

---

## Designing interconnected haptic interfaces and actuators for teleoperations in mobile ad hoc networks

---

George Kokkonis\* and Elias Gounopoulos

Department of Business Administration,  
Western Macedonia University of Applied Sciences,  
Grevena, Greece  
Email: gkokkonis@teiw.m.gr  
Email: elgounop@teiw.m.gr  
\*Corresponding author

Dimitrios Tsiamitros, Dimitrios Stimoniaris and  
George F. Fragulis

Department of Electrical Engineering,  
Western Macedonia University of Applied Sciences,  
Kozani, Greece  
Email: dtsiamitros@teiw.m.gr  
Email: dstimoniaris@teiw.m.gr  
Email: gfraguli@teiw.m.gr

**Abstract:** The growing acceptance of wireless ad hoc networks in war situations, emergency situations, research applications, and conference rooms has resulted in the massive growth of these networks. Interconnecting haptic interfaces in virtual environments is rather a demanding task. This paper presents the topology of the mobile ad hoc networks. Moreover, it outlines the thresholds, the limitations, and the techniques for using haptic interfaces for teleoperations in a mobile ad hoc network. Routing algorithms and congestion control techniques for using haptic interfaces in mobile ad hoc networks are analysed. Simulations are taken place in order to test whether an ad hoc network can support the interconnection of haptic interfaces. Resulting this, we could say that on-demand protocols, such as DSDV, are suitable for networks with a large number of nodes and frequent modifications of network topology.

**Keywords:** haptic interfaces; ad hoc networks; haptics; mobile ad hoc networks; MANETS; tactile interfaces; tactile feedback; haptic feedback; performance evaluation; congestion control; destination sequenced distance vector routing; DSDV; mobile networks.

**Reference** to this paper should be made as follows: Kokkonis, G., Gounopoulos, E., Tsiamitros, D., Stimoniaris, D. and Fragulis, G.F. (2020) 'Designing interconnected haptic interfaces and actuators for teleoperations in mobile ad hoc networks', *Int. J. Entertainment Technology and Management*, Vol. 1, No. 1, pp.43–63.

**Biographical notes:** George Kokkonis is a Laboratory Teaching Staff at the Department of Business Administration of Western Macedonia University of Applied Sciences, Greece. He received his PhD from the Department of

Applied Informatics, University of Macedonia, Greece. He received a five-year Engineering Diploma from the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki and MSc in Information Systems from the University of Macedonia. He has been working as a Lecturer in the last 12 years at the Department of Computer Applications in Management and Economics, Technological Educational Institution (TEI) of West Macedonia, Grevena, Greece. He has been teaching the laboratory lessons of multimedia systems, pc programming and databases. For the last three years he is responsible of the NOC of the branch of the TEI of West Macedonia in the City of Grevena, Greece. His research interests include haptic interfaces, multi-sensory protocols, teleoperations and human-computer interaction.

Elias Gounopoulos is an Information's Technology Lecturer, teaching undergraduate and postgraduate IT courses, with extensive working experience in private and public sector ICT projects. He has also worked as an evaluator in many EU funded projects in Greece and abroad [Cyprus Research Promotion Foundation (RPF)]. He holds a PhD in Informatics, MSc in Management Information Systems and BSc in Applied Informatics, all from the University of Macedonia, Greece, and BSc degree from the Department of Environmental Studies, Aristotelian University of Thessaloniki, Greece. Since August 2011, he holds a PMP certification.

Dimitrios Tsiamitros has received his PhD in Electrical and Computer Engineering from the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Greece and the respective Diploma from the same university. He is currently an Associate Professor at the Department of Electrical Engineering, University of Applied Science of Western Macedonia, Kozani, Greece. He is also scientific responsible of TEIWM at the CEF program Crocodile 2 and Technical Manager at the Interreg Europe projects FINERPOL and SUPER. He was the Technical Manager at the IPA Cross-Border Programme 'Greece-FYROM 2007–2013' projects with Acronyms PEEBPE and BIOFOSS.

Dimitrios Stimoniaris has received his PhD degree from the School of Electrical and Computer Engineering of the National Technical University of Athens (NTUA), Greece, MSc in Electrical and Computer Engineering from the Department of Electrical and Computer Engineering, Democritus University of Thrace, Greece and the respective Diploma from the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Greece. He is currently an Assistant Professor at the Department of Electrical Engineering, Technological Educational Institute of Western Macedonia, Kozani, Greece.

George F. Fragulis received his BSc in Mathematics and PhD on the subject of Systems and Control Theory from Aristotle University of Thessaloniki in 1985 and 1990, respectively. In 1994, he was an elected Assistant Professor in the Department of Computer Science and Mathematics of the Western Macedonia Univ. of Applied Sciences, Kozani, Hellas, and since 1998, he was a Professor in the same department till 2012. Since 2013, he is a Full Professor in the Department of Electrical Engineering in the same university.

---

## **1 Introduction**

Real-time teleoperations are gaining more and more attention. These teleoperations demand control devices, network infrastructure and actuators. For actuators, robotic arms are commonly used. For network infrastructure, a local area network (LAN) was also often used. But with the evolution of the internet, the teleoperations have broadened their geographical reach. Actuators and controller are now connected through the internet worldwide. An interesting scenario is when there is no network infrastructure and the connection between the actuator and the controller has to be made over a mobile ad hoc networks (MANETS) and wireless sensor networks. Applications like this can be encountered in military operations (Elliott et al., 2011) and natural disasters (Fernández-Lozano et al., 2018).

Wireless networks allow users to access information and services electronically, regardless of their geographic location. Wireless networks can be divided into two categories:

- a wireless networks based on wired getaways
- b and wireless networks without any fixed router which is called ‘MANET’ (Giordano, 2002; Stergiou et al., 2018b).

The network of these routers creates an arbitrary shape. Routers are free to move at random. This ability makes the wireless topology of the network change quickly and unpredictably.

According to RFC2501, MANETS support efficient and reliable operation on mobile wireless networks by implementing routing algorithms on mobile nodes. Such networks are designed to have dynamic, fast alternating, random, multi-hop topologies that can be implemented by relatively limited bandwidth wireless connections.

Inside the internet community, routing support for mobile hosts has now been formulated as ‘mobile IP’. This technology supports roaming where the host can connect in various ways to the internet without using its fixed address (Perkins et al., 1997; Psannis and Ishibashi, 2006).

Supporting this form of mobility requires address management and enhanced to interoperability network protocols. The purpose of mobile ad hoc networking is to extend mobility so that a set of nodes (which may be routers and hosts) can itself set up the routing table of the network in an ad hoc manner.

Ad hoc network technology is simply an improved IP-based networking technology for dynamically autonomous wireless networks. Additionally, this feature could be supported due to ad hoc’s relativity to cloud, where the hardware infrastructure remains static and also provides services to the mobile users (Stergiou and Psannis, 2016; Plageras et al., 2018).

