A Scheduling Scheme for Throughput Optimization in Mobile Peer-to-Peer Networks

Odysseas Shiakallis[1], Constandinos X. Mavromoustakis[1], George Mastorakis[2], Athina Bourdena[1], Evangelos Pallis[2], Evangelos Markakis[3], Ciprian Dobre[4]
[1]*E-mail: Odysseas.Shiakallis@cyfusion.com, mavromoustakis.c@unic.ac.cy, bourdena.a@unic.ac.cy*
[1]*University of Nicosia, Department of Computer Science, 46 Makedonitissas Ave., 2414 Engomi, Nicosia, Cyprus*
[2]*E-mail: gmastorakis@staff.teicrete.gr, pallis@pasiphae.eu*
[2]*Technological Educational Institute of Crete, Department of Informatics Engineering, Estavromenos, 71500, Heraklion, Crete, Greece*
[3]*E-mail: markakis@pagni.gr*
[3]*University General Hospital of Heraklion, IT Department, Voutes 71110, Heraklion, Crete, Greece*
[4]*E-mail: ciprian.dobre@cs.pub.ro*
[4]*University Politehnica of Bucharest, Faculty of Automatic Control and Computers, 313, Splaiul Independentei, 060042, Bucharest, Romania*

Abstract— Mobile Cloud Computing (MCC) paradigm includes all the emerging technological advances, mechanisms and schemes for the efficient resource offloading and the energy-efficient provision of services to mobile users. In addition, the mobile nodes will act as flexible networking points in emerging mobile networking architectures, where several challenges have to be addressed, like the high energy consumption and the data packets transmission failure, under a Mobile Peer-to-Peer (MP2P) approach. Towards addressing such challenges, several factors that contribute to the increased consumption of the energy, have to be investigated, as well as issues related with the loss of data during the provision of services. In this framework, a Traffic-based S-MAC scheme is proposed in this chapter, towards increasing the data exchange and minimize the energy consumption, between mobile nodes operating under an Ad-Hoc approach. The performance of the proposed scheduling scheme was thoroughly evaluated, through a number of simulation experiments.

Keywords- Energy Conservation, Throughput Optimization, TS-MAC, Ad-hoc Networks, Scheduling Mechanism, Sleep Wake state, Medium Access Control, Duty Cycle.

I. Introduction

The portable devices (e.g. smart phones, tablets etc.) in emerging mobile network architectures have shrunk in size, incorporating advanced functions and mechanisms (Mastorakis, Mavromoustakis, Bourdena, Kormentzas & Pallis, 2013). This allows a node (or a mobile device) to act as a wireless terminal, or as a repeater creating a network for efficient resources sharing (Mavromoustakis, Dimitriou & Mastorakis, 2013). A self-organizing and adaptive collection of such mobile devices connected with wireless links is now referred to as a Mobile Ad Hoc Network (i.e. MANET). An Ad Hoc Network is not in need of any centralized control. The network should be able to detect any new nodes in range and induct them unobstructed (Mousicou, Mavromoustakis, Bourdena, Mastorakis, & Pallis, 2013). Nevertheless, if any node moves out of the range of the network, the remaining nodes should automatically reconfigure themselves, in order to adapt in the new topological scenario. A working group named MANET has been set up by the Internet Engineering Task Force (IETF) for promoting research in this area. Most usually, there are two types of architectures in Ad Hoc Networks: the flat and the hierarchical architecture (Chakrabarti & Mishra, 2001; Toh, 2002). Each node has a transceiver, an antenna and a power source. The properties of these nodes can vary regarding their size, transmission range, battery power and processing ability. It is common that some nodes can be used as servers, by lending themselves, others as clients and even some may be operational to act in conjunction (server and client), depending on the situation. There are specific cases though, where each node may need to act as a router, in order to channel information between nodes (Royer & Toh 1999; Haas & Tabrizi1998).

In this context, a Traffic Sensor Media Access Control (TS-MAC) protocol is examined in this chapter, based on S-MAC, a Sensor Media Access Control that uses three atypical techniques to reduce energy consumption and support self-configuration (Mavromoustakis, Bourdena, Mastorakis, Pallis, & Kormentzas, 2014). In order to bring energy consumption to the lowest level possible in listening to an idle channel, nodes periodically sleep (Mavromoustakis, Mousicou, Papanikolaou, Mastorakis, Bourdena, & Pallis, 2015). Neighboring nodes form *virtual clusters* so that they can auto-synchronize between sleep schedules. S-MAC, which was influenced by PAMAS, also sets nodes to sleep during transmissions of other nodes. Unlike PAMAS, it only uses in-channel signaling. Finally, S-MAC applies *message passing* to reduce contention latency for sensor-network applications that require store-and-forward processing as data move through the network (Ye, 2002). In addition, the latency, connectivity, energy and memory are the essential elements of today's mobile environments, whose performance may be significantly improved, by caching techniques. The TS-MAC protocol can be used in various topological patterns with varying packet sizes (Andreou, Mavromoustakis, Mastorakis, Bourdena, & Pallis, 2014).

Adaptively, the proposed TS-MAC scheme modifies the duty cycle of a node, adapting the nodes' sleep wake time in order to save power and maximize both the throughput and the network lifetime. Therefore, the proposed TS-MAC scheme manipulates the network state in a distributed form and actively controls each node's sleeping duration, by using specific methods, in order to minimize the energy cost of peer-to-peer communication among mobile terminals. In addition, this protocol provides schemes for both delayed sensitive packets, as well as without deadlines (also known as don't-care packets). Finally, this model could be applied to Ad Hoc networks of various nodes position, number and topology (Mastorakis, Bourdena, Mavromoustakis, Pallis, & Kormentzas, 2013). In this framework, the organization of the chapter is as follows: Section II discusses the related work that has been performed on various MAC protocols, sleeping mechanisms and functions, as well as on implementation techniques. Section III discusses the system model and mechanisms of the S-MAC, which further on introduces the proposed Traffic Sensor Media Access Control (TS-MAC) protocol, followed by Section IV which provides the evaluation and simulation results of the proposed scheme. Finally, Section V concludes with a summary of our contribution and further research.

II. Related Work and Reasearch Motivation

This section provides a brief overview of existing energy conserving MAC protocols in asynchronous and synchronized approaches. Medium Access Control Protocols can be classified in Contention-free, Contention-based and Schedule-based Protocols. This chapter will focus on the Contention-based and Schedule-based protocols, which are used in Ad Hoc Networks. Contention-based protocols are the protocols that focus their attention and workability on the contention/traffic of a network. According to the workload of a network, a node can determine how long it will sleep, if the MAC protocol at hand supports that functionality, and in what way it will act in order to conserve energy, improve throughput and minimize latency. In Schedule-based protocols, the nodes are assigned specific time slots, in order to know when to turn ON or OFF the transmission or the reception of data. P.Karn in (Karn, 1990), introduces MACA, in an effort to solve the problem of the hidden and the exposed terminal. Multiple Access with Collision Avoidance (MACA), makes use of the Ready to Send (RTS) and/or Clear to Send (CTS) messages, in order for nodes to avoid collision of data. Data collision is not ensured to be avoided, as the nodes add in their packet schedule a random amount of time for other nodes to wait before they transmit. If a collision occurs, the nodes wait in a random amount of time, until they retransmit their data. Also, MACA provides poor energy conservation, as nodes do not go to sleep when there is no data for transmission. PAMAS (Singh & Raghavendra, 1998) protocol, which is based on the MACA protocol, introduced the addition of a separate signaling channel while transmitting data packets. In an effort to conserve energy, the PAMAS protocol emphasizes on the node's ability to sleep, while there is no transmission, either sending or receiving data, between itself and a neighboring node. Through the use of a separate signaling channel, nodes can exchange RTS and/or CTS messages without the use of the signaling channel for transmission of data. By applying this, the nodes are able to determine at what specific time and for how long time period they can turn themselves off until the next retrieval or transmission of data. In this case (Yadav, Varma & Malaviya, 2009), the node sensor costs and complexity that arise by the use of two different channels, are enormous as there is excessive energy consumption because of the frequent switching of sleep/wake states of a node.

S-MAC (Joshi, Jaiswal, & Tyagi, 2013), or Sensor-MAC is a simple MAC protocol, inspired by PAMAS protocol, which was designed in an effort to decrease energy consumption of nodes, while trying to send and receive data. The way S-MAC works is, by using a function called Listen-Sleep, integrated with an in-channel signaling methodology. Through this function a node listens for a specific amount of time for any type of RTS, CTS, SYNC or Acknowledgment (ACK) communication method to identify through the network. If the node stumbles upon a RTS or CTS, which is intended for a different node, then it goes back to sleep. When the node is closing by its listen period, it tries to communicate with other nodes for any prospective data packets it may be entitled of. If it finds no data intended for that node, it goes back to sleep. While in Sleep Period, it tries to turn off its radios so that it minimizes the energy consumption and extend its lifetime. The S-MAC uses as well, a Periodic Neighbor Discovery process, which is applied by sending periodic SYNC packets between nodes, in order to determine a nodes' neighboring node, so that they can exchange a synchronization schedule between them, in order to avoid collision. By avoiding collision, the non-corruption of data is ensured. Despite these, the S-MAC trades off energy with throughput and latency, as throughput is decreased because of the single use of the Listen Period of a node and latency is increased because of the extended creation/generation of RTS, CTS and ACK, even while in Sleep mode. T-MAC, or Timeout-MAC (Joshi, Jaiswal, & Tyagi, 2013), is a modified version of the S-MAC protocol, which provides better results in low load conditions, as well as better energy conservation and higher network lifetime. The way the T-MAC functions is that it transmits data packets through bursts of variable length. Between those transmissions, there is a gap called the sleep time/state of a node, and this serves as a function of minimizing idle listening. A frame is created (frame referred to as a complete sleep/wake state), which contains the messages that are about to be transmitted and are currently residing in buffers. The frame is transmitted, when there is an active connection between nodes, and the transmission is over when there are no more frames left to be sent. T-MAC performs poorly in terms of general energy conservation, as it stays idle waiting for data packets that might be transmitted towards it. The amount of energy consumed by staying idle and waiting for data packets is exactly the same with transmitting or receiving data. That means there is a significant energy consumption that minimizes the node lifetime and efficiency. In addition, when neighboring nodes compete for any available medium at the time, data collision and corruption occurs, so packets either get lost or corrupted (Bourdena, Pallis, Kormentzas, & Mastorakis, 2013).

