytes

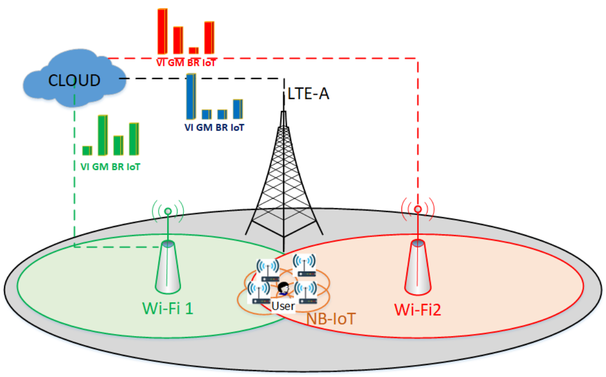


FIGURE 9: Scenario 1: Static User

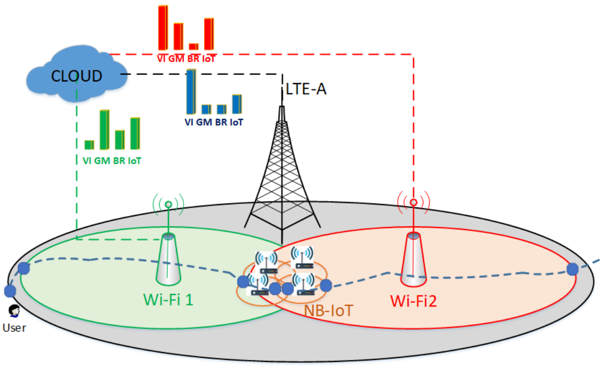


FIGURE 10: Scenario 2: Mobile User with no Background Traffic

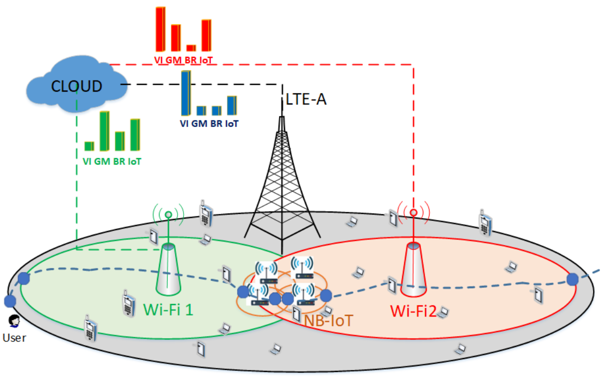


FIGURE 11: Scenario 3: Mobile User with Background Traffic

with the initial simulation data.

The traffic has been differentiated in such a way that it is possible to simulate the four different types of services offered. The different network data rates used are shown in table 4.

A. SCENARIO 1

In the first scenario, a single user without mobility case was considered, as shown in fig. 9. It employs the 4 different types of traffic for the duration of 3 minutes each. This allows for population of the reputation tables of the individual networks and at the same time evaluation of the behavior of the TYDER algorithm in comparison with the situation when the proposed algorithm was not deployed.

The user can use a different type of device (e.g. smartphone, laptop, tablet, smartwatch, mobile device, etc.) and request access to one of the services shown in the Table 4.

B. SCENARIO 2

In the second scenario, we have a pedestrian mobile user, walking at 3 km/h along a linear path of 500 m, as shown in fig. 10. The duration of the simulation is 450 seconds, to allow the user to complete the entire route. This scenario simulates realistically the behavior of pedestrians and vehicles in dense urban environments and heavy traffic.

This second simulation tests the dynamic behaviour of the algorithm over time and space. Along the route, the user performs a service change every 10 seconds, supporting a homogeneous distribution of service requests during the simulation. During user movement, multiple handovers are performed.

C. SCENARIO 3

Finally, in the third scenario, background traffic was considered. In scenario 3, other 200 users were introduced, distributed in a pseudo-random way among the different networks and who request the 4 different types of traffic, as shown in fig. 11. The traffic of these users significantly influences the reputation of the various networks.