Another interesting feature that ad hoc networks should take into consideration is their security (Chhabra and Gupta, 2014). As ad hoc network transfer valuable information as teleoperation commands from haptic interfaces to robotic arms, DDoS attacks should be taken into account during the design of transport protocols. A novel solution for preventing DDoS attacks is analysed in Chhabra et al. (2013). MANETS are often used in the internet of things. In this area an interesting security mitigating technique is proposed in Yaseen et al. (2018). All the above security issues, especially those that are targeting in cryptography are analysed in Gupta et al. (2016) and Stergiou

et al. (2018a). As MANETS have limited resources regarding bandwidth and energy, special frameworks should be used by clients to share haptics, video and other multi-sensory information without sacrificing integrity and quality (Hossain et al., 2018). An interesting factor that should be taken into consideration for MANETS is the cooperation and the resource fairness among multiple users in MANETS. A study that describes the cooperation among rational parties in cloud networks is illustrated in Li et al. (2018).

Apart from network infrastructure, a controversial sector of teleoperations is the controller. In most teleoperations and virtual games, simple controllers, without force or tactile feedback were used. Recent studies revealed that haptic controllers immerse users and increase their quality of experience (QoE) (Lopes et al., 2018; Delazio et al., 2018; Schneider et al., 2018).

In the following work we check congestion for a particular topology in hybrid and ad hoc networks. In order to control congestion, we observe some performance parameters, such as packet delivery ratio, average end-to-end data packet delay, and normalised routing load. We investigate whether haptic interfaces can be used for teleoperations in a MANET.

In Section 2, an introduction to MANETS is made, describing their operating rules, their characteristics and their use mainly in information technology and other applications. Additionally, the routing of protocols of ad hoc networks and their classification in the table-driven and on-demand driven categories is mentioned.

In the next section, Section 3 describes the routing algorithms that are used in a MANET.

In Section 4, congestion control mechanisms for using haptic interfaces are analysed.

Section 5 describes haptic interfaces, presents the QoE that haptics offer to users, it outlines the quality of service (QoS) that a network should fulfil for haptic teleoperations and proposes the congestion control algorithms that a haptic interface should enforce in order to avoid congestion in MANETS.

Following in Section 6, we define the concept of simulation, describe the structure of the ns2 simulator and the simulation scenario is presented. We present the elements of the topology on which the simulation will be based to control the performance of MANETS for haptic applications.

Section 7, presents the results of the simulation scenario are given. Packet delivery ratio, average end-to-end delay and normalised routing load are depicted.

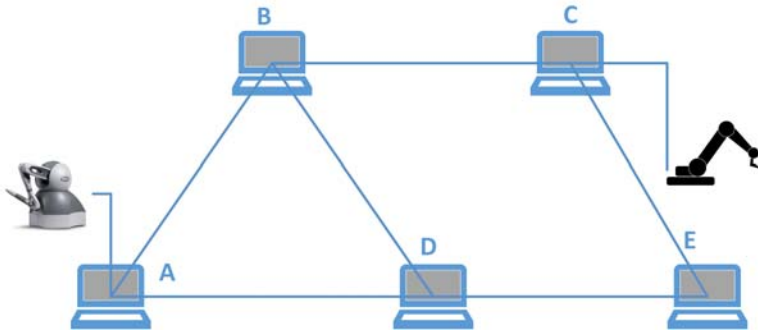
Finally, Section 8 concludes the paper and outline future work and perspectives.

## 2 Mobile ad hoc networks

To illustrate the general operating rules of a MANET, Figure 1 is presented. It depicts the ‘peer’ level, ‘multi-hop’ performance for a simple ad hoc network. Here, mobile node A communicates directly (single hop) with another identical node B when a wireless channel with sufficient propagation characteristics is available between them. On the other hand, a ‘multi-hop’ communication is necessary when one or more intermediate nodes have to act as routers between the communicating nodes. Let us assume that there is a haptic device connected to node A and we want to teleoperate a robotic arm that is connected to node C. There is no wireless passage (as shown by lines) between nodes A and C as shown in Figure 1. Nodes B and D and E should serve as intermediate routers in

the communication between nodes A and C. Thus, a feature distinguished in ad hoc networks is that all nodes must be able to act as routers on demand and at the same time act as a source and destination nodes.

**Figure 1** Connection of haptic interfaces with robotic arm through ad hoc network (see online version for colours)

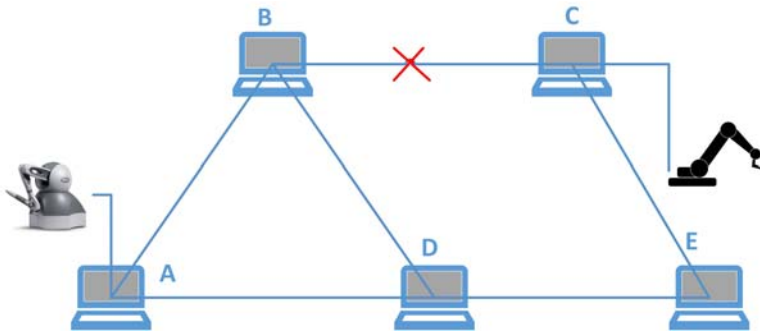


An ad hoc network starts with at least two nodes, which announce their presence with the corresponding address information. If node A is able to communicate directly with node B as shown in Figure 1, which is confirmed by exchanging the appropriate messages between them, then both update their routing tables. When a third node joins the network, then we have two possible scenarios. The first is when both nodes A and B specify that a single-hop communication with node C is feasible. The second scenario occurs when one of the two nodes, even B, recognises the signal coming from C and makes direct communication with it. In the first case, all routing is direct. On the other hand, the routing update first occurs between nodes B and C, then between B and A, and then again between B and C, confirming the mutual approximation between A and C through B.

As nodes move, it is possible over time to cause some changes in approach relationships, requiring routing to be updated. Let us consider that, for some reason, the connection between nodes B and C is not possible, as shown in Figure 2. Then, nodes A and C are still approaching each other, through nodes D and E. Equally, the original ‘loop-free’ path (A-B-C) is now replaced by a new ‘loop-free’ path (ADEC). All five nodes in the network are required to update their routing tables appropriately to reflect the change of topology that should first be detected by nodes B and C.

This interconnection between nodes can be changed for a variety of reasons, for example, when a node is wandering off the grid, if its battery runs out, or if there is a software or technical error. As more and more nodes join the network or some of the existing ones disappear, topology updates can become more complex and frequent, thus reducing the ability to exchange network information.

Finding a loop-free path between a source-destination pair may be impossible if network topology changes occur too often. Excessive frequency means that there may not be enough time to propagate to all relevant nodes the changes that occurred since the last change in network topology. For this reason, the ability to communicate is degraded by increasing mobility. A network is only cohesive when the changes in the topology occur too slowly to allow all necessary topology updates to be successful, or if the routing algorithm is capable of propagating network changes before the next change occurs.

**Figure 2** Updating ad hoc topology due to a connection failure (see online version for colours)

A MANET is made up of mobile platforms referred to here as ‘nodes’ that move arbitrarily. Nodes may be on airplanes, ships, trucks even on people. A MANET is an autonomous system of mobile nodes. The system can operate in isolation, or it can operate, through gateways and interact with a conventional network.

