INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Energy efficient architectures for the current and future Internet

Tiago Filipe da Cruz Silva
(Licenciado)

Dissertação para obter o grau de Mestre em

Engenharia de Redes de Comunicações

Júri

Presidente: Prof. Dr. Paulo Jorge Pires Ferreira
Orientador: Prof. Dr. Artur Miguel do Amaral Arsénio
Vogal: Prof. Dr. Teresa Maria Sá Ferreira Vazão Vasques

Maio 2012

Acknowledgements

First of all, I would like to express my appreciation to my advisor, Prof. Dr. Artur Arsénio, for his assistance and guidance in both research and implementation stages of this thesis.

Also, to my friends a big thanks for the great support and companionship during these last years. Their help was very important to overcome the most stressful and exhausting times.

Finally, i would like to address a very special thanks to my family for their unconditional love and support throughout my life. It was them who made this possible.

Lisbon, May 2012
Tiago Silva

Resumo

A escalabilidade, a mobilidade, a segurança, a performance e a eficiência energética são os principais desafios que a Internet tem de enfrentar actualmente. A resolução destes problemas pode conduzir a uma completa reestruturação da arquitectura actual. Existem algumas propostas para a Internet do futuro que desenvolveram novas arquitecturas baseando-se no *"Clean Slate Design"*, p. ex. 4WARD [1], ANA [2] e PSIRP [3]. Estas propostas fazem uso de novos paradigmas, tais como: virtualização, *autonomic networking* e *publish-subscribe*. Por outro lado, existem algumas propostas que se focam na melhoria de alguns dos componentes utilizados na arquitectura actual, p. ex. FIRA [4], NIRA [5] e XCP [6].

Hoje em dia, o consumo de energia tornou-se uma grande preocupação devido a razões ambientais e económicas. A actual infraestrutura da Internet desperdiça grandes quantidades de energia por causa dos elementos de rede estarem sempre a trabalhar na sua máxima capacidade, mesmo numa situação de pouca procura de tráfego. Para reduzir este desperdício de energia da Internet é necessário permitir que os elementos de rede entrem em modos de poupança de energia. No entanto, estes mecanismos de poupança de energia podem deteriorar o desempenho da rede.

Neste trabalho foi desenvolvido um modelo de poupança de energia a ser aplicado na arquitectura da Internet e na arquitectura do PSIRP. O modelo tem em conta o *tradeoff* entre a redução do consumo de energia e o desempenho da rede. Isto é alcançado através da classificação dos elementos de rede de acordo com a sua importância no envio de pacotes. Esta solução foi implementada e avaliada no simulador de rede NS3.

Os resultados da avaliação demonstram que é possível alcançar uma poupança média de energia de 45% numa situação de escassa procura de tráfego. Na existência de muita procura, o consumo de energia é apenas reduzido em 23% na média. A avaliação demonstra também que desactivar apenas os links não traz quaisquer beneficios no que diz respeito à poupança de energia.

Palavras Chave

Internet, *Clean Slate Design*, Consumo de energia, Poupança energética, Engenharia de tráfego, Desligar elementos de rede.

Abstract

The current Internet has to face many challenges, such as: scalability, mobility, security, performance and energy efficiency. To address these concerns it might be necessary to rethink the current architecture. There are some proposals for the future Internet that developed a completely new one, following the principle of the "Clean Slate Design", e.g. 4WARD [1], ANA [2] and PSIRP [3]. These proposals follow completely new paradigms, such as: virtualization, autonomic networking and publish-subscribe. On the other hand, there are some proposals that will only try to improve some components used in today's architecture, e.g. FIRA [4], NIRA [5] and XCP [6].

The energy consumption is a major concern nowadays not only because of environmental issues but also economical ones. The current Internet infrastructure wastes a lot of energy because the network elements are always working at their full capacity even with a low traffic demand. This energy waste of the Internet can be reduced by allowing some network elements to enter in energy saving modes. However, it may lead to the decrease of the network performance.

This work consists on applying an energy saving model to the current Internet architecture and to the PSIRP architecture, taking into account the tradeoff between energy saving and network performance. This is mainly achieved by classifying the network elements according to their importance in the packet delivery process. Also, this solution was implemented and evaluated using the NS3 simulator.

The evaluation results show that with a low traffic demand the energy consumption can be reduced by 45% in average. On the other hand, with a high traffic demand the energy consumption is reduced by 23% in average. The evaluation also shows that turning off only the links will not allow any significant reduction in the energy consumption.