The simulation has the same duration as in the second simulation and the evaluations are carried out by focusing on a pedestrian user with linear mobility with a speed of 3 Km/h. This scenario allows for testing the behavior of the algorithm in a realistic situation with ordinary traffic. The number of users introduced enables validation of the operation of the algorithm in loaded traffic conditions.

V. RESULTS

In this section the proposed TYDER algorithm is evaluated and compared with MEW [10] and E-PoFANS [11] algorithms. Fig. 12 shows the type of traffic selected and analyzes the performances between the proposed TYDER and the two algorithms used for comparison. The obtained results are oriented to QoS performance evaluation. In the case of VI, GM, BR, and IoT traffic types, an improvement in terms of maximum throughput that a user could obtain is observed.

Figures 15, 16, and 17 show the throughput, packet delay and packet loss rate, respectively, when the user is moving as explained in scenario 2. The mobility of the user implies the consideration of dynamics aimed at intelligent selection of the access network. The proposed TYDER algorithm is able to select in real time the best candidate based on the QoS parameters examined. In these cases we can see a considerable increase in average throughput and a decrease in the packet delay and loss rate for both types of traffic.

Figures 18, 19, and 20 show the throughput, packet delay and packet loss rate, respectively, when a dense traffic scenario has been proposed. As indicated in scenario 3, the mobility of the user is combined with the traffic contribution of the other users present in the scenario. Mobility and background traffic have significantly influenced the choice of the candidate network. In fact, the reputation obtained the input of users' feedback in the simulated system. As we can see from the results, which show a better trend compared to the algorithm compared.

Finally, figures 21, 22, 23, and 24, show the average throughput trends during the simulation in scenario 3 for the different types of traffic. The graphs are plotted with a 10-second step, to take into account the change in reputation caused by background traffic. It has been simulated that a user changes his traffic type 30% of the time, randomly. This leads to a trend in average throughput that is not perfectly linear. In general, we can see how the reputation algorithm behavior performs better than MEW and E-PoFANS, in all four services required.

TYDER has an average improvement of 8% against MEW and 5% for E-PoFANS. This improvement is noted above all in the Video and IoT services, where TYDER records the biggest improvements.

VI. CONCLUSION AND FUTURE WORK

This work has proposed TYDER, a traffic type-based solution for reputation-based network selection. TYDER computes network reputation by taking into account the different