The MANETs have some remarkable features such as:

- 1 Dynamic topologies: nodes may move arbitrarily. Therefore, the topology of the network, which is typically multi-hop, can be changed randomly, at rapid transmission rates, and indefinitely.
- 2 Connections subject to bandwidth and variable capacity constraints. Wireless links will continue to maintain significantly lower capacity than conventional wired connections. In addition, the actual capacity of wireless communications – taking into account the effects of noise and the conditions of intervention, is far less than the maximum transmission capacity of a radio transmission. An effect of the relatively low link capacities is that network congestion is often.
- 3 Energy consumption limitations. Some or all nodes in a MANET rely on batteries. For these nodes, the issue of energy management is one of the most important optimisation issues when designing the entire system.
- 4 Limited physical security: mobile wireless networks are more vulnerable to security than conventional wired networks. They should pay particular attention to leakage issues and denial of service attacks. Existing security methods, such as encryption, are often applied to wireless networks to minimise security threats. As an advantage, it can be mentioned that the decentralised nature of a MANET network management provides greater robustness than more centralised management methods.
- 5 Scalability. Some networks (e.g., mobile military networks) may be particularly large, tens or hundreds of nodes per routing area. The need for easy scalability is crucial.

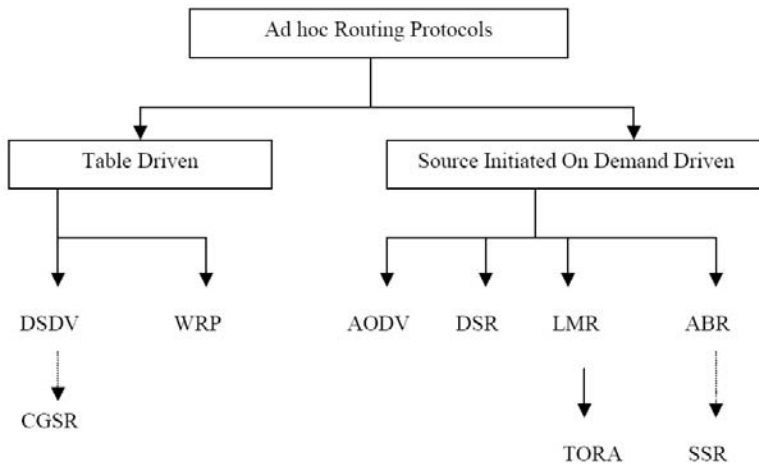
The above features create a set of assumptions and efficiency issues that should be considered when designing protocols that are beyond those of conventional networks.

### 3 Routing in mobile ad hoc wireless networks

Many of the routing algorithms proposed for ad hoc wireless networks have been adapted from routing techniques that are involved in wired networking and have a stable network topology. However, routing algorithms that are designed for wired networks are based on determining the shortest route between source and destination. Therefore, these routing algorithms cannot be applied to ad hoc networks without modifications. In ad hoc wireless networks, update packets take longer time to be promoted to other nodes. If topology information is not updated quickly, nodes may continue to route messages based on out of date topology information. This often leads to packet loss.

Routing protocols in ad hoc networks can be categorised into two categories: table-driven and on-demand routing-based protocols, as shown in Figure 3.

**Figure 3** Ad hoc routing protocols



The solid lines in Figure 3 show the direct descendants while the dashed lines show logical offspring. Although designed for the same type of network, the features of each protocol show several differences.

Table-driven routing protocols include destination sequenced distance vector (DSDV) (Perkins and Bhagwat, 1994), wireless routing protocol (WRP) (Johnson et al., 2001), global state routing (GSR) (Pei et al., 2000), fisheye state routing (FSR) (Pei et al., 2000), hierarchical state routing (HSR) (Pei et al., 1999), zone-based hierarchical link state routing protocol (ZHLS) (Takahashi et al., 2005), clusterhead gateway switching routing (CGSR) (Chiang et al., 1997).

On-demand routing protocols include: cluster base routing protocol (CBRP) (Chen et al., 2009), ad hoc demand distance vector routing (AODV) (Perkins et al., 2003), dynamic source routing protocol (DSR) (Johnson et al., 2007), temporally ordered routing algorithm (TORA) (Park and Corson, 1997), and signal stability routing (SSR) (Dube et al., 1997).

In the table-driven routing protocols, each node comprises one or more tables containing routing information to every other node in the network. All nodes upgrade these tables to maintain a consistent form of the network. When the network topology changes, nodes send update messages to inform the remaining nodes to update their routing tables.

Unlike table-driven routing protocols, the on-demand protocols all upgraded trails are not maintained in each node, but instead created if and when needed. When a source wants to communicate with a node, it initiates the destination finder mechanisms to find the path to that destination. The route remains valid until the destination becomes inaccessible or until the communication between source and destination is terminated.

### *3.1 The protocol (DSDV) – dynamic destination-sequenced distance-vector*

The destination-sequenced distance-vector routing algorithm is based on the concept of the Bellman-Ford classic algorithm, containing some substantial improvements. Each mobile node contains a routing table that includes all the available destinations, the number of hops needed to reach that destination, and the sequence number given by the destination node. The sequence number is used to distinguish the permanent paths from but new to the network to avoid creating loops. Periodically the nodes send the routing tables to each neighbouring node. Also, a node sends the routing table in the event of a significant change in its table since the last upgrade. Therefore, the upgrade can be described as both time-driven and event-driven.

Upgrades to routing tables can be sent in two ways: full-dump or partial incremental. According to the first way, the entire routing table is sent to the neighbours of the network and either whole or in packages. According to the second way only those routing table documents that have been changed since the last network upgrade are sent. If there is space in the upgrade package, then the routing rows of the nodes are included. When the network is relatively stable, incremental upgrades are sent to avoid extra traffic resulting in the avoidance of full dumps. In a rapidly changing network, incremental packets can grow sharply, resulting in frequent full dumps. Each upgrade package, except for the routing table, also contains a unique serial number given by the sender. The path with the largest serial number, i.e., the most recent one, is used. If two paths have the same serial number, the one that has the smallest path is used.

## **4 Congestion control**

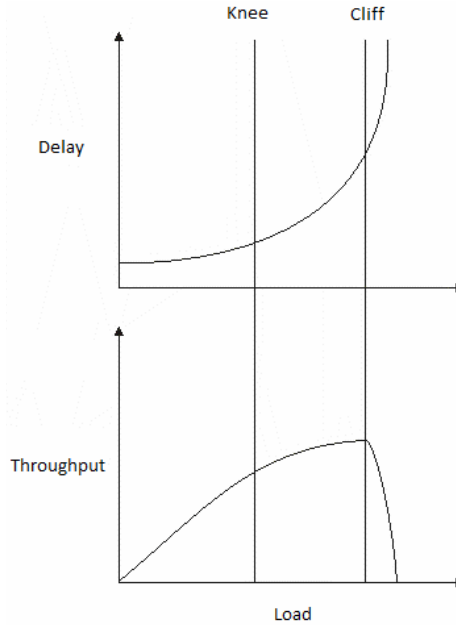
Congestion occurs in a part of the network when the total amount of data sent to the network exceeds its available capacity. Congestion is revealed by the overdue delay of packets as well as by packet loss. There are various factors that can contribute to congestion, such as the amount of data that is placed on the network, the underlying network architecture, network device specifications, packet size, and routing transfer protocol.