Yang, Tseng, Wu & Chen (2005) propose U-MAC, which stands for “Utilization Based Duty Cycle Tuning” MAC Protocol, based on its predecessor, S-MAC. U-MAC synchronous protocol, emphasizes on lowering the power consumption of a node without increasing latency, by modifying the duty cycles of a node and managing the sleep period after transmission of data packets. Based on U-MAC design, each node does not act according to the same constant duty cycle, but instead, it reconfigures itself according to the last synchronization the node had, and according to the workload it needs to handle. In addition, the node sends its reconfigured schedule to other nodes across the network to its neighboring nodes via broadcasting. All in all, U-MAC can be considered, as a well-thought out protocol that is easily adaptable in different scenarios and can produce desired results. Despite these, although the design is smart and efficient, the overall throughput is not taken under consideration, as well as the data packets transmission between different workloads. Synchronized protocols, such as S-MAC (Wikipedia, 2014), T-MAC (Webopedia, 2010) and U-MAC (Tanenbaum, 2003) can specify when nodes are awake and asleep within a frame. In S-MAC, neighboring nodes form virtual clusters to synchronize the working schedules (Kumar, 2004). B-MAC Protocol (Berkeley-MAC Protocol) was developed in an attempt to decrease energy consumption, providing high avoidance of collisions, as well as efficiently utilizing data transfer channels. As stated by Polastre, Hill and Culler (Polastre, Hill, & Culler, 2004), B-MAC is an asynchronous, reconfigurable MAC protocol that is designed with minimum coding, able to adapt into Wireless Sensor Network environments, providing bidirectional functionalities, towards middleware applications. While B-MAC’s smart and simple Low Power Listening provides good results, in terms of energy efficiency, excess latency (Buettner, Yee, Anderson, & Han, 2006) is present by its long preamble throughout each hop a data packet is sent, thus excess energy consumption is wasted through nodes that act as a part of the bridge of transmitting data to its destination. For that reason, X-MAC (Buettner, Yee, Anderson, & Han, 2006) was devised in an effort to improve the asynchronous B-MAC protocol, by introducing a low power communication technique, which would retain and increase energy conservation, as well as all its other advantages discussed above, by shortening the preamble and abrogating the sleep/wake schedules of the transmitting/receiving nodes. X-MAC introduces a shorter preamble in an effort to increase energy efficiency and decrease latency, by embedding the destination node address information in the preamble so that that non-target nodes will go back to sleep, as soon as they transmit the data packets (overhearing problem). Issues faced with X-MAC was that the timing mechanism contained in the protocol could not create functional sleep schedules for periods of less than 20ms, which in fact would not provide any improvement, regarding the optimal energy-consumption scheme, thus forcing nodes to sleep a lot longer than they should. As a result, latency is increased and throughput is decreased because of these.

WiseMAC (El-Hoiydi, Decotignie, Enz, & Le Roux, 2003) or Wireless Sensor MAC, uses the adaptation of preamble sampling by utilizing the neighboring nodes schedule, thus decreasing the overhearing problem. That is applied by listening to the radio for a specific constant *Tw* period. While that, the transmitter node embeds a wake up preamble in front of its data packets, for the destination node to acknowledge and be awaken when the transmission of data occurs. Because of the nodes independent sampling schedule offsets and in an effort to minimize the overhearing problem, which means nodes expecting for a long time period for a preamble that may contain data transmission towards them without any result because of them not being destined to receive data packets, each node learns the offset of their neighbor’s sampling schedule, so that it can be in synchronization when it comes to transmitting or receiving data. WiseMAC may be efficient for short length multi-hop data transmission, but when it comes to long distance multi-hop data transmission, it fails to accomplish its cause. B-MAC, X-MAC and WiseMAC asynchronous protocols, make use of the adaptive preamble sampling technique in an effort to increase throughput and minimize latency, duty cycles and idle listening. One of the most widely used Ad Hoc Network Protocols is 802.11. The 802.11 protocol (Kumar, Raghavan, & Deng, 2006), which is a CSMA with Collision Avoidance protocol, offers two modes of MAC protocol. The DCF mode, which stands for Distributed Coordination Function, is used in the Ad Hoc Networks and the PCF mode which stands for Point Coordination Function and is used for infrastructure –based networks. The DCF mode uses the messages that are widely used in other protocols, such as the CTS, RTS and ACK. Also, it makes use of the DATA message. 802.11 does not only make use of a Physical Carrier Sensing procedure, but also it introduces the Virtual Carrier Sensing procedure. 802.11 performs well when it comes to low latency and throughput, but performs poorly in terms of energy conservation because of the idle listing problem. The idle listening problem is when a node stays idle without going to a sleep state in order to save energy, but instead listens to the network for any traffic. It has been proven that the energy consumed for idle listening is the same with the energy consumed for transmitting data. Most of the recent works focus on how to prolong a sensor or network battery lifetime. However, these works do not consider the performance of data transmission when sensors have different workload. Our work differs from all of the above-mentioned approaches, by striking the balance between the battery life and transmission latency according to the ongoing traffic situations. The main difference is that the proposed TS-MAC protocol is based on the demand of data transmission of sensors. The use of a traffic-aware protocol can conserve sensors battery life and reduce the transmission latency caused by sleep (Mavromoustakis, Andreou, Mastorakis, Papadakis, Bourdena, & Stratakis, 2014).

In addition, the recent paradigm of Mobile Peer-to-Peer (MP2P) networks can be exploited in future networking architectures. The mobile devices, such as laptops, smart phones, PDAs etc are becoming very popular, supporting the data exchanging like text files, movies, music and applications of voice over IP and gaming. This is why researchers are very interested on MP2P systems and the related architectures that are able to store and retrieve data in a mobile network. The well-known client/server model has been overcome and the main goal of a MP2P architecture is to distribute the load among all mobile nodes (peers) rather than relying on a central powerful server. MP2P systems are a combination of a mobile Ad Hoc network (MANET) underlay with a distributed hash table overlay. MANETs and MP2P networks share concepts of self-organization and decentralization. They do not require a preexisting communication infrastructure for operation but they both operate without a central, coordinating entity. Considering these similarities, combining MANETs and P2P networks, is reasonable to obtain a fully distributed, decentralized and infrastructure-less communication substrate. They operate with a non pre deployed infrastructure and are different from wired ones because each node is able to move around, having a limited radio range. As peers move, the topology of such a system changes constantly. Due to all these, MP2P networks are suitable not only for file sharing but for collaborative activities, social applications, disaster relief, tactical scenarios and development aid projects.

In addition, the evolution of technology, the demand of the modern user of digital devices and the popularity of the new type devices brought together two quite different fields of computer science. The fast and reliable data exchange in conjunction with the growing use and sales of the new generation of mobile devices brought together the mobile web services and the peer to peer networks. In the mobile web services domain big leaps have occurred lately. The efficiency of the mobile devices (i.e. smartphones) is increasing, while the users request even more multimedia services provision. The web services need more software applications day by day, in order to satisfy the increasing number of the users. The skyrocketing of the web services field recently is due to the latest development of the 3G and the 4G technology in combination with the vast number of the wi-fi installations in almost every market. At the same time, the recent mobile phones present several advantages, such as the interconnection between different networks. Some other new features that we can pinpoint, is the increased storage capacity, the faster processor and the longer-lasting batteries. This exponentially exploding situation of the Internet and the new devices has caused also the need of the accurate and easy exchange of even bigger packets of data. Fortunately, at this exact point, the use of the Peer-to-Peer (P2P) networks technology seems to be matching quite perfectly. In normal wired wide networks (e.g. Internet) the P2P automation seems to provide a stable, reliable and robust architecture for exchanging resources. That is exactly the reason why this kind of file sharing systems have over-passed, in content transferring, technologies such as the email and World Wide Web. P2P is a file sharing system, where devices equally share data directly to each other.

Furthermore, much work has been done in the Peer-to-Peer resource sharing for wireless mesh networks, which in addition is compatible with all the network interfaces. A number of researchers have, also, indicated the need of a classification system in order to categorize the Peer-to-Peer applications, due to their evolution. Some more Peer-to-Peer programs have been created to facilitate the resource sharing over the mobile devices. This kind of programs can be, for instance, the hybrid Peer-to-Peer for mobiles, the symtorrent and the WinMobile Torrent. The most recent mobile devices present a number of new capabilities, such as combining several types of networks, either wide or close distant ones. The Peer-to-Peer networking is taking advantage of it and has over-passed the problem of using networks with different characteristics between them, for instance, the stability of the IP address. At this point, we should pinpoint another restriction that came up while trying to install the Peer-to-Peer application on new devices. On the other hand, there are still some disadvantages that could create unwanted signal reduction while transferring information over the air interface. One of the most popular applications in the Peer-to-Peer networking in static computers is the utorrent. The torrent-based technology is one more tool adopted by the mobile devices. The utorrent program, a little bit differentiated, is ready to provide the exact services in new devices. For the creation of this application, the developers use C++ as a programming language. Though, SymTorrent is developed in a Symbian platform, thus we cannot use it in any other operating system. While implementing it, the restrictions were almost zero, as long as the operating system of Sybian is a multithreaded operating system where the developers can easily reach and work with the advanced services of the platform. This, also, means that there are no difficulties or restrictions for hosting applications which use several sockets, file-access and complex user interface. Applying small changes to torrent-based application, other more similar ones can be successfully installed in alternative operating systems except the Sybian. Another example of a torrent-based implementation is the MobTorrent. It is developed in Java ME since the most of the mobile phones support the Java ME. As a result, it gives us the options of downloading and uploading files and thus we can use the Peer-to-Peer evolution. Maybe the latest of this king of applications is the WinMobile Torrent or else the wmTorrent. The wmTorrent is the first torrent client for mobile devices that is running under windows operating systems. The latest version of this torrent application offers advanced features and supports trackerless torrenting, peer exchange (a protocol that augments bitTorrent), Protocol Encryption and HTTP Seeding, which allows data to be exchanged via the hypertext transfer protocol.