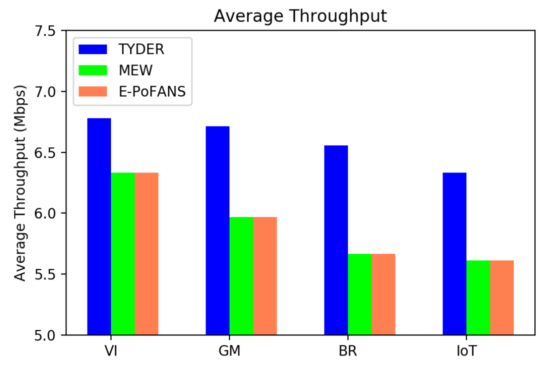


FIGURE 12: Average Throughput Scenario 1

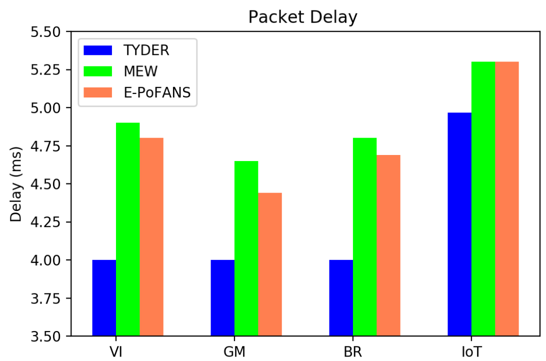


FIGURE 13: Packet Delay Scenario 1

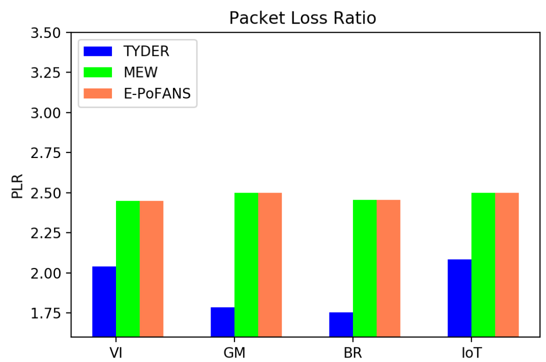


FIGURE 14: Packet Loss Ratio Scenario 1

types of traffic and by performing most appropriate network selection makes a significant improvement in the system performance, with regard to the QoS parameters considered for each traffic type. This makes it an ideal candidate for the optimization of 5G networks and D2D technologies.

To evaluate the effectiveness of the algorithm, it was compared with two different algorithms, MEW and E-PoFANS. Through the OMNeT++ network simulator, they were tested in three different scenarios. It proved to have better per-