There are two general solutions to the problem of congestion, avoidance and control. Congestion avoidance attempts to predict when congestion is about to occur and reduces the rate of transmission of information at the given time. The algorithm must work in such a way that it keeps the response time based on the load and the rate of delivery and on the load to the left of the knee position as shown in Figure 4. The congestion control



attempts to fully exploit the resources networks to transfer data at a rate close to network capacity. Network capacity is defined as the point at which any increase in traffic will increase the delay but not the rate of return. Congestion control algorithms try to increase traffic until network capacity is reached, and then slow down the transmission rate. Therefore, these algorithms try to operate to the left of the ‘cliff’ of Figure 4.

**Figure 4** Load diagram in function of delay and throughput



#### 4.1 Congestion control mechanisms

Some of the most common congestion control mechanisms act as a reduction requirement that incorporates:

- *refusal of services*: preventing further resource allocation
- *downgrade of services*: forcing network users to reduce the load they place on the network
- *design*: network users are designing their requests to use their resources without overloading them.

To enable each of these mechanisms to work, information about the status of the network is required. Various approaches have been proposed and developed to facilitate this:

- observation of congestion data such as packet loss and delay
- deliver feeds from the point of congestion to the source
- rejection of incoming traffic from the congested node
- congestion control packets sent through the network

- packages may have feedback fields that indicate the network congestion status at source.

All these mechanisms for their efficient operation must work quickly, have a low header, be fair and allow the network to be used at its limit.

#### *4.2 Congestion issues in wireless networks*

Within ad hoc wireless networks several issues complicate the identification and the control of congestion:

- intervention from other nodes
- path failures
- variable quality of radio signals
- congestion
- transmission power.

In the case of packet loss, appropriate action is not easy because determining the cause of the loss is difficult. There have been several mechanisms proposed to help sort the causes of packet loss, but they all add extra complexity, may not be compatible with existing protocols, and do not cover all possible causes.

#### *4.3 Weight connection links in ad hoc networks*

To control congestion, most routing algorithms try to detect the state of the links between the nodes. Each link is defined as a weight based on a period of time in which a packet is transmitted. These weights are summed to give the overall weight for a route. The route with the lowest total weight is selected. This route selection method selects connections with the least delay and higher bandwidth, and solves many congestion-related issues. Transmission times are shorter, require less power consumption and cause less congestion.

#### *4.4 Solving local congestion control*

Since congestion is local, it means that congestion should be addressed locally. The packets should be routed around the congestion area. This will avoid packet loss due to congestion and would not reduce the transmission rate. The nodes should measure the number of queued packets. If for a period of time the average number of packages exceeds a certain threshold, the node is considered to be saturated. Immediately the saturated node should transmit a congestion notification package to all its neighbours. All nodes control and record the volume of data they transmit through neighbours. If the average volume of data over a period of time to any neighbouring node exceeds a certain level, this node records that it has exceeded the shipping limit for that neighbouring node.

On receipt of a congestion notification, the sending node checks whether it has exceeded the shipping limit to the saturated neighbour node. In this case, the sending node should try to find an alternate path to its destination, avoiding the saturated node. If all available routes contain nodes that have recently announced congestion and the

source node has exceeded the shipping limit to those nodes, then it must wait until an alternate path is available. The purpose of the shipping limit allows the algorithm to target those nodes that transmit large volumes of traffic through a saturated node.

## **5 Haptic interfaces**

Touch is a sensation of key importance for the understanding of the surrounding environment and its understanding. The sensation of touch involves the skin's discretion to separate the surface features of objects in the surrounding area, and the kinetic ability of the muscles/tendons to determine the body's position and the distances of the limbs from the objects. The sensory organ of the touch, the skin, covers the entire surface of the human body and concentrates different types of stimulus receptors, showing different sensitivity to a variety of different tactile stimuli.

The feeling of touch is difficult to synthesise. Additionally, the visual and auditory sensory channels predominate over the haptic channel. In contrast, however, with both vision and hearing are only passive senses, the touch allows both passive intake of the stimulus but also may produce an feedback through active motion response.

Haptic interfaces allow human (skin and kinaesthetic) interaction with virtual or remote robotic systems. They are feedback devices that are used to simulate functions for realistic transfer of skin and muscular irritations to the user. An integrated tactile interaction system includes the interaction device and a simulation, a representation of the virtual environment with which the device handler interacts in real-time.

By incorporating tactile feedback into a virtual environment, users can recognise, push, pull, understand and manipulate environmental objects beyond their simple visualisation.

Haptic interfaces are characterised by two basic functions:

- 1 they are used to measure the movement of the user (position, speed, acceleration) and enforce feedback on the user
- 2 they enforce spatial and temporal constraints of the user's environment based on physical laws.

The haptic interfaces should:

- allow perception of movement/position of body limbs
- improve the skilful execution of tasks
- allow training in a secure, virtual environment
- for remote control systems: improve operator performance.

Some crucial factors that should be taken into consideration in designing an integrated haptic interface are:

- The QoS that the network should fulfil in order to successfully transfer the haptic feeling to the user. Recent experiments revealed the QoS requirement that a network should fulfil are depicted in Table 1.

- The synchronisation of the supermedia streams such as the haptic data stream, voice, video and graphics. A successful synchronisation technique for haptic interfaces is the enhanced VTR and is analysed in Kokkonis et al. (2017).
- If the QoS of the network is not adequate for haptic data transferring, then congestion control techniques should be enforced in order to avoid congestion and degrade the QoE of the user. The most suitable congestion control techniques for haptic interfaces are described in Kokkonis et al. (2015).

The congestion control techniques that are used in this research is the grouping of packets (adaptive packetisation) which results in adaptive transmission rate. As described in Table 1, most haptic interfaces offer update rate up to 1,000 packets per sec. This update rate is extremely high for the MANETS. As multiple hops may occur until a packet reaches its destination, the transmission rate should be as small as possible. In previous research (Kokkonis et al., 2018) we concluded that as the number of hops increases, the number of packet loss and delay is also increased.

**Table 1** QOS requirements for supermedia streams

<i>QOS</i>	<i>Haptics</i>	<i>Video</i>	<i>Audio</i>	<i>Graphics</i>
Jitter (ms)	$\leq 2$	$\leq 30$	$\leq 30$	$\leq 30$
Delay (ms)	$\leq 50$	$\leq 400$	$\leq 150$	$\leq 100\text{--}300$
Packet loss (%)	$\leq 10$	$\leq 1$	$\leq 1$	$\leq 10$
Update rate (Hz)	$\geq 1,000$	$\geq 30$	$\geq 50$	$\geq 30$
Packet size (bytes)	64–128	$\leq \text{MTU}$	160–320	192–5,000
Throughput (kbps)	512–1,024	25,000–40,000	64–128	45–1,200

*Source:* Eid et al. (2011), Iwata et al. (2010), Suzuki and Katsura (2013), Isomura et al. (2013) and Hamam and El Saddik (2013)

In Hasegawa et al. (1999) the authors proposed a haptic controller that reduces the high-speed refresh rate of a haptic device from 1 KHz to 20 Hz refresh rate of a virtual world manager. This sampling coherency problem is solved by introducing an inter-process communication, in which a haptic controller sends back the information on the force integration to the virtual world manager besides the position of the haptic device. Brooks (1990) reported that it is enough for teleoperation systems to send the information of the position of the haptic interface to the robotic arm with update rate of 5 to 10 Hz.