The Peer-to-Peer evolution has also made leaps at the field of web services on mobile phones. Online services are much easier to work efficiently on this kind of phones when the wireless signaling rate has been improved that much. This Peer-to-Peer method is used, in order to advertise the online services and the new creation of the JXTA network is responsible to relate distributed components-services to each other, taking under consideration the existed formal interfaces. JTXA, in fact, is a model of Service Oriented Architecture (SOA), which is implemented to support the Peer-to-Peer networking. In details, a Service Oriented Architecture as the JTXA decides, whether a couple of application should communicate over the web. More specific, we refer to applications that provide services over the internet. The most exciting fact about this particular SOA is the simplification of the Mobile Host's entry to a Peer-to-Peer domain by using Advertisements. The implementation of the JTXA advertising consists of language-neutral metadata written in XML form. In the end, except the pairing of the web services through the Peer-to-Peer technology, we also gain access to software, by using primarily the SOAP and secondarily the HTTP, BEEP, UDP for the transmission.

Going further to one more technological prodigy, we find the m-Chord. The m-chord, in fact, is a modified chord in order to manage and support the Peer-to-Peer mobility in all IP networks. Before explaining the way the m-chord works, we should take a brief on the mobile IP and how it works. Although Mobile Nodes switch some times their subnets, the Foreign Agent which belong to other networks still keeps a copy of a temporary Care-of-Address. The Home Agent of the mobile Node is informed about the Care-of-Address and it relates it in a table with the permanent IP of the mobile node. So, the Home Agent tunnels to the Foreign Agents all the packets, which are addressed to Mobile Nodes of his home network. The development of the m-Chord idea, is based on the Peer-to-Peer topology. Mobile Agents are taking advantage of this topology advertising, communicating, exchanging information and in general, using all the features of the peer networking. For instance, m-Chord supports the feature of connecting and disconnecting from the network without affecting the operations, for every Mobile Agent. The researchers of this technology used the Chord networking as a foundation of the new one due to its performance, provable correctness, and the simplicity of the topology. M-Chord uses ring topology of Mobile Agents, agents that could be either Home of Foreign ones. These Agents are capable of hosting more than one Mobile Nodes. Both the agents and the nodes are having a Parent/Child relationship, which is a usual protocol used in hierarchical unstructured Peer-to-Peer networking. M-Chord could be considered as a one-hop wireless transmission technology, where a mobile node has one Mobile Agent and advertises or exchange data in the Chord over the Mobile Agent using just one hop.

In MP2P, it is important to reduce the overhead generated by the lookup algorithms. Each node that is looking for an object compares its own ID with the object ID. The next hop is selected after calculating the matching prefix of those IDs but lookup messages are not always routed on the shortest physical paths. Several architectures have been proposed where each node ID is related with its position. In addition the operation area is splitted in several clusters with their own IDs. The ID of each node in the cluster is defined by this prefix and a random suffix. Clustering level is defined by the depth of the clustering. Furthermore, several researchers propose new routing table methods in order to improve the system performance because existing MP2P methods assume that all peers are on a single layer. These new methods classify peers in two categories. Super peers and sub-peers. Super peers manage the sub-peers that are connected to them and keep information such as id, address, file list of the sub-peers. So each super peer has a routing table with each sub-peer's information. When a sub-peer wants to find a file, it asks its super peer. Super informs the sub-peer where is the file if it is found in its routing table. If file does not exist in super peer's routing table, the super peer informs other super peers with a message in order to locate the file. If any of super peers has the file in its routing table, it informs the requested super peer with a response message. This routing table algorithm prevents multi-broadcasting.

There are also many other studies that propose a double-layered P2P system in which super-nodes are selected based on their mobility pattern. Node energy level is taken in consideration along with mobility factor in order to enhance the system stability and reliability. Super-peer topology construction and maintenance scheme based on network coordinates is also a proposition. On the other hand RobP2P, which is a new proposition that integrates many factors to efficiently select super peers, including the node's current mobility, immediate energy level, network capability, mean uptime, and connectivity degree. There are three main objectives for RobP2P: 1) develop a robust and efficient super-peer selection protocol; 2) reduce the overhead traffic of network topology maintenance; 3) increase the reliability and stability of the network infrastructure through enabling peers to flexibly change their role. Several regions are created for the P2P network. Each group selects (according to an algorithm) a super peer that represents the group head, while the rest of the peers become ordinary peers. From this point, ordinary peer's messages (hello, advertise and query) are sent to the respective super peer. Super peers collect and index the group information including active peers, advertised resources and offered services in order to manage the group communications and resolve queries addressed to the group. The super peer is also responsible of maintaining the group state including selecting new super peers, in case its context changes or moves away. MP2P networks have often been considered as the "new-comer" of the computer communication family, while the client/server file-share scheme has been the "incumbent." In MP2P nodes span a wide range of mobile device form factors and wireless sensors with different features and variety hardware and software stacks. Some of these peers are high-end mobile devices such as laptops, tablets, smartphones, and vehicles equipped with processing power and network connectivity. Other peers have limited processing power, battery life, storage, and network capabilities, such as featured phones, low-end wireless sensors, cameras, etc. We assume that peers form P2P groups based on location proximity, within a few hops from one another, regardless of their interests and resources they may share. Peers are on the move and their context changes dynamically. This technology will enable a next generation of innovative killer applications, collaboration technologies will promote the creation of worldwide user communities and will permit low cost sophisticated content publishing and distribution tools (net radio, net tv)

MP2P should provide a continuously connection – disconnection and direct communication with other mobile nodes at broadband speeds based on their physical proximity, in a low level overhead traffic. The selection of super peers is the most critical point. Randomly selection of super peers is not a solution so, in order to select super peers we have to take in consideration many parameters, like the distance cost, processing power, content similarity and high capacity. Periodically peers build super peer candidates which are responsible for answering the queries that come from simple peers. If a super peer cannot answer a simple's peer query (file request), it communicates with other super peers in order to satisfy simple's peer query. Finally it is critical for nodes to have the ability to change easily their role from simple peer to super peer and vice versa.

III. Proposed **T**raffic based **S-MAC** Protocol (ts-mac)

System model and mechanisms of S-MAC

Power control in wireless devices aims to determine the transmit periods and the associated power level, such that the energy consumed is steadily reduced, whereas at the same time it aims to guarantee the resource sharing stability. Since nodes in wireless networks typically rely on their battery energy, this work, proposes a mechanism which hosts a traffic-aware scheme for conserving energy in wireless environments using a Clustered-based mechanism. This mechanism is encapsulated in a middleware as in (Mavromoustakis, Dimitriou, Mastorakis & Pallis, 2013) and encompasses a scheme, which evaluates the scheduled activity periods of each node, in order to measure the scheduled time that each node can safely sleep. The proposed framework uses a quantitative model based on the supervision of the Sleep-Proxy (a role-based node) which actively, in a clustered-environment, takes into consideration the cooperative caching and the resource exchange scheme for enabling Energy Conservation in the Cluster.

Nodes in wireless networks typically rely their survivability on their battery energy. Energy consumption should be minimized in order to save power in a decentralized form. The effects on reducing each node's power consumption has been studied thoroughly (Mavromoustakis, Dimitriou, Mastorakis & Pallis, 2013). However an association of the capacity and caching characteristics and measures with respect to efficient message passing for self-tuning energy consumption is explored in this work. The energy conservation mechanism has to be closely collaborative with the upper layer protocols used, to maintain the packet forwarding mechanism, in an error free mode. A best candidate solution must be cost-effective and adaptive, and should be able to keep abreast of any possible changes in the network, in terms of load and failures. The above issues combined with the unpredictable movements of the users, and with each device's technical characteristics (like link asymmetry) should be balanced. This work proposes a way to avoid any technical discrepancies that may exist between different devices, while it balances energy consumption on each device or sudden energy deficiency. In comparison with the CSMA/CA protocol, S-MAC is more energy efficient as it conserves more energy due to the use of duty cycles. Although this is true for low load networks, it is not as efficient with heavy load networks, where sleeping schedules are not applied between nodes due to the constant transmission of data. S-MAC synchronization mechanism is designed in a way that forms virtual clusters in Ad-Hoc networks and utilizes energy conservation, by avoiding overhearing and transmitting packets and long messages in an efficient manner (Mavromoustakis & Karatza, 2004). Although S-MAC synchronization mechanism utilizes a sleep-wake schedule in an effort to conserve energy, it does not provide flexibility when it comes to bursty traffic because of the fixed duty cycles operation and fixed sleep periods of nodes. In a low load network, while each node sleeps through a specific period of time, it has to wake up at the beginning of each frame, in order to receive the SYNC packet, which will determine where and what to send on its next wake period. An important and constant amount of energy is consumed in this procedure because of this fixed duty cycle structure. Nonetheless, if faced with a heavy load network, the listen period may prove to be too short, in order to accommodate traffic, leading to unnecessary delays between packet transmissions amongst Sender and Destination nodes. The sleep time duration can be measured using the following expression (Mavromoustakis & Karatza, 2004):

$$S(t_{new})_D = S(t)_{D(\tau-1)} - S(t)_{D(\tau-1)} \cdot \left[\frac{Packets\,Cached}{Total\ Packets\,destined\ for\ D} \right]$$

While S-MAC uses virtual clusters where all nodes sleep at the same time and transmit data during the listen period in an effort to conserve energy, it actually trades off latency. Although idle listening duration is reduced, it also increases the end-to-end latency amongst nodes and data packets. This makes S-MAC unreliable when it comes to delay-sensitive data. Furthermore, increased collision of data packets has been noticed in heavy load networks, constituting S-MAC protocol as inappropriate for this type of networks, where heavy congestion and high loads are present. In order to enhance the already existing S-MAC protocol for a Wireless network, to save energy and transmit data packets, the scheduling mechanism had to be modified and applied in various topologies, with data packets of variable size. The results of these modifications are shown in part IV, of the experiments. The scheduling mechanism is based on preset duty cycle of each node, using the S-MAC protocol. By changing the duty cycle of each node, the purpose of modifying its sleep and wake state, either by increasing their sleep state or decreasing, is achieved, thus providing results based on different types of topologies that these modifications were tested. The main attempt is to achieve greater and more productive results in low and high density networks, by increasing or decreasing the sleep wake states, in order to provide an enhanced version of the S-MAC protocol.