Keywords

Internet, Clean Slate Design, Energy consumption, Energy saving, Traffic engineering, Turning off network elements.

Contents

List of Figures

List of Tables

Acronyms

4WARD	Wired and Wireless World Wide Architecture and Design
ANA	Autonomic Network Architecture
AP	Access Point
COST	European Cooperation in Science and Technology
EU	European Union
FARA	Forwarding directive, Association, and Rendezvous Architecture
FP6	Sixth Framework Programme
FP7	Seventh Framework Programme
GHG	Greenhouse Gas
ICT	Information and Communications Technology
IP	Internet Protocol
ISP	Internet Service Provider
LAN	Local Area Network
MANET	Mobile Ad-Hoc Network
MPLS	Multiprotocol Label Switching
NREN	National Research and Education Network
NIRA	New Internet Routing Architecture
OSPF	Open Shortest Path First
P2P	Peer-To-Peer
PSIRP	Publish-Subscribe Internetworking Routing Paradigm
QoS	Quality of Service
RP	Rendezvous Point

RTT	Round Trip Time
SCTP	Stream Control Transmission Protocol
SVP	Synchronous Virtual Pipe
TCP	Transmission Control Protocol
TF	Time Frame
UDP	User Datagram Protocol
UTC	Universal Time Coordinated
VM	Virtual Machine
WAN	Wide Area Network
XCP	Explicit Control Protocol

1

Introduction

Contents

The Internet adoption has significantly grown in the past two decades, becoming essential in our daily live. Despite the tremendous success of the Internet its current architecture is no longer the most suitable for certain applications, such as: Peer-To-Peer (P2P), audio and video streaming. These kind of services or applications require high bandwidth for taking full advantage of its potential. With the current Internet architecture, the desirable performance for this kind of applications can hardly be provided. In addition to the performance issues, the current Internet architecture also lacks security, mobility, scalability, and energy efficiency [7].

Today, more and more users are concerned about the security problems in the current Internet, because they do not want their private information to be compromised while browsing the Internet. In fact, several malicious attacks are performed each day by hackers with the goal of stealing critical information from Internet users. Implementing security natively in a future Internet architecture is of great importance, since it will be more difficult for attackers to perform malicious attacks against users private information or Internet service access.

The principle behind the design of the Internet architecture was that network nodes were not able to change their location, i.e. nodes need to have a fixed or static location. Because of this it was not addressed the possibility of dynamic changes (mobility) in the network topology. This has become a problem, since nowadays more and more users use mobile devices to access the Internet, e.g. for checking their e-mail or their social network updates. Therefore, the future Internet architecture must allow a good integration between mobile devices and fixed devices.

Nowadays, the current Internet routing system may be facing scalability issues, due to *"the ever-increasing user population, as well as multiple other factors including multi-homing, traffic engineering, and policy routing"* [8]. Therefore, new technologies or approaches need to be developed for the future Internet architecture, allowing good scalability for the upcoming applications and services.

Finally, the energy consumption is a major concern nowadays, not only for environmental reasons but also economical ones. This makes the energy efficiency an important subject that must also be addressed in the design of new network architectures. Several research is being made in the green networking area, which focus on bringing energy awareness to the underlying network infrastructure that effectively lacks energy saving solutions. This represents a significant change since most of the research was about reducing the energy consumption in battery powered devices.

1.1 Motivation

The energy consumption is a major concern nowadays due to the global warming effect, which is leading to major climate changes. A recent report from the European Union (EU) estimates a necessary reduction of about 15% - 30% in the emission of Greenhouse Gas (GHG) until 2020, in order to keep the increasing of the temperature below 2% [9].

Even though the energy spent by the Internet may be negligible in comparison with the rest of the society colossal energy consumption, it is important that significant work is made to reduce the energy consumption of the Internet, which will make an important contribution in the reduction of the GHGs emission. According to the report in [10], the energy consumption of Information and Communications Technology (ICT) is about 5% of the total energy consumption of developed countries and that the GHGs originated from ICT will be increased by 130% from 2002 to 2020 (see Figure 1.1).