In Kokkonis et al. (2015) it is proposed an algorithm that reduces the sending rate of the data packets of the haptic interface based on the conditions of the network.

This reduction of the update rate is taken into consideration, in order to successfully transfer the haptic data over the network. In Kokkonis et al. (2016) it is reported that as the number of the intermediate wireless nodes or the update rate increases, the delay, the jitter and the packet loss is also increased.

## 6 Simulation scenario

Simulation is a technique that mimics the function of a real system as it develops over time. The simulation can be considered as a statistical sampling from a mathematical or logical model in order to find results that are essentially estimators of the actual operating

and performance parameters of the system under study. It is worth noting that in the simulation we distinguish three phases: the preliminary process and modelling, the transfer of the model into code of a programming language and the analysis of the results.

### *6.1 Advantages and disadvantages of simulation*

The advantages of simulation as a modelling process, has many advantages:

- 1 The simulation compresses the real-time needed for an experiment. The experiment is virtually carried out on the computer and it imitates the time parameter. So, an experiment that may take years in real-time, with the simulation may take a few minutes.
- 2 The use of the model does not cause harassment in the actual system.
- 3 It is easier to avoid the sampling error.
- 4 The modelling theory that supports it is well defined and implemented without modifications.
- 5 Analytical models require too many assumptions to solve, resulting in deviation from the operating conditions of the actual system. In a simulation model, it is not necessary to make many assumptions.
- 6 Every simulation scenario is repeated several times to study different system configurations or policies. However, this is difficult to do when a real-world experiment is actually done. There is the capability to stop the simulation, control the results and continue the simulation from the interruption point.
- 7 In the process of constructing the model, system variabilities are detected, which may not be visible during the operation. These variabilities are included in the model and are taken into account through statistical analysis during optimisation.
- 8 Simulation is sometimes an excellent tool for training staff, executives, students, etc.
- 9 Finally, the simulation model can help design a new system that does not exist.

On the other hand, there are some drawbacks concerning simulation models:

- 1 The techniques are not a clear analytical process of optimising the operation of a system.
- 2 The development of the model may be time consuming for software development and computing resources.
- 3 The collection of the initial input data may be difficult and costly.
- 4 The simulation model is documented by a series of logical charts and algorithms. Using the program without full knowledge of the assumptions may lead to erroneous conclusions.

### *6.2 The ns2 simulator*

The ns2 simulator is an open source discrete event simulator widespread in the research community. It has been developed at Berkeley University in California and expanded to

Carnegie Mellon University where it was used to simulate wireless networks. These extensions provide a detailed behavioural pattern of the physical and connective layer of a wireless site and allow the arbitrary movement of the nodes within the network. Some of the most recent proposed routing protocols (DSDV, TORA, DSR, and AODV) are also linked to ns2. In each run the simulator takes a scenario that describes the exact movement of each mobile node along with the packet frequency coming from each node as the time progresses.

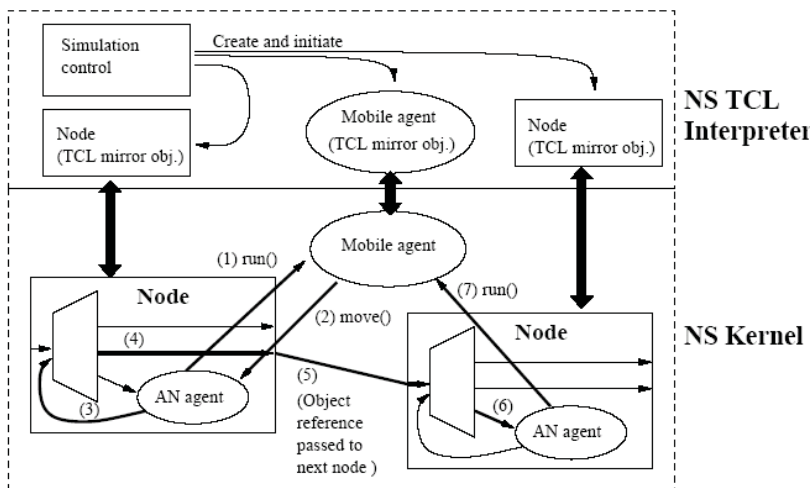
### 6.3 The structure of ns2

The ns2 is a network simulator that simulates a variety of IP networks and is written in C++ and Otcl (Issariyakul and Hossain, 2012). It implements transport protocols like TCP and UDP, traffic source behaviours such as FTP, telnet, web, and more. Ns2 also implements multicasting and some of the MAC layer protocols for local LAN simulations.

In addition, ns2 is an object-oriented Tcl command file interpreter with a simulation event programmer and network object libraries (Issariyakul and Hossain, 2012). Thus, in order to use ns2, we can program in the OTcl command file language. For organising and running a simulation scenario, a user must write a script of OTcl commands that initiates an event scheduler, defines network topology using network objects, sets traffic sources, and start and stop sending packets through the event scheduler.

Ns2 as mentioned above is written not only in OTcl but also in C++. In order to reduce processing time, the event scheduler and basic network objects are written and compiled using C++. These compiled objects are available to the interpreter by creating a corresponding OTcl object. In this way, C++ object are given in OTcl. Objects in C++, which should not be tested in a simulation or used internally by another object, should not be connected to OTcl. In addition, an object can be implemented entirely in OTcl. The structure of ns2 is depicted in Figure 5.

Figure 5 Structure of ns2

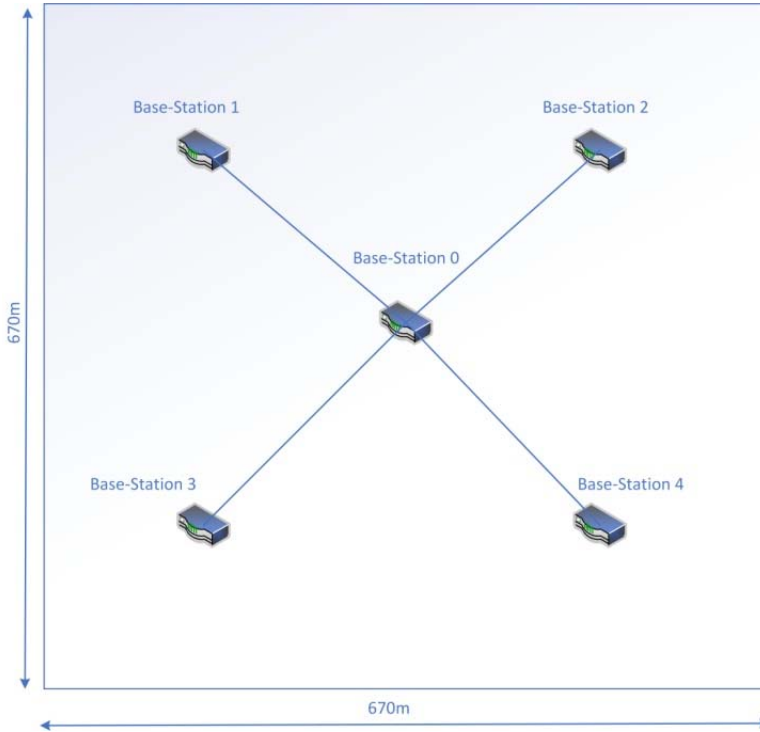


Source: Issariyakul and Hossain (2012)

#### 6.4 Topology of simulation

In the present work a  $670\text{ m} \times 670\text{ m}$  square is simulated, in which freely movable nodes are communicating. The number of nodes ranges from 10 to 50. Inside this block are also fixed nodes that communicate with the mobile nodes with the help of base-stations. The base-station nodes are placed in appropriate locations to adequately cover the entire geographic area of the simulation. Five base-station nodes are used in a star topology. All base-stations are wired through the base-station BS0, as shown in Figure 6.