IV. Performance Evaluation Analysis, Experimental Results and Discussion

In this section, four different scenarios are examined, were the duty cycle of all nodes was modified in order to extract results, in comparison with the proposed TS-MAC protocol. For this implementation Network Simulator 2 (NS2) was used, a tool that helps simulating different network scenarios and providing the flexibility of selecting protocols, node quantities, topologies and much more. The first scenario consists of 3 nodes in a random order, sending 512bytes data packets as shown in Figure 1.

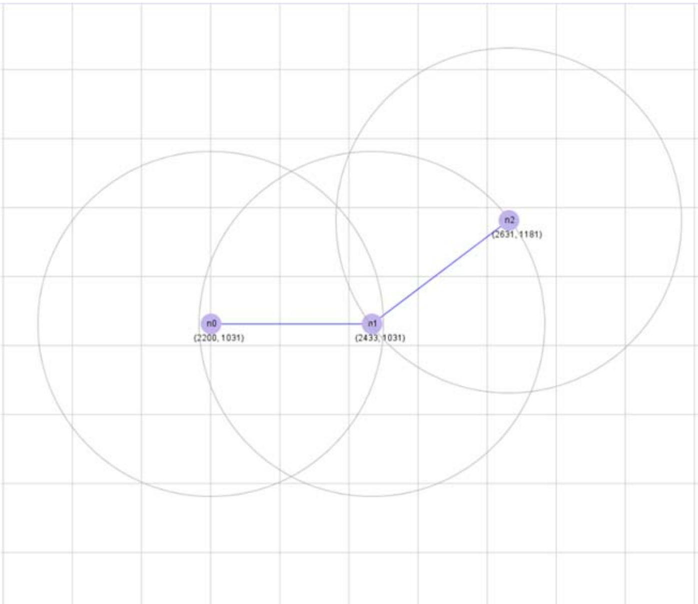

Figure 1. 3 Nodes in Random Order Topology with 512bytes data packets (Scenario 1)

As seen through all Scenarios, the transmission range of each node is visible, as it covers the neighboring node were the transmission of data packets is feasible. In the second scenario (Figure 2), 5 nodes send 256bytes data packets, again in a random order.

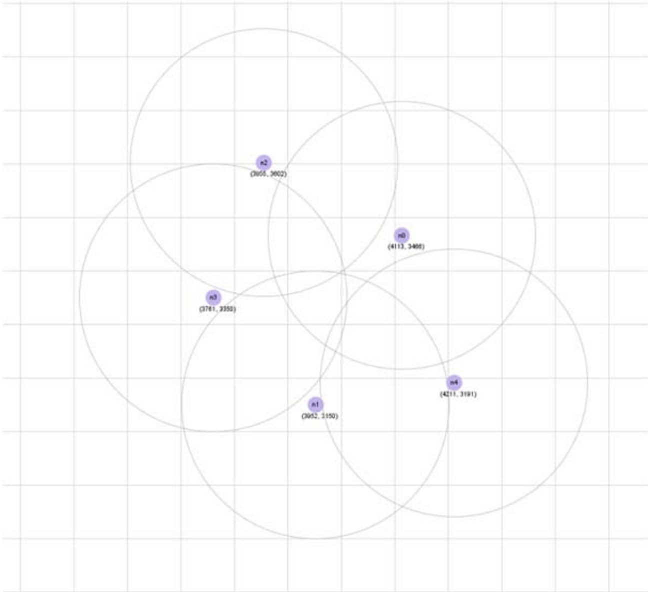

Figure 2. 5 Nodes in Random Order topology with 256bytes data packets (Scenario 2)

For the third scenario, 7 nodes have been placed in a chain order, sending 512bytes of data packets as shown in Figure 3.

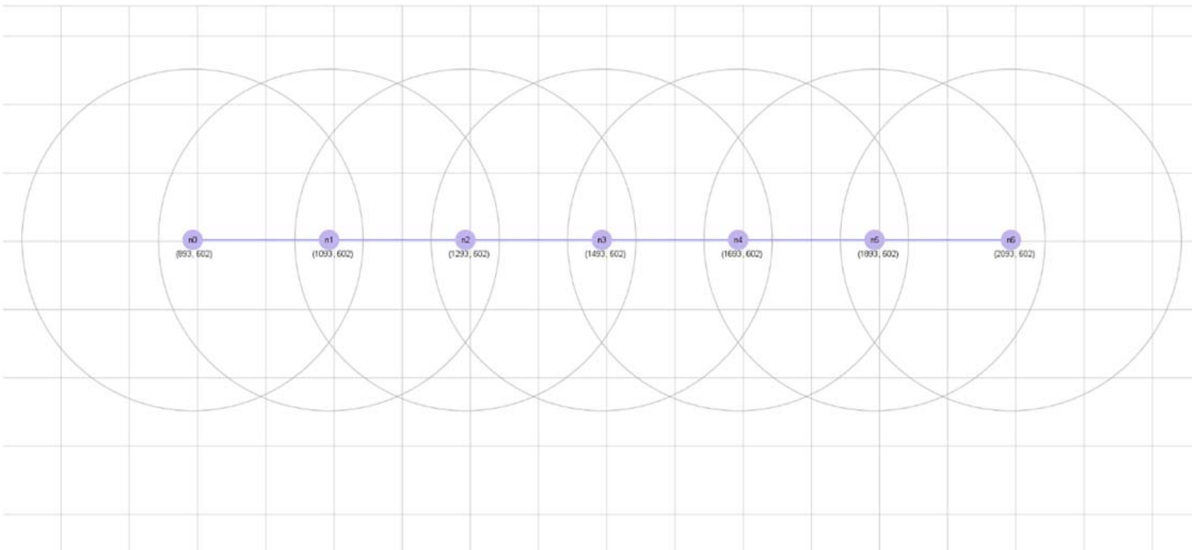

Figure 3. 7 Nodes in Chain Order topology with 512bytes data packets (Scenario 3)

For the last scenario (Figure 4), 16 nodes have been used in a grid order, sending 512bytes of data packets.

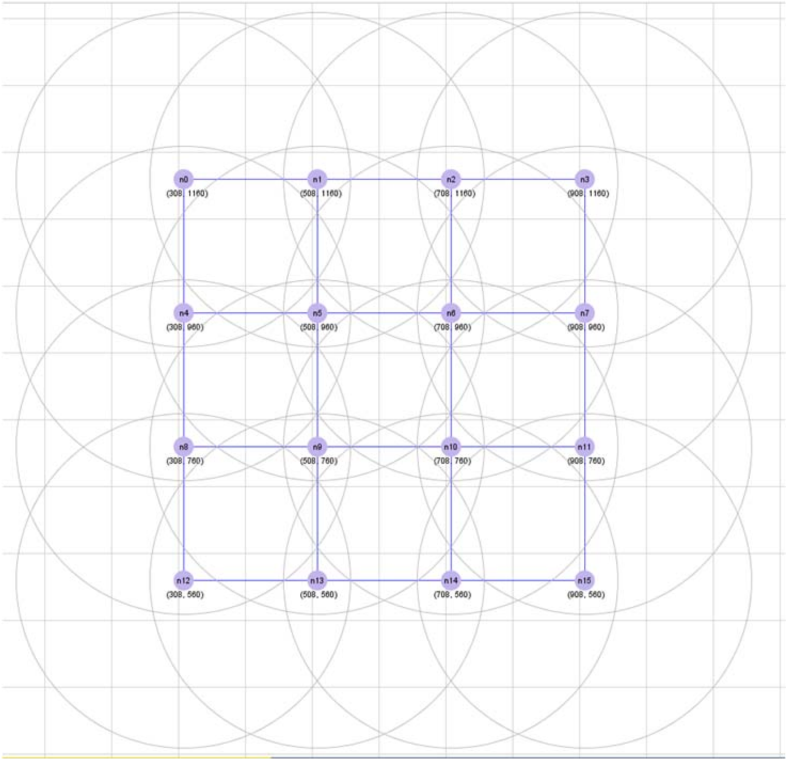

Figure 4. 16 Nodes in Grid Order topology with 512bytes data packets (Scenario 4)

For a more detailed description of the scenarios discussed above, the following table (Table 1) provides all the necessary details accordingly.