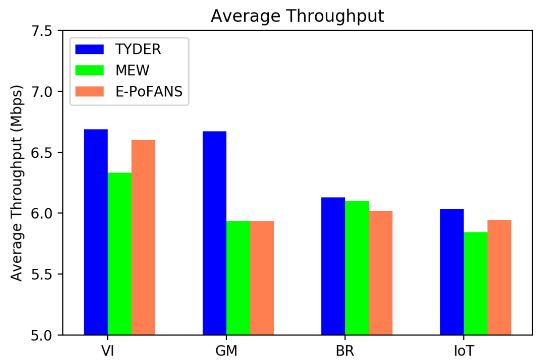


FIGURE 15: Average Throughput Scenario 2

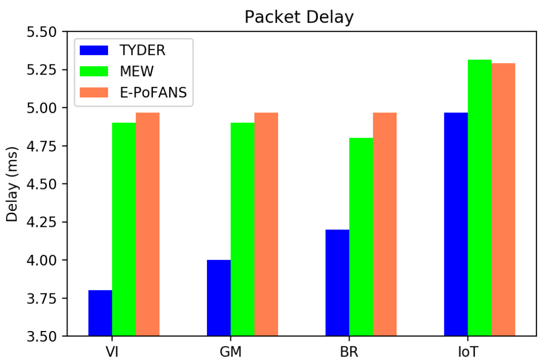


FIGURE 16: Packet Delay Scenario 2

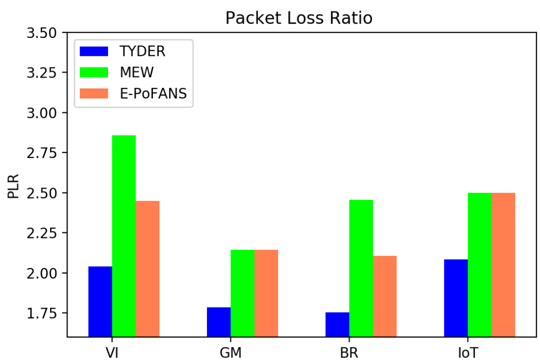


FIGURE 17: Packet Loss Ratio Scenario 2

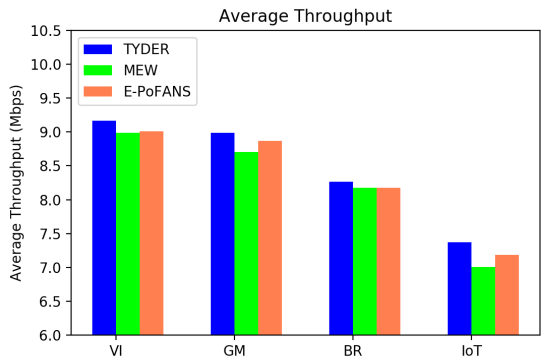


FIGURE 18: Average Throughput Scenario 3

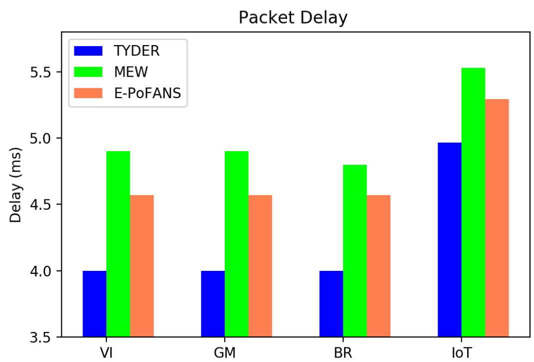


FIGURE 19: Packet Delay Scenario 3

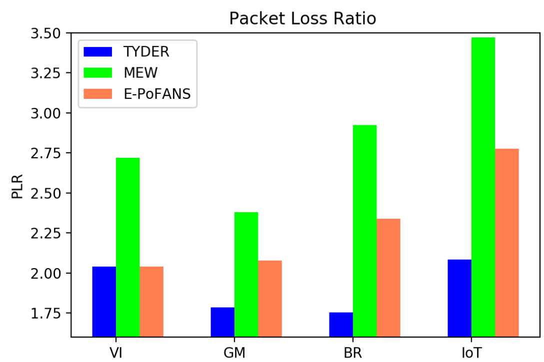


FIGURE 20: Packet Loss Ratio Scenario 3

formances in terms of QoS. TYDER showed an average improvement on all QoS values taken into consideration on MEW and E-PoFANS, about 8% and 5% respectively.

In the future, the algorithm can be improved by allowing for a more realistic mixed choice of the type of service requested by the user, not only individual traffic type. This would allow for the use multiple types of services at the same time, thus improving the reputation of the networks present in the system.

ACKNOWLEDGMENT

The research activities described in this paper have been conducted within the R&D project "Cagliari2020" partially funded by the Italian University and Research Ministry (grant # MIUR_PON04a2_00381). The support of the European Union's Horizon 2020 Research and Innovation programme under Grant Agreement no. 688503 for NEWTON project (http://newtonproject.eu) is gratefully acknowledged.