**Figure 6** Simulation topology, position of base-stations (see online version for colours)



The initial placement of the mobile nodes is done randomly. The ‘setdest’ program contained within ns2 is used to assist with this placement.

#### 6.5 Mobility of nodes

The mobility model applied to the simulation is the ‘random way-point’. According to this model the nodes move at a constant speed which takes values between the range  $[0, V_{\max} = 1.5\text{ m/s}]$ . The velocity vector changes at regular intervals, while between this interchange the node remains stationary for  $t = 40\text{ s}$ . Such mobilities are encountered in conferences, in emergency situations and in the simple case where a person walks in the street. All nodes remain stationary for the first 40 seconds and then begin to move. To achieve this mobility, the setdest program was used with the following syntax:

- `./setdest -v 1 -n 40 -p 40 -M 1.5 -t 900 -x 670 -y 670`

In the above command,  $v$  indicates the version of `setdest`,  $n$  the number of mobile nodes,  $p$  the pause time in sec,  $M$  the maximum node speed,  $t$  the simulation time, and  $x, y$  the geographic dimensions of the simulation. The total simulation time is set to  $t = 900$  sec.

## 6.6 Communication of nodes

The nodes communicate with each other at a fixed transmission rate of four packets per second. The packet size is set to 512 bytes. This packet size was chosen as it represents a frame of eight packets from 64 bytes each. We enforced the congestion control technique of grouping of packets which reduces the packet rate. The grouping of eight packets per frame does not decrease the QoE of the user as shown in Fujimoto and Ishibashi (2005). Sending packets begins at  $t = 40$  sec and stops at the end of the simulation at  $t = 900$  s. The maximum number of connections between the nodes was set 8. To achieve this, the `cbrgen` program has the following syntax:

- `./ns cbrgen.tcl -type cbr -nn 50 -seed 1 -mc 8 -rate 0.25`

In the type parameter we state the type of connection and we can choose between `cbr` and `tcp`. In parameter `nn` we select the number of nodes, in the `seed` we select the randomness coefficient, in `mc` the maximum number of connections, and in the `rate` the packet transmission rate per second.

At the beginning of the simulation and during all the nodes send signals to their neighbours to declare their presence.

Each neighbour node receiving this signal registers this node in the routing table and sends a message to its neighbours as well. This answer is shown in Figure 7.

The data, i.e., the interchange of data between nodes, occurs after 40 sec.

In the event that a node cannot communicate with another node because it could not update its routing table then the packet drops.

## 6.7 Performance metrics

To measure performance of routing protocols, various performance parameters are taken into account. The parameters that were measured for the study of the routing protocols are:

- *Package delivery ratio*: the packet delivery ratio in the simulation is defined as the ratio between the packet numbers sent to the ‘constant bit rate’ (CBR) sources and the packet numbers delivered to their destination. It describes the percentage of packets reaching its destination.

$$\text{Package delivery ratio} = \text{Number of packets delivered} / \text{Number of packets sent}$$

- *Normalised routing load*: it depicts the total number of routing packets sent during the simulation via the number of packets sent from the source nodes. It is noted that each ‘hop’ of a routing packet is counted as a different routed packet.

*Normalised routing load*

$$= \left( \begin{array}{l} \text{Number of packets routed} \\ - \text{Number of packets sent from sources} \end{array} \right) / \text{Number of packets sent from sources}$$



The normalised routing load express the efficiency of the routing protocols that enforced in the simulation.

- *Average end-to-end delay*: it includes all possible delays caused by:
  - a temporary storage during route finding
  - b delay as packets wait in queue
  - c retransmission delays at MAC level
  - d transmission duration.

The average end-to-end delay is very important to haptic applications as they require little delay for data transfer. Table 1 show that the delay of haptic packets should be below 50 ms, in order to have maximum QoE for the user.

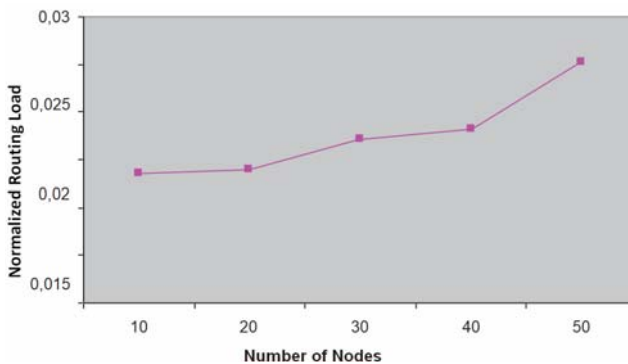
## 7 Simulation results

Running the model described in the above section for 10, 20, 30, 40 and 50 nodes, respectively, the following diagrams appear:

### 7.1 Normalised routing load

As can be seen from Figure 7, the normalised load increases as the number of wireless nodes increases. It can be noted that until 40 nodes the normalised routing load is almost steady. For 50 nodes the normalised load increases quick, which means that congestion is starting to appear. Until 40 nodes the routing load is near 0.023. This means that the number of routed packets was increased only by 2.3%. The normalised routing load was quite small because the nature of ad hoc networks. Nodes try to communicate peer-to-peer directly with each other and try to avoid the intermediate routers.

**Figure 7** Normalised routing load (see online version for colours)

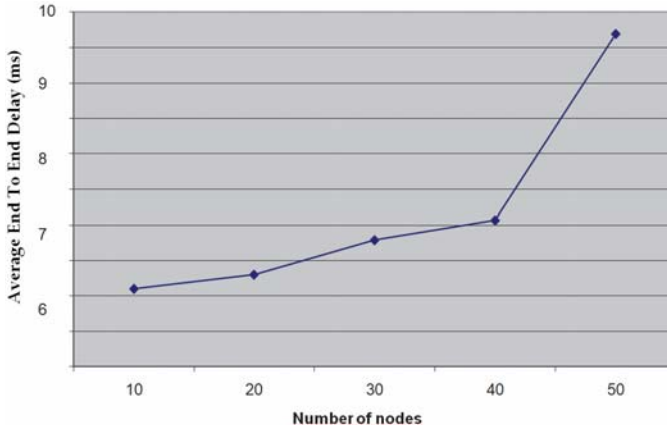


### 7.2 Average end-to-end delay

As expected, the average end-to-end delay increases with the increase in node number, as the intermediate nodes increase, the routing packets increase. Again, as in Figure 7, it is

obvious that until 40 nodes the average end-to-end delay is almost steady. For 50 nodes the network is heading to congestion. For 40 nodes the delay is 7.1 ms, which is barely acceptable for interconnecting haptic interfaces. For 50 nodes the delay is 9.65 ms which is unacceptable for haptic applications.