TABLE I. DETAILED EXPLANATION OF SCENARIOS

Scenario	**1**	**2**	**3**	**4**
Nodes (Number)	3	5	7	16
Topology	800 x 800	16321 x 100	14306 x 932	3295 x 959
Order of Nodes	Random	Random	Chain	Grid
Data Packets (Bytes per packet)	512bytes	265bytes	512bytes	512bytes
Simulation Time (seconds)	35000sec	35000sec	35000sec	35000sec
Maximum Packets (quantity)	50000	50000	50000	50000

In each scenario, various changes of the Duty Cycle have been implemented ranging from 0% to 100%, incrementing by 10% at a time. For example, a script with 10% duty cycle was compiled, another one with 20%, 30% and so forth. By applying these modifications to the scenarios, one at a time, 40 different samples for the TS-MAC protocol have been created, and another 4 samples using the standard S-MAC protocol for comparison reasons, from which results of the graphs were extracted and analyzed regarding the following:

A) Average Throughput
B) Average Instant Jitter
C) Average End-to-End Delay between Sent and Received Packets in seconds
D) Average End-to-End Delay between Sent and Dropped Packets in seconds
E) Packet Delivery Ratio for Sent, Received and Dropped packets
F) Received over Sent Packets Ratio
G) Energy Consumption (Joules)

More specifically, the Throughput has been obtained below over specific Duty Cycles, through all 4 scenarios. The results have compared between the SMAC protocol and the proposed TS-MAC Protocol. It should be noted that based on the following findings, the differences between the standard S-MAC protocol that has a fixed duty cycle operation, are examined and all the variations of the duty cycles of the TS-MAC protocol are investigated, in order to distinguish which duty cycle of the TS-MAC is most effective, thus using it as the default duty cycle of the proposed TS-MAC protocol. In scenario 1, the Average Throughput (measured in bytes) extracted by the TS-MAC protocol is much higher than the Average Throughput of the S-MAC protocol, as shown in Figure 5. When the Duty cycle of the TS-MAC protocol is modified to 10 duty cycles per sleep state, a significant decrease is observed regarding the Average throughput of Packets. Nonetheless, when the duty cycle is modified to TS-MAC/20DC, there is a significant improvement, much higher than the extracted Throughput of packets by the S-MAC protocol. This improvement is noticed between TS-MAC/20DC, TS-MAC/50DC, as well as TS-MAC/90DC per sleep state.

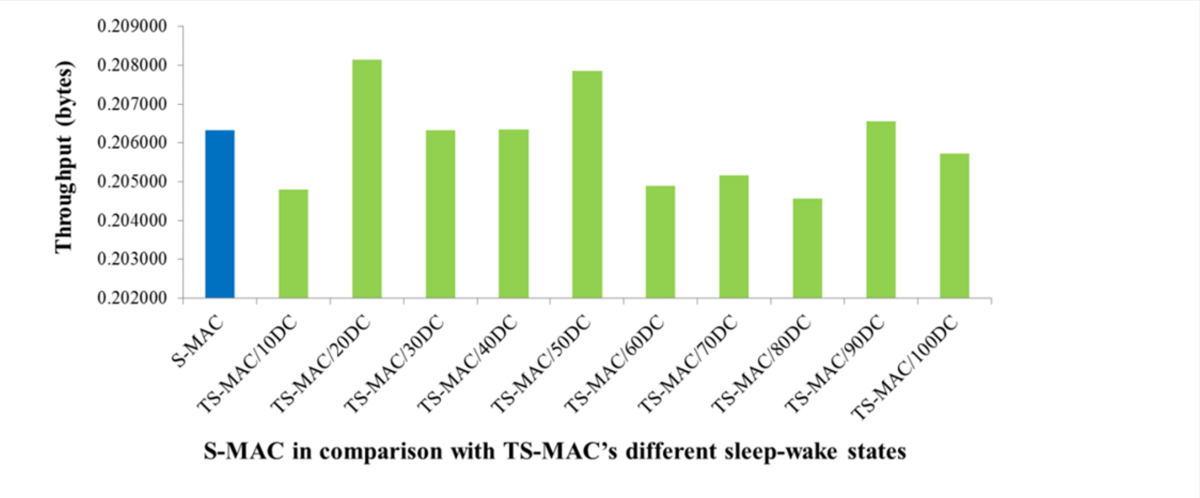

Figure 5. Average Throughput (Scenario 1)

In scenario 2 (Figure 6) of the Figure below, the same principles apply, but in this case only the TS-MAC/100DC exceeds the Average Throughput of the S-MAC protocol. It can be observed that there is a greater number of nodes (5 in comparison with 3 in Scenario 1), yet in a Random order as before. Nonetheless, the Data Packets size has been significantly decreased for this Scenario and the possibility of the number of the nodes affecting the Average Throughput, should be taken under consideration, as through this experiment there is a noticeable difference regarding the Average Throughput between the two protocols.

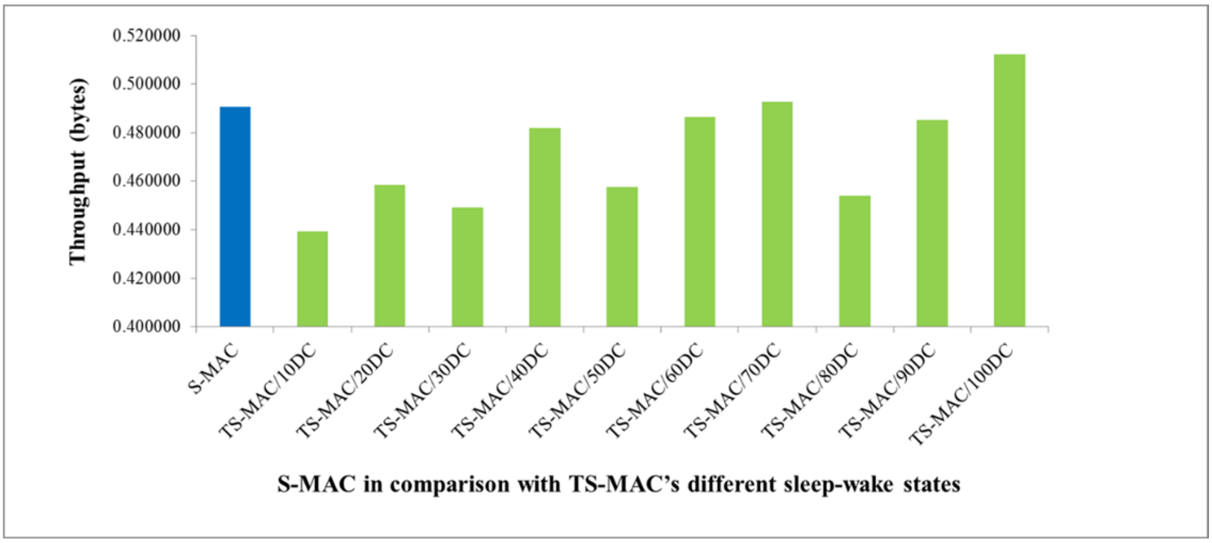

Figure 6. Average Throughput (Scenario 2)

In scenario 3, a different topology has been implemented. In this case, the nodes are 7, placed in a Chain order as seen in the previous page, with 512 bytes of Data Packets and the Simulation lasted for 35000 seconds. In this Scenario, there is no significant change to the Average Throughput regarding the TS-MAC, as the S-MAC exceeds its rival regarding the Average Throughput as seen in Figure 7. Nonetheless, this is not a drawback, as there are also other factors that need to be taken under consideration, in order to evaluate and improve a proposed network transfer protocol, as discussed further on.

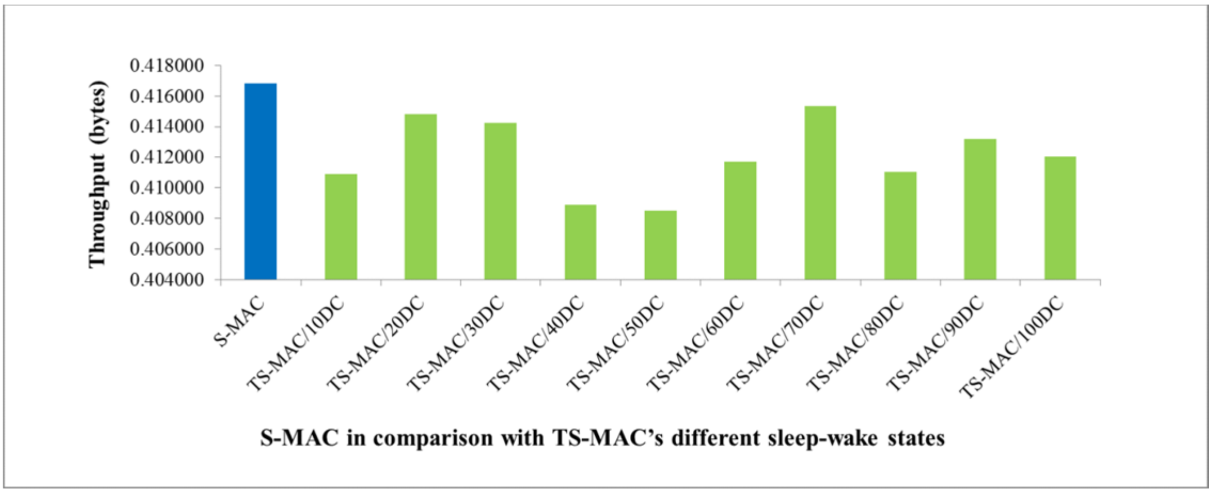

Figure 7. Average Throughput (Scenario 3)

In scenario 4, in Figure 8, 16 nodes are introduced in the topology, in a Grid order with the same specifications as used in Scenario 1 and 3, with different dimensions of the topology. By taking a closer look, and studying the extracted results, the TS-MAC exceeds the S-MAC in many aspects. Also the duty cycles that do not exceed the S-MAC protocol are decreased from 10 in Scenario 3, to 6 in Scenario 4, which is considered as a 40% increased throughput. It is of significant value, that it is kept in mind the purpose of these Scenarios, which is to extract the most efficient duty cycle of TS-MAC, in comparison with S-MAC.