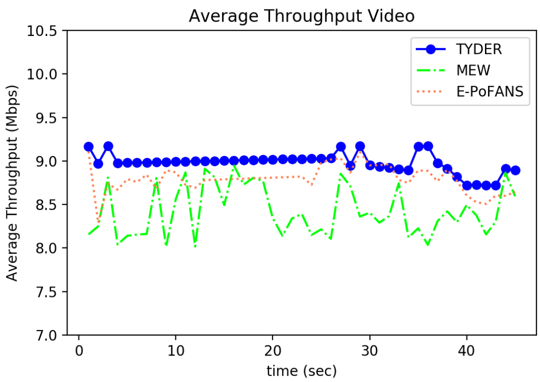


FIGURE 21: Average Throughput Scenario 3 VI

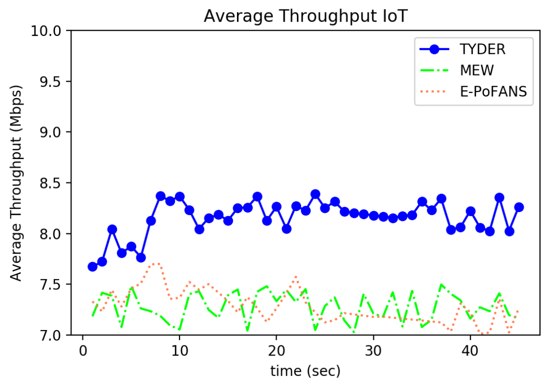


FIGURE 24: Average Throughput Scenario 3 IoT

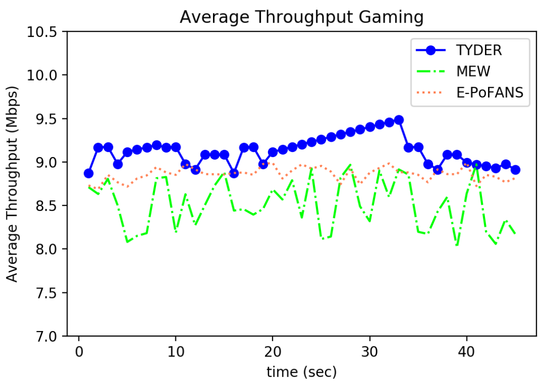


FIGURE 22: Average Throughput Scenario 3 GM

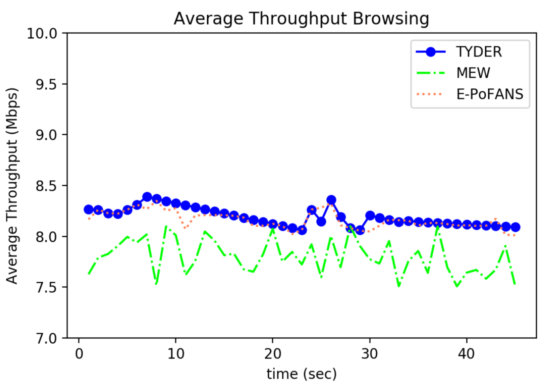


FIGURE 23: Average Throughput Scenario 3 BR

REFERENCES

[1] R. B. Santos, F. R. M. Lima, W. C. Freitas, and F. R. P. Cavalcanti, "QoS based radio resource allocation and scheduling with different user data rate requirements for OFDMA systems," in 2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 1–5.

[2] Z. H. Hussien and Y. Sadi, "Flexible radio resource allocation for machine type communications in 5g cellular networks," in 2018 26th Signal Processing and Communications Applications Conference (SIU), pp. 1–4.

[3] A. A. Senan, S. Manjunath, and S. Abdulhayan, "Direct device-to-device communication in 5g networks," in 2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), pp. 216–219.

[4] IEEE 802.21. [Online]. Available: http://www.ieee802.org/21/

[5] LTE-advanced - the mobile broadband standard.

[6] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. D. Silva, F. Tufvesson, A. Benjebbour, and G. Wunder, "5g: A tutorial overview of standards, trials, challenges, deployment, and practice," vol. 35, no. 6, pp. 1201–1221.

[7] Cisco visual networking index: Global mobile data traffic forecast update, 2016âĂŞ2021 white paper. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[8] R. Ding and G.-M. Muntean, "Device characteristics-based differentiated energy-efficient adaptive solution for video delivery over heterogeneous wireless networks," in 2013 IEEE Wireless Communications and Networking Conference (WCNC), pp. 4588–4593.

[9] T. Bi and G. M. Muntean, "Location-aware network selection mechanism in heterogeneous wireless networks," in 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 583–588.

[10] C. Desogus, M. Anedda, and M. Murroni, "Real-time load optimization for multimedia delivery content over heterogeneuos wireless network using a MEW approach," in 2017 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), pp. 1–4.

[11] R. Trestian, O. Ormond, and G. M. Muntean, "Enhanced power-friendly access network selection strategy for multimedia delivery over heterogeneous wireless networks," vol. 60, no. 1, pp. 85–101.

[12] L. Wang and G. S. G. S. Kuo, "Mathematical modeling for network selection in heterogeneous wireless networks #x2014; a tutorial," vol. 15, no. 1, pp. 271–292.

[13] A. Chinnappan and R. Balasubramanian, "Complexity #x2013;consistency trade-off in multi-attribute decision making for vertical handover in heterogeneous wireless networks," vol. 5, no. 1, pp. 13–21.

[14] A. Habbal, S. I. Goudar, and S. Hassan, "Context-aware radio access technology selection in 5g ultra dense networks," vol. 5, pp. 6636–6648.

[15] T. Bi, R. Trestian, and G. M. Muntean, "Reputation-based network selection solution for improved video delivery quality in heterogeneous wireless network environments," in 2013 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), pp. 1–8.