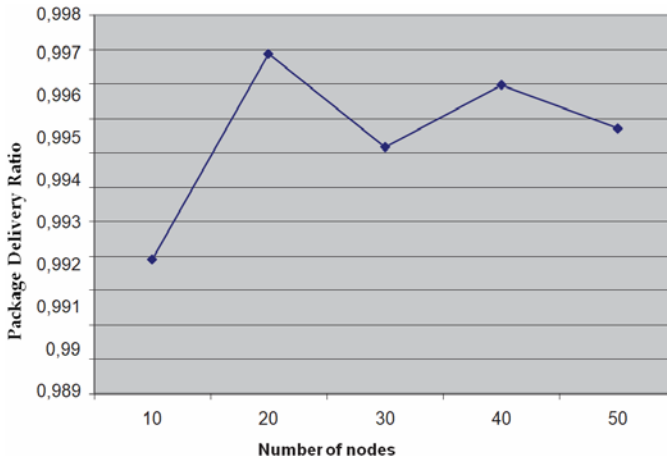
**Figure 8** Average end-to-end delay (see online version for colours)



### 7.3 Package delivery ratio

Figure 9 shows that although the nodes increase, the packet delivery ratio is almost steady near to 0.995. Congestion has not begun to appear as packet loss is fairly low. If we combine Figures 8 and 9, for 50 nodes, the delay has started to increase but the packet loss is little. This means that the buffers at the routers have many queued packets but their buffers are not yet overflowing.

**Figure 9** Package delivery ratio (see online version for colours)



## 8 Conclusions

From the above charts, and from the protocols analysis that preceded it, one concludes that for small MANETS, with few haptic interfaces, the DSDV routing protocol works satisfactorily. As the number of nodes grows and new haptic interfaces are added, congestion occurs. The result of this congestion is to increase the end-to-end delay of the packets, but also to increase the number of lost packets. A good congestion control technique is to group the number of packets into frames, so as to lower the packet rate. An intermediate haptic controller should be enforced in order to lower the update rate of the haptic packets that are transmitted through the network.

The DSDV protocol, as mentioned above, is a table-driven protocol, so as the network grows, its routing tables grows. Spreading this routing table to all network nodes is intensive and time consuming. On the other hand, on-demand protocol, when a node wants to communicate with one another, the routing tables will be upgraded and only the nodes joining the two communicating nodes. Therefore, on-demand protocols, such as DSR, are suitable for networks with a large number of nodes and frequent modifications of network topology.

It has also been observed that as the number of wireless nodes increases in a wireless network, both the normalised routing load and the average end-to-end delay increase. On the other hand, packet delivery ratio shows that you do not change in the same way, because at the first stages of congestion, the routers buffer the queued packets. At that point the source should decrease its sending rate, in order to avoid congestion.

As a future work the authors intent to enforce the DSDV and the DSR protocol for real world experiments on interconnecting haptic interfaces with robotic arms over MANETS. The number of the intermediate nodes and the update rate will be variously changed in order to study the variation of delay, jitter and packet loss of the network. The maximum QoE of the haptic user will also be investigated the mean opinion score evaluation.

## References

- Brooks, T.L. (1990) 'Telerobotic response requirements', *Proc. of the 1990 IEEE International Conference on System, Man and Cybernetics*, pp.113–120.
- Chen, G., Li, C., Ye, M. and Wu, J. (2009) 'An unequal cluster-based routing protocol in wireless sensor networks', *Wireless Networks*, Vol. 15, No. 2, pp.193–207.
- Chhabra, M. and Gupta, B.B. (2014) 'An efficient scheme to prevent DDoS flooding attacks in mobile ad-hoc network (MANET)', *Research Journal of Applied Sciences, Engineering and Technology*, Vol. 7, No. 10, pp.2033–2039.
- Chhabra, M., Gupta, B. and Almomani, A. (2013) 'A novel solution to handle DDOS attack in MANET', *Journal of Information Security*, Vol. 4, No. 3, p.165.
- Chiang, C.C., Wu, H.K., Liu, W. and Gerla, M. (1997) 'Routing in clustered multihop, mobile wireless networks with fading channel', in *Proc. of IEEE SICON*, Vol. 97, pp.197–211.
- Delazio, A., Nakagaki, K., Klatzky, R.L., Hudson, S.E., Lehman, J.F. and Sample, A.P. (2018) 'Force jacket: pneumatically-actuated jacket for embodied haptic experiences', in *Proc. of the 2018 CHI Conference on Human Factors in Computing Systems (CHI'18)*, ACM.
- Dube, R., Rais, C.D., Wang, K.Y. and Tripathi, S.K. (1997) 'Signal stability-based adaptive routing (SSA) for ad hoc mobile networks', *IEEE Personal Communications*, Vol. 4, No. 1, pp.36–45.