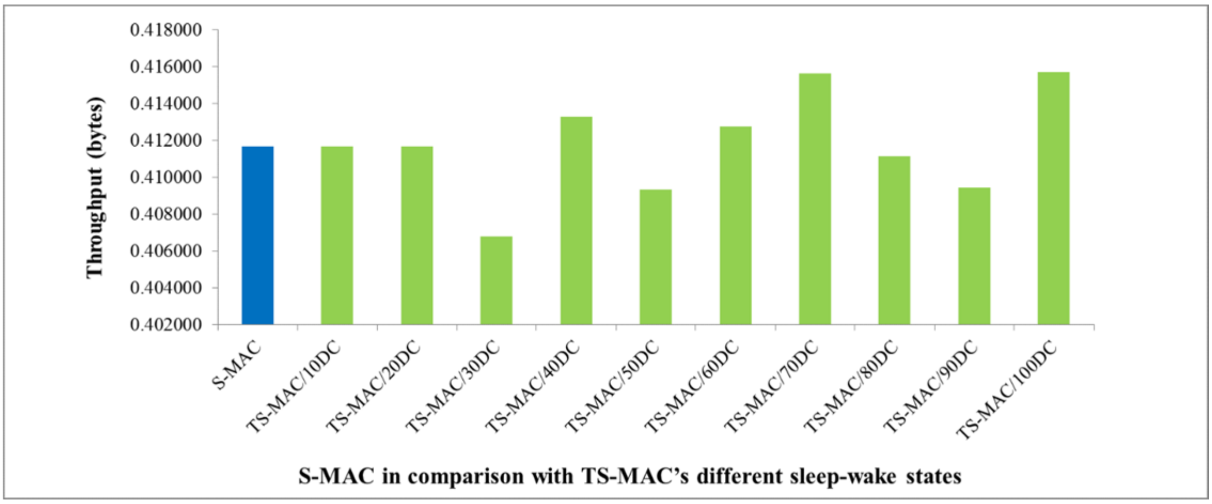

Figure 8. Average Throughput (Scenario 4)

Throughout these, the conclusion is made that scenario 4 has the best application of the proposed TS-MAC. In comparison with the other three scenarios, TS-MAC has achieved greater throughput than S-MAC, throughout different sleep-wake states. In scenario 4, the vast majority of these different duty cycles of TS-MAC have produced greater results, than any other scenarios. Nonetheless, scenario 4 will not only be used in the rest of the experiments, but also other scenarios will be included as well, in order to show that positive results are extracted throughout any type of topology or number of nodes. TS-MAC/70DC exploits the default duty cycle scheduling mechanism for TS-MAC, in order to experiment with the compiled scenarios and use the required findings for the further development of the TS-MAC. In addition, the Average Instant Jitter between TS-MAC and S-MAC protocols has been extracted. Different Scenarios will be examined for the rest of the experiments, in order to notice the differences between the applied protocols throughout the 4 scenarios.

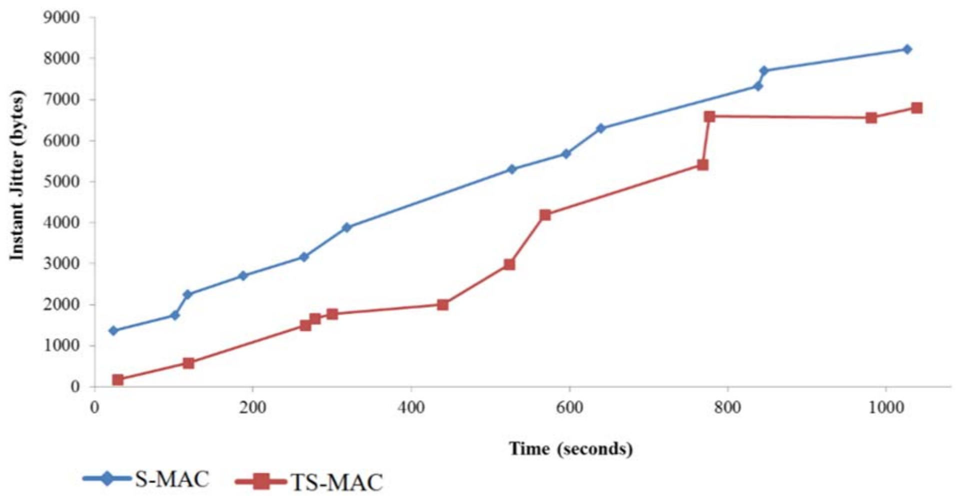

Figure 9. Average Instant Jitter (Scenario 4)

In scenario 4 in Figure 9, it is clearly shown than depending on the time, the Average Instant Jitter (measured in bytes) of the S-MAC, mostly exceeds the one of the TS-MAC. In this case, it is not desirable for the instant jitter of TS-MAC to exceed S-MAC as this plays a determining role in the energy consumption of the nodes. There are also, other duty cycles that produce less jitter than the S-MAC protocol, thus providing less energy consumption for the nodes. In this specific scenario, the 30, 40, 50, 70 (used as the default duty cycle), 90 and

100DC produce the desirable outcome. Furthermore, the end to end delay between the nodes has been examined, as well as the Average Packet Duration (measured in seconds) between the Sent and Received Packets (Figure 10) has been calculated. It is quite interesting that all 4 scenarios produced the same results. There is no specific delay between the Sent and Received packets between nodes, both in TS-MAC and S-MAC protocols.

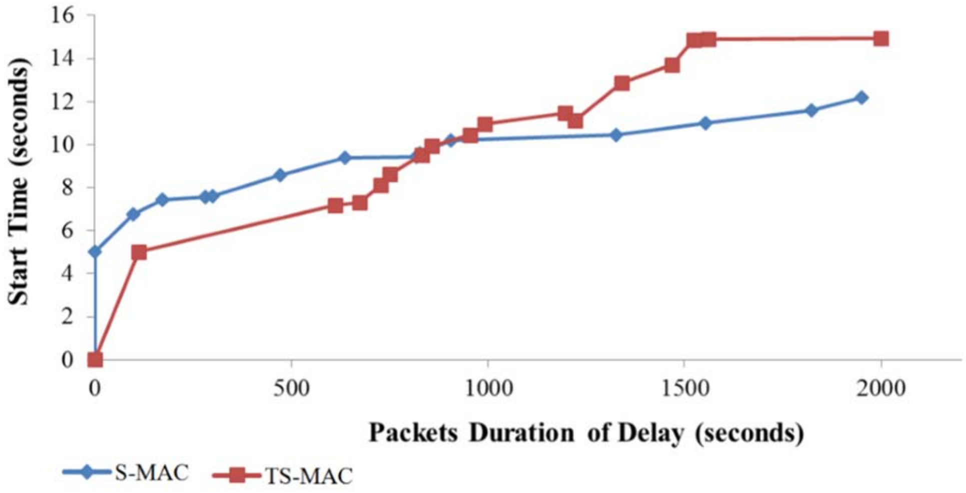

Figure 10. End to End delay_Average Packet Duration_Sent-Dropped (in seconds) (Scenario 2)

In addition, the end to end delay between the Sent and the Dropped packets has been examined as well. The outcome was defined between the Default Duty Cycles of S-MAC and the TS-MAC protocol, as there was a difference of 0,11 seconds for the Average Packet Duration between the Sent and Dropped Packets. Nonetheless, this is also applicable for all 4 Scenarios, thus a very important factor is signified, in order to improve the proposed TS-MAC protocol: by taking under consideration the effectiveness of the S-MAC protocol regarding the end to end delay between Sent and Dropped Packets, implementing and modifying accordingly in order to provide greater effectiveness. Further on, the Packet Delivery Ratio (measured in packets) for all Sent, Received and Dropped Packets has been examined. Below there are some extensive results for scenario 2, portraying the number of packets Sent (Figure 11), Received (Figure 12) and Dropped (Figure 13) between S-MAC and TS-MAC protocol.

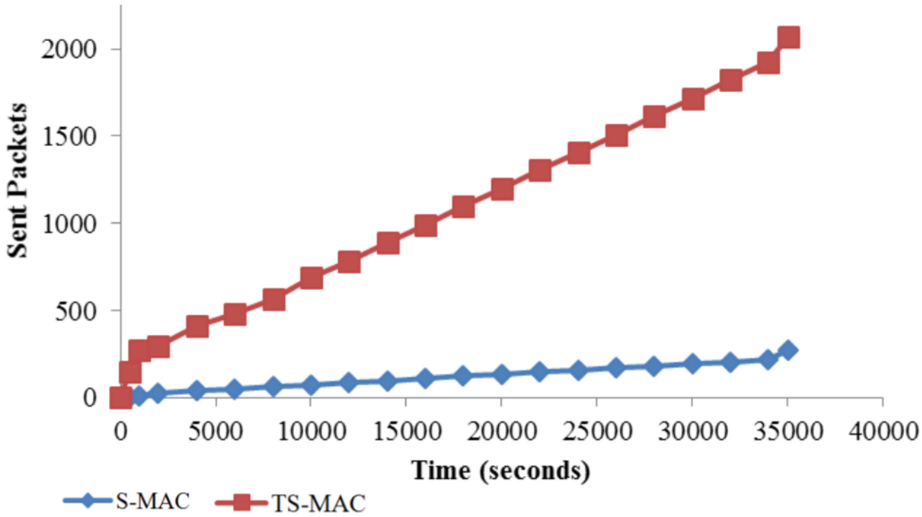

Figure 11. Packet delivery Ratio for Sent Packets (in packets) (Scenario 2)

In Figure 11, the TS-MAC was captured sending a larger number of packets in comparison with S-MAC, within the same time frame. As the sent packets were increased in order to extract the results for the modified proposed protocol, a vast difference between the numbers of packets sent between the two protocols is visible. The received packets increased (Figure 12) in an orderly manner as time progressed. Despite the increased number of Sent packages of TS-MAC, it should be mentioned that the Default S-MAC protocol received more packets than its rival, in comparison with the sent and the received packets.