[16] S. Radouche, C. Leghris, and A. Adib, "MADM methods based on utility function and reputation for access network selection in a multi-access mobile network environment," in 2017 International Conference on Wireless Networks and Mobile Communications (WINCOM), pp. 1–6.

[17] Y. Lin and H. Shen, "CloudFog: Leveraging fog to extend cloud gaming for thin-client MMOG with high quality of service," vol. 28, no. 2, pp. 431–445.

[18] S. Baidya and M. Levorato, "Edge-assisted content and computation-driven dynamic network selection for real-time services in the urban IoT," in 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 796–801.

[19] J. Monks and G. M. Muntean, "A distributed energy-aware cooperative multimedia delivery solution," in 2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops), pp. 66–74.

[20] S. Hu, X. Wang, and M. Z. Shakir, "A MIH and SDN-based framework for network selection in 5g HetNet: Backhaul requirement perspectives," in 2015 IEEE International Conference on Communication Workshop (ICCW), pp. 37–43.
[21] L. Zou, R. Trestian, and G. M. Muntean, "E #x00b3;DOAS: Balancing QoE and energy-saving for multi-device adaptation in future mobile wireless video delivery," vol. PP, no. 99, pp. 1–15.
[22] M. M. Hasan, S. Kwon, and J. Na, "Adaptive mobility load balancing algorithm for LTE small-cell networks," vol. Vol. 17,, pp. 2205–2217.
[23] G. Araniti, P. Scopelliti, G. M. Muntean, and A. Iera, "A hybrid unicast-multicast network selection for video deliveries in dense heterogeneous network environments," pp. 1–11.
[24] ITEC âĂŞ dynamic adaptive streaming over HTTP.
[25] L. Yong and W. Bin, "Adaptive rate algorithm for DASH streaming media based on probabilistic prediction," in 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 1186–1189.
[26] R. Pantos. HTTP live streaming 2nd edition. [Online]. Available: https://tools.ietf.org/html/draft-pantos-hls-rfc8216bis-03
[27] Real time streaming protocol (RTSP) - RFC 2326. [Online]. Available: https://www.ietf.org/rfc/rfc2326.txt
[28] V. Jacobson, R. Frederick, S. Casner, and H. Schulzrinne. RTP: A transport protocol for real-time applications. [Online]. Available: https://tools.ietf.org/html/rfc3550
[29] Open game protocol. [Online]. Available: http://www.open-game-protocol.org/
[30] N. R. Council. Executive summary of the NRC report on transport protocols for department of defense data networks. [Online]. Available: https://tools.ietf.org/html/rfc939
[31] J. Postel. UDP - user datagram protocol. [Online]. Available: https://tools.ietf.org/html/rfc768
[32] MQTT for sensor networks - MQTT-SN | MQTT. [Online]. Available: http://mqtt.org/2013/12/mqtt-for-sensor-networks-mqtt-sn
[33] MQTT. [Online]. Available: http://mqtt.org/
[34] STOMP. [Online]. Available: http://stomp.github.io/
[35] AMQP - advanced message queuing protocol. [Online]. Available: https://www.amqp.org/
[36] A. Chunan and F. Feidong, "A supplement to saaty's consistency theory of judgment matrix in the analytic hierarchy process," in 2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE), pp. 603–607.
[37] OMNeT++ discrete event simulator - home. [Online]. Available: https://www.omnetpp.org/

CRISTINA DESOGUS received her Bachelor Degree in Computer and her Master Degree in Telecommunication Engineering from the University of Cagliari, Italy, in September 2016, with a thesis on 'Real-Time Load Optimization for multimedia Transmission on Heterogeneous Wireless Network with Game Theory'. She is currently pursuing the Ph.D. degree in the Multimedia Communication Lab (MCLab), Department of Electrical and Electronic Engineering (DIEE), University of Cagliari, under the supervision of Prof. Maurizio Murroni. She was a visiting researcher at Dublin City University under the supervision of Dr. Gabriel-Miro Muntean, performing research on reputation of the new generation networks with focus on IoT networks and 5G applications. She is a member of the IEEE Communications Society, IEEE Broadcast Technology Society and IEEE BTS Young Professionals. Her research interests are network planning, coverage optimization and optimization algorithms, real-time applications, wireless networks and network selection, rate control, mobile device energy consumption and energy efficiency, adaptive multimedia streaming, heterogeneous radio access environments, multicast, and 5G applications.