- Eid, M., Cha, J. and El Saddik, A. (2011) ‘Admux: an adaptive multiplexer for haptic-audio-visual data communication’, *IEEE Trans. Instrum. Meas.*, Vol. 60, No. 1, pp.21–31.
- Elliott, L., Schmeisser, E. and Redden, E. (2011) ‘Development of tactile and haptic systems for U.S. infantry navigation and communication’, in Smith, M. and Salvendy, G. (Eds.): *Human Interface and the Management of Information. Interacting with Information, ser. Lecture Notes in Computer Science*, Vol. 6771, pp.399–407, Springer Berlin, Heidelberg.
- Fernández-Lozano, J.J., Mandow, A., Martín-Guzmán, M., Martín-Ávila, J., Gomez-Ruiz, J.A., Socarras-Bertiz, C., Miranda-Páez, J. and García-Cerezo, A. (2018) ‘Integration of a canine agent in a wireless sensor network for information gathering in search and rescue missions’, *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, pp.5685–5690.
- Fujimoto, M. and Ishibashi, Y. (2005) ‘Packetization interval of haptic media in networked virtual environments’, in *Proc. of 4th ACM SIGCOMM Workshop on Network and System Support for Games*, ACM, pp.1–6.
- Giordano, S. (2002) ‘Mobile ad hoc networks’, *Handbook of Wireless Networks and Mobile Computing*, pp.325–346, Wiley, New York.
- Gupta, B., Agrawal, D.P. and Yamaguchi, S. (2016) *Handbook of Research on Modern Cryptographic Solutions for Computer and Cyber Security*, IGI Global, Hershey, PA, USA.
- Hamam, A. and El Saddik, A. (2013) ‘Toward a mathematical model for quality of experience evaluation of haptic applications’, *IEEE Trans. Instrum. Meas.*, Vol. 62, No. 12, pp.3315–3322.
- Hasegawa, S., Ishii, M., Koike, Y. and Sato, M. (1999) ‘Inter-process communication for force display of dynamic virtual world’, *ASME Dyn. Syst. Control Div. Publ. Dsc.*, Vol. 67, pp.211–218.
- Hossain, M.S., Muhammad, G., Abdul, W., Song, B. and Gupta, B.B. (2018) ‘Cloud-assisted secure video transmission and sharing framework for smart cities’, *Future Generation Computer Systems*, Vol. 83, pp.596–606, ISSN 0167-739X, <https://doi.org/10.1016/j.future.2017.03.029>.
- Isomura, E., Tasaka, S. and Nunome, T. (2013) ‘A multidimensional QoE monitoring system for audiovisual and haptic interactive IP communications’, *IEEE Consumer Communications and Networking Conference, CCNC 2013*, pp.196–202.
- Issariyakul, T. and Hossain, E. (2012) ‘Introduction to network simulator 2 (NS2)’, in *Introduction to Network Simulator NS2*, pp.21–40, Springer, Boston.
- Iwata, K., Ishibashi, Y., Fukushima, N. and Sugawara, S. (2010) ‘Qoe assessment in haptic media, sound, and video transmission: effect of playout buffering control’, *Comput. Entertain.*, Vol. 8, No. 2, pp.1–12.
- Johnson, D., Hu, Y.C. and Maltz, D. (2007) *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4*, No. RFC 4728.
- Johnson, D.B., Maltz, D.A. and Broch, J. (2001) ‘DSR: the dynamic source routing protocol for multihop wireless ad hoc networks’, in *Ad Hoc Networking*, pp.139–172, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Kokkonis, G., Psannis, K.E. and Roumeliotis, M. (2015) ‘Network adaptive flow control algorithm for haptic data over the internet-NAFCAH’, *Int. Conference on Genetic and Evolutionary Computing*, Springer, pp.93–102.
- Kokkonis, G., Psannis, K.E. and Roumeliotis, M. (2016) ‘Real time haptic data transferring’, in *Wireless Days (WD)*, IEEE, pp.1–3.
- Kokkonis, G., Psannis, K.E., Roumeliotis, M. and Schonfeld, D. (2017) ‘Real-time wireless multisensory smart surveillance with 3D-HEVC streams for internet-of-things (IoT)’, *The Journal of Supercomputing*, Vol. 73, No. 3, pp.1044–1062.
- Kokkonis, G., Psannis, K.E., Roumeliotis, M., Ishibashi, Y., Kim, B.G. and Constantinides, A.G. (2018) ‘Transferring wireless high update rate supermedia streams over IoT’, in Yager, R. and Pascual Espada, J. (Eds.): *New Advances in the Internet of Things. Studies in Computational Intelligence*, Vol. 715, Springer, Cham.

- Li, T., Gupta, B.B. and Metere, R. (2018) 'Socially-conforming cooperative computation in cloud networks', *Journal of Parallel and Distributed Computing*, Vol. 117, pp.274–280, ISSN: 0743-7315, <https://doi.org/10.1016/j.jpdc.2017.06.006>.
- Lopes, P., You, S., Ion, A. and Baudisch, P. (2018) 'Adding force feedback to mixed reality experiences and games using electrical muscle stimulation', in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI'18)*, ACM, New York, NY, USA, Paper 446, 13pp, DOI: <https://doi.org/10.1145/3173574.3174020>.
- Park, V.D. and Corson, M.S. (1997) 'A highly adaptive distributed routing algorithm for mobile wireless networks', *Sixteenth Annual Joint Conf. IEEE Computer and Communications Societies. Driving the Information Revolution*, Vol. 3, pp.1405–1413.
- Pei, G., Gerla, M. and Chen, T.W. (2000) 'Fisheye state routing: a routing scheme for ad hoc wireless networks', *IEEE Int. Conf. on Communications*, Vol. 1, pp.70–74.
- Pei, G., Gerla, M., Hong, X. and Chiang, C.C. (1999) 'A wireless hierarchical routing protocol with group mobility', in *Wireless Communications and Networking Conference*, IEEE, Vol. 3, pp.1538–1542.
- Perkins, C., Belding-Royer, E. and Das, S. (2003) *Ad Hoc On-Demand Distance Vector (AODV) Routing*, No. RFC 3561.
- Perkins, C.E. and Bhagwat, P. (1994) 'Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers', in *ACM SIGCOMM Computer Communication Review*, Vol. 24, No. 4, pp.234–244.
- Perkins, C.E., Alpert, S.R. and Woolf, B. (1997) *Mobile IP: Design Principles and Practices*, Addison-Wesley Longman Publishing Co., Boston, MA, USA ©1997, ISBN: 0201634694.
- Plageras, A.P., Psannis, K.E., Stergiou, C., Wang, H. and Gupta, B.B. (2018) 'Efficient IoT-based sensor big data collection-processing and analysis in smart buildings', *Future Generation Computer Systems*, Vol. 82, pp.349–357, <https://doi.org/10.1016/j.future.2017.09.082>.
- Psannis, K. and Ishibashi, Y. (2006) 'Impact of video coding on delay and jitter in 3G wireless video multicast services', *EURASIP Journal on Wireless Communications and Networking*, Article ID 24616, pp.1–7, <https://doi.org/10.1155/WCN/2006/24616>.
- Schneider, O., Shigeyama, J., Kovacs, R., Roumen, T.J., Marwecki, S., Boeckhoff, N., Gloeckner, D.A., Bounama, J. and Baudisch, P. (2018) 'DualPanto: a haptic device that enables blind users to continuously interact with virtual worlds', in *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST'18)*, ACM 2018.
- Stergiou, C. and Psannis, K.E. (2016) 'Recent advances delivered by mobile cloud computing and internet of things for big data applications: a survey', *Wiley, International Journal of Network Management*, pp.1–12, DOI: 10.1002/nem.1930.
- Stergiou, C., Psannis, K.E., Kim, B.G. and Gupta, B. (2018a) 'Secure integration of IoT and cloud computing', *Future Generation Computer Systems*, Vol. 78, pp.964–975, <https://doi.org/10.1016/j.future.2016.11.031>.
- Stergiou, C., Psannis, K.E., Plageras, A.P., Ishibashi, Y. and Kim, B-G. (2018b) 'Algorithms for efficient digital media transmission over IoT and cloud networking', *Journal of Multimedia Information System*, Vol. 5, No. 1, pp.1–10.
- Suzuki, N. and Katsura, S. (2013) 'Evaluation of QoS in haptic communication based on bilateral control', *IEEE Int. Conf. on Mechatronics, ICM 2013*, pp.886–891.
- Takahashi, M., Bandai, M. and Sasase, I. (2005) 'Multilevel zone-based hierarchical link state routing with location search technique applying hierarchical request for mobile ad hoc networks', *Electronics and Communications in Japan*, Vol. 88, No. 1, pp.44–52.
- Yaseen, Q., Aldwairi, M., Jararweh, Y., Al-Ayyoub, M. and Gupta, B. (2018) 'Collusion attacks mitigation in internet of things: a fog based model', *Multimedia Tools and Applications*, Vol. 77, No. 14, pp.18249–18268.