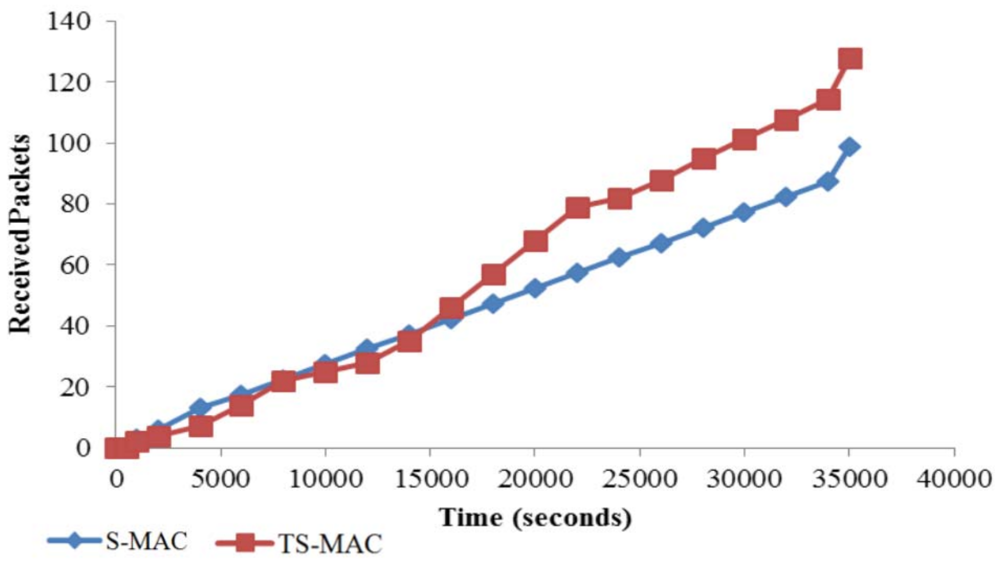

Figure 12. Packet delivery Ratio for Received Packets (in packets) (Scenario 2)

While the time frame progressed, so did the dropped packets of TS-MAC (Figure 13), thus surpassing the received packets. This is a very important pointer as the potential of corrupted packets between nodes makes the protocol not as efficient and productive as it is intended to.

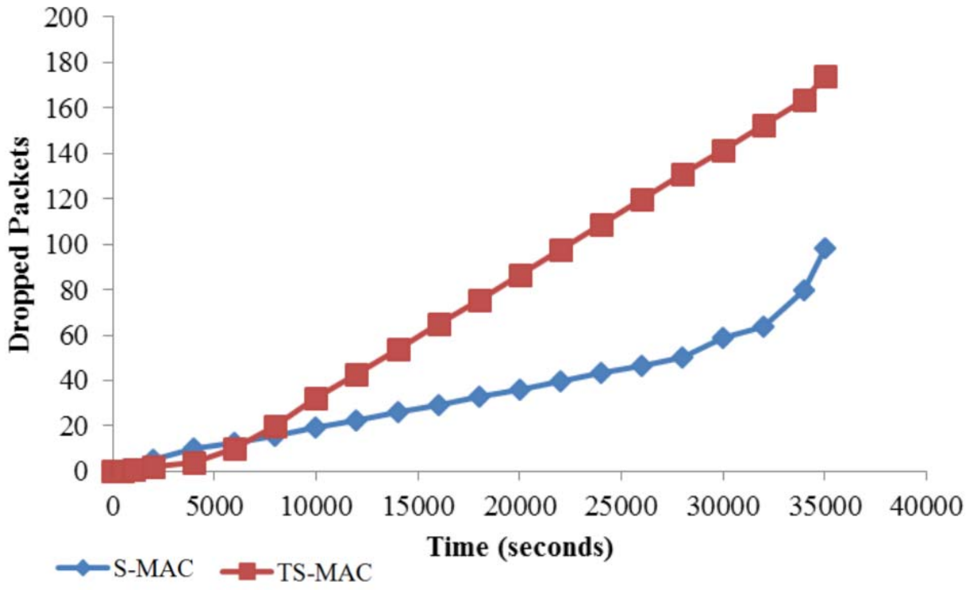

Figure 13. Packet delivery Ratio for Dropped Packets (in packets) (Scenario 2)

Moreover, in the following graph below (Figure 14), the Received over the Sent Packets Ratio (measured in packets) for the Packet Delivery Ratio (Average) is examined. Once again, it is very important not to overlook the differences between the TS-MAC and the S-MAC protocols. The TS-MAC protocol surpasses the S-MAC protocol regarding the Received/Sent Ratio as it exceeds the values in all attempts to apply the proposed TS-MAC.

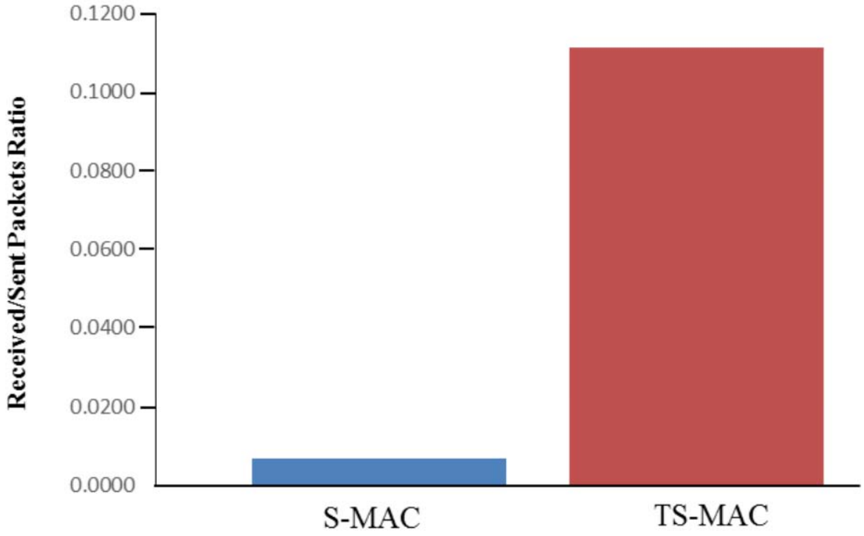

Figure 14. Packet delivery Ratio – Received over Sent Packets Ratio (Scenario 2)

The final experiment, for testing and extraction was implemented for the Energy Consumption for both protocols. In Figure 15, scenario 4, the TS-MAC protocol consumed less energy, than the S-MAC, which shows that the nodes lasted a lot longer throughout the experiment. The TS-MAC utilized the nodes energy, 2.5 times more than the S-MAC, which implies that power consumption is handled in a better manner by the proposed protocol.

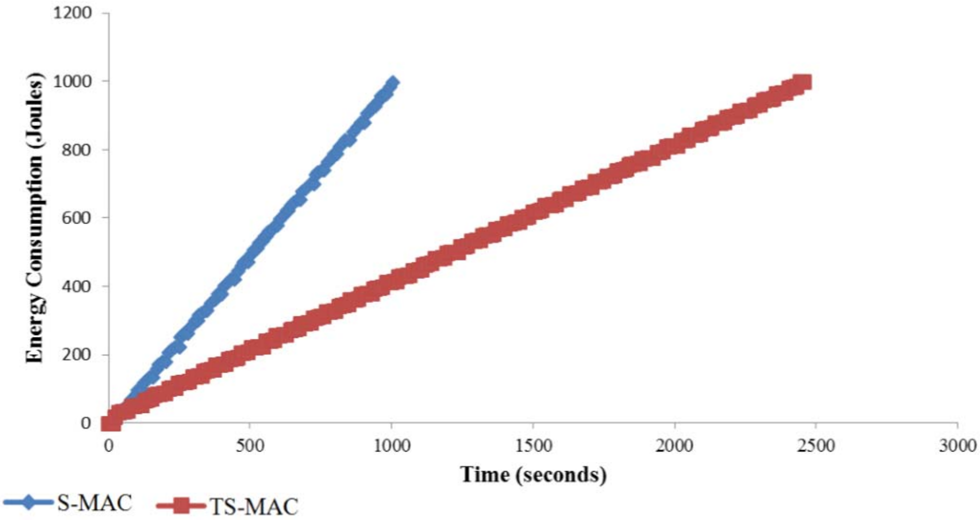

Figure 15. Energy Consumption in Joules (Scenario 4)

Based on the graphs and the extracted results, the modification that has been applied to all the scenarios, was alternating throughout the experiments. There were variations, according to the scheme of each scenario. For instance, scenario 2 had the highest throughput than any other, in comparison for the 100DC, but in scenario 3 and 1, it had an in between value, in comparison with the other examples. Also, the overall throughput for each scenario depicts which of the duty cycles were proven most efficient. Regardless, it should be noted that an important number of advantages is applicable, when it comes to the TS-MAC protocol, as well as some important thought-provoking matters, such as transmission of packets that will help, in improving a protocol, which is both efficient and productive for its use. Overall, the TS-MAC has been proven to work best for large data packets transmission, as well as in larger topologies with a vast number of nodes, handling energy consumption in a more efficient way than the rival protocol in comparison (scenario 4).