MATTEO ANEDDA received a B.Sc. degree in electronic engineering and a M.Sc. degree (summa cum laude) in telecommunication engineering from the University of Cagliari in 2011 and 2012, respectively, where he is currently an Assistant Professor. He has been a visiting Erasmus student for eight months with the University of Basque Country, Bilbao, Spain, in 2010, where he has carried out his M.Sc. thesis under supervision of Prof. P. Angueira. For his M.Sc. thesis Dr. Anedda has received an award from the Order of the Engineers of Cagliari. He was a visiting researcher with Dublin City University, Ireland, under the supervision of Prof. G. Muntean in 2015 and Universidad de Montevideo, Uruguay, under the supervision of Prof. R. Sotelo in 2016, for seven months each. Dr. Anedda's research interests are network planning, coverage optimization and optimization algorithms, real-time applications, wireless networks and network selection, rate control, mobile device energy consumption and energy efficiency, adaptive multimedia streaming, and heterogeneous radio access environment. He is a member of the IEEE Communications, IEEE Broadcast Technology, and IEEE Vehicular Technology societies.

MAURIZIO MURRONI received a M.Sc. degree in Electronic Engineering in 1998 and a Ph.D. degree in Electronic Engineering and Computers in 2001 from the University of Cagliari, where he is associate professor with Dept. of Electrical and Electronic Engineering (DIEE). He was a graduate visiting scholar at the School of Electronic Engineering, Information Technology and Mathematics, University of Surrey, UK in 1998 and a visiting Ph.D. scholar at the Image Processing Group, Polytechnic University, Brooklyn, NY, USA, in 2000. In 2006 and 2014, he was visiting lecturer at the Dept. of Electronics and Computers at the Transilvania University of Brasov in Romania and in 2011 visiting professor at the Dept. Electronics and Telecommunications, University of Basque Country, Spain. Since October 2011, he is coordinator of the research unit of the Italian University Consortium for Telecommunications (CNIT) at the University of Cagliari. Since 2016 is chair of the IEEE Broadcast Technology Society Italy chapter. He is distinguished lecturer for the IEEE Broadcast Technology Society and associate editor for the IEEE Transactions on Broadcasting. Dr. Murroni is co-author of many journal articles and peer-reviewed conference papers and received several best paper awards. He served as chair for various international conferences and workshops while he was guest editor for several journals. He was co-author of the 1900.6-2011 - IEEE Standard for Spectrum Sensing Interfaces and Data Structures for Dynamic Spectrum Access and other Advanced Radio Communication Systems. His research focuses on quality of experience, multimedia data transmission and processing, broadcasting, cognitive radio system, signal processing for radio communications. Dr. Murroni is senior member of IEEE, IEEE ComSoc, IEEE BTS, IEEE VTS, IEEE SPS.

GABRIEL-MIRO MUNTEAN is an Associate Professor with the School of Electronic Engineering, Dublin City University (DCU), Ireland, and the Co-Director of the DCU Performance Engineering Laboratory. He has published over 300 papers in top-level international journals and conferences, authored three books and 17 book chapters, and edited six additional books. His research interests include quality, performance, and energy saving issues related to multimedia and multiple sensorial media delivery, technology-enhanced learning, and other data communications over heterogeneous networks. He is an Associate Editor of the IEEE Transactions on Broadcasting, the Multimedia Communications Area Editor of the IEEE Communications Surveys and Tutorials, and chair and reviewer for important international journals, conferences, and funding agencies. He is a Project Coordinator for the EU-funded project NEWTON (http://www.newtonproject.eu).