V. Conclusions

This work proposes a reflective scheduling mechanism, by taking into consideration the differentiation of the wake sleep duty cycle of each node. Through the modification of the duty cycles, it has been derived that it can produce both positive and negative outcomes, as some factors can differentiate the results, such as topology, density, number of nodes, data packets size and the duty cycle of each node. If this mechanism is developed in more depth, it will produce good results, by minimizing even more the energy consumption and increasing the throughput, decreasing dropped packets, by modifying the sleep wake state of each node. The development of the protocol must focus on the factors mentioned above, such as the quantity of the sent and the dropped packages, as well as the end to end delay between them. If these important factors, mentioned throughout this research, are taken under serious consideration, the development and implementation of the TS-MAC may prove to be a success, producing desired results and increase the efficiency and the energy of the nodes, thus making an Ad-hoc Network effective towards the current principles of Network Performance and Design. In this direction, future work includes the expansion of the utilization of the traffic to be mapped onto traffic patterns. These patterns will include traffic engineering models (Bourdena et al., 2014), in order to map and figure-out the behavior of such dynamically changing scenarios (Dimitriou, Mavromoustakis, Mastorakis & Pallis, 2013). Furthermore, this work is planned to be expanded into a framework (Mavromoustakis, Pallis & Mastorakis, 2014) that uses the combined infinitesimal perturbation analysis so that stochastic algorithms may depict the performance gradient of the system. The above issues comprise of open-end research dimensions with many research concepts for future examination (Mavromoustakis, Mastorakis, Bourdena & Pallis, 2014).

Acknowledgment

The work presented in this chapter is co-funded by the European Union, Eurostars Programme, under the project 8111, DELTA "Network-Aware Delivery Clouds for User Centric Media Events".

References

Andreou, A., Mavromoustakis, C. X., Mastorakis, G., Bourdena, A., & Pallis, E. (2014). Adaptive Heuristic-Based P2P Network Connectivity and Configuration for Resource Availability. In Resource Management in Mobile Computing Environments (pp. 221-240). Springer International Publishing.

Bourdena, A., Mavromoustakis, C. X., Kormentzas, G., Pallis, E., Mastorakis, G., & Yassein, M. B. (2014). A resource intensive traffic-aware scheme using energy-aware routing in cognitive radio networks. Future Generation Computer Systems.

Bourdena, A., Pallis, E., Kormentzas, G., & Mastorakis, G. (2013). Efficient radio resource management algorithms in opportunistic cognitive radio networks. Transactions on Emerging Telecommunications Technologies.

Buettner, M., Yee, G.V., Anderson, E., & Han, R. (2006). X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks. *4th international conference on Embedded networked sensor systems (SenSys '06). ACM, New York, NY, USA,* (pp.307-320).

Chakrabarti, S., & Mishra, A. (2001). QoS issues in Ad Hoc wireless networks. *IEEE Communication Magazine 39 (2),* 142–148.

Dimitriou, C. D., Mavromoustakis, C. X., Mastorakis, G., & Pallis, E. (2013, June). On the performance response of delay-bounded energy-aware bandwidth allocation scheme in wireless networks. In 2013 IEEE International Conference on Communications (ICC 2013), Workshops, pp. 631-636.

El-Hoiydi, A., Decotignie, J.D., Enz, C., & Le Roux, E. (2003). Poster abstract: wiseMAC, an ultra-low power MAC protocol for the wiseNET wireless sensor network. *1st international conference on Embedded networked sensor systems (SenSys '03). ACM, New York, NY, USA, (*pp.302-303).

Haas, Z.J., & Tabrizi, S. (1998). On some challenges and design choices in Ad Hoc communications", *IEEE MILCOM98*, *Vol. 1.*

Joshi, S., Jaiswal, A. Kr., & Tyagi P. Kr. (2013). A Novel Analysis of T Mac and S Mac Protocol for Wireless Sensor Networks Using Castalia. *International Journal of Soft Computing and Engineering (IJSCE). ISSN: 2231-2307, Volume-2, Issue-6.*

Karn, P. (1990, September). *MACA-a new channel access method for packet radio*. Paper presented at ARRL/CRRL Amateur Radio 9th Computer Networking Conference.
Kumar, S. (2004). Medium Access Control protocols for Ad Hoc wireless networks: A survey. (pp. 326 – 329).

Kumar, S., Raghavan, V.S., & Deng, J. (2006). Medium Access Control protocols for Ad Hoc wireless networks: A survey. *Vol. 4, Issue 3,* 326–358.

Mastorakis, G., Bourdena, A., Mavromoustakis, C. X., Pallis, E., & Kormentzas, G. (2013, July). An energy-efficient routing protocol for ad-hoc cognitive radio networks. In Future Network and Mobile Summit (IEEE FutureNetworkSummit), 2013 (pp. 1-10).

Mastorakis, G., Mavromoustakis, C. X., Bourdena, A., Kormentzas, G., & Pallis, E. (2013, September). Maximizing energy conservation in a centralized cognitive radio network architecture. In 2013 IEEE 18th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (IEEE CAMAD 2013), (pp. 175-179).

Mavromoustakis, C., Andreou, A., Mastorakis, G., Papadakis, S., Bourdena, A., & Stratakis, D. (2014, August). Energy Consumption Optimization through Pre-scheduled Opportunistic Offloading in Wireless Devices. In EMERGING 2014, The Sixth International Conference on Emerging Network Intelligence (pp. 22-28).

Mavromoustakis, C. X., Bourdena, A., Mastorakis, G., Pallis, E., & Kormentzas, G. (2014). An energy-aware scheme for efficient spectrum utilization in a 5G mobile cognitive radio network architecture. Telecommunication Systems, 1-13.

Mavromoustakis, C., Dimitriou, C. D., & Mastorakis, G. (2013). On the Real-Time Evaluation of Two-Level BTD Scheme for Energy Conservation in the Presence of Delay Sensitive Transmissions and Intermittent Connectivity in Wireless Devices. International Journal On Advances in Networks and Services, 6 (3 and 4), 148-162.

Mavromoustakis, C. X., Dimitriou, C., Mastorakis, G., Pallis, E. (2013). Real-Time Performance Evaluation of F-BTD scheme for optimized QoS Energy Conservation in Wireless Devices. Proceedings of IEEE GlobeCom 2013, (pp. 1156-1161).

Mavromoustakis, C. X. & Karatza, H. D. (2004). Adaptive Energy Conservation Model using Dynamic Caching for Wireless Devices. *IEEE,* pp. 4.

Mavromoustakis, C. X., Mastorakis, G., Bourdena, A., & Pallis, E. (2014). Energy efficient resource sharing using a traffic-oriented routing scheme for cognitive radio networks. IET Networks.

Mavromoustakis, C. X., Mousicou, P., Papanikolaou, K., Mastorakis, G., Bourdena, A., & Pallis, E. (2015). Dynamic Cloud Resource Migration for Efficient 3D Video Processing in Mobile Computing Environments. In Novel 3D Media Technologies (pp. 119-134). Springer New York.

Mavromoustakis, C. X., Pallis, E., & Mastorakis, G. (2014). Resource Management in Mobile Computing Environments. Modeling and Optimization in Science and Technologies. Springer

International Publishing.

Mousicou, P., Mavromoustakis, C. X., Bourdena, A., Mastorakis, G., & Pallis, E. (2013, November). Performance evaluation of dynamic cloud resource migration based on temporal and capacity-aware policy for efficient resource sharing. In Proceedings of the 2nd ACM workshop on High performance mobile opportunistic systems (pp. 59-66).

Polastre, J., Hill, J., & Culler, D. (2004). Versatile low power media access for wireless sensor networks. *2nd international conference on Embedded networked sensor systems (SenSys '04), ACM, New York, NY, USA,* (pp.95-107).

Royer, E.M., & Toh , C.K. (1999). A review of current routing protocols for Ad Hoc mobile wireless networks. *IEEE Personal Commun. 6 (2),* 46–55.

Singh, S. & Raghavendra, C.S. (1998). PAMAS-power aware multi-access protocol with signalling for Ad Hoc Networks. *SIGCOMM Comput. Commun. Rev. 28, 3*, 5-26.

Tanenbaum, A.S. (2003). *Computer Networks, 4th ed.*, Pearson Education Inc., ISBN 0-130-66102-3.

Toh, C.-K. (2002). *Ad Hoc Mobile Wireless Networks: Protocols and Systems.* Prentice-Hall PTR, NJ.

Webopedia. The 7 Layers of the OSI Model. Retrived September 24, 2010, from http://www.webopedia.com/quick_ref/OSI_Layers.asp

Wikipedia. Network Layer. Retrived February 18, 2014, from http://en.wikipedia.org/wiki/Network_layer

Yadav, R., Varma S., & Malaviya, N. (2009). A survey of MAC protocols for Wireless Sensor Networks. *UbiCC Journal, Volume 4, Number 3.*

Yang, S., Tseng, H., Wu, E.H.-K., & Chen, G. (2005). Utilization based duty cycle tuning MAC protocol for wireless sensor networks. *Global Telecommunications Conference, Vol.6, (pp.32-62).*

Ye, W. (2002). SCADDS: Scalable Coordination Architectures for Deeply Distributed Systems. Retrieved July 1, 2014, from http://www.isi.edu/scadds/projects/smac/

KEY TERMS AND DEFINITIONS

Energy Conservation: It refers to reducing energy consumption through using less of an energy service.

Mobile Cloud Computing: It is the combination of cloud computing, mobile computing and wireless networks to bring rich computational resources to mobile users, network operators, as well as cloud computing providers.

Mobile Computing: It is the human–computer interaction, by which a computer is expected to be transported during normal usage.

Network Performance: It refers to measures of service quality of a telecommunications product as seen by the customer.

Network Management: Network management is the operation, administration, maintenance, and provisioning of networked systems.

Peer-to-Peer: Peer-to-peer (P2P) computing or networking is a distributed application architecture that partitions tasks or work loads between peers.

Ad Hoc Network: A wireless Ad Hoc network (WANET) is a decentralized type of wireless network. The network is Ad Hoc because it does not rely on a pre-existing infrastructure, such as routers in wired networks or access points in managed (infrastructure) wireless networks.