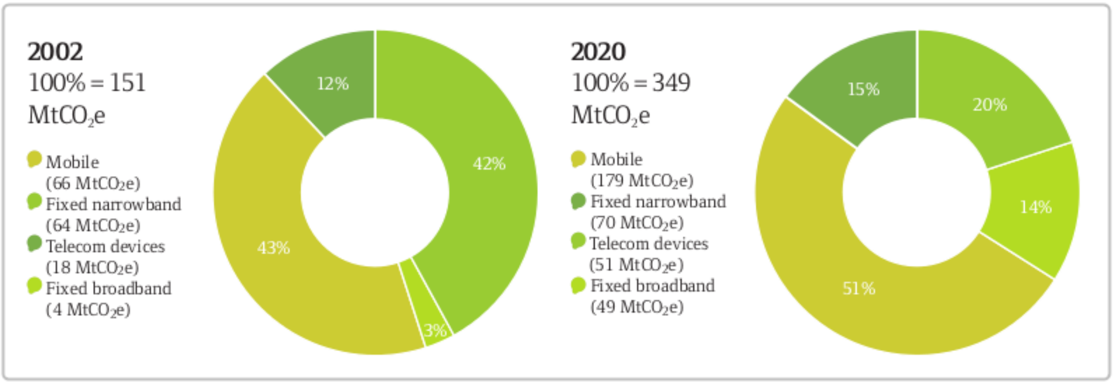


Figure 1.1: Global telecommunications emissions (figure extracted from [10]).

When building a more energy efficient architecture must be taken into account the impact produced in the performance of the network. This is important because many of the current Internet services require high bandwidth, e.g high definition content. Therefore it must be researched and developed efficient energy saving models that can reduce the energy consumption of the current Internet, without producing a major impact in its performance.

The unbounded energy consumption is mostly caused by two main factors. Primarily, the energy consumption does not vary linearly according to the utilization of network nodes and links, which ideally should be zero in case of no utilization. On the other hand, the network nodes are always powered on to maintain the network connectivity at all times. By enabling the network elements to enter in an energy saving mode, it will be possible to greatly reduce the energy consumption when they are idle or underused. This will not only allow a reduction in the emissions of GHG but will also allow a reduction in energy associated costs.

1.2 Contributions

The main focus of this thesis is to provide an energy saving mechanism that can be embedded into the current and future Internet architectures. The proposed solution offers energy efficiency at the expense of a slightly lower network performance. The main contributions of this thesis can be enumerated as follows:

- A brief survey, analysis and comparison of previous work done in the areas of the future Internet and green networking.

- Provides an energy consumption model that estimates the energy consumption of a network.

- Provides a traffic aggregation mechanism, which allows to transfer traffic from underused links to most used ones.

- Provides a ranking mechanism that reflects the importance of a network element in the packet delivery process.

- Finally, it is provided a sleeping mechanism that enables a power saving mode in the network nodes without a huge decrease in the performance of the network.

1.3 System Requirements

The main goals of this work are to evaluate the energy consumption of the current Internet architecture and to develop an energy saving mechanism that can reduce the amount of energy that is necessary to properly route all the traffic in a network. To achieve these goals, the following system requirements were defined for this work.

- Evaluate the current Internet model.

- Evaluate a completely new architecture and its applicability in the future Internet architecture.

- Perform an evaluation of the energy consumption of both systems.

- Design a solution that reduces the overall network energy consumption.

- Evaluate the tradeoff between energy savings and network performance in different scenarios.

1.4 Document Organization

During this chapter it was given an introduction to the problem to be solved and the motivation behind this work. The remaining contents of this dissertation are organized as follows:

In chapter 2 it will be addressed the state of the art in this area, mainly focusing on the Internet architectural problems and on solutions to mitigate them. It will be addressed not only solutions over the current Internet architecture, but also completely new solutions that will not rely on it. Finally, it will also be addressed solutions to reduce the energy consumption of the Internet infrastructure.

In chapter 3 it will be described the implemented modules of the proposed solution and the respective approach taken, in order to reduce the overall network energy consumption.

Chapter 4 explains the experimental evaluation scenarios, goals, and settings.. An overview over the metrics and chosen scenarios will be given, ending with the presentation and discussion of the evaluation results.

In chapter 5 it will be given a brief conclusion and a summary of the results of the proposed solution, being also addressed some directions for further work.

2
State Of The Art

Contents

The main goals of this chapter are to provide an overview of the issues of the current Internet architecture and to identify the contribution of the different proposals for the future Internet architecture. First of all it will be discussed different kinds of wireless networks, technologies and security that must be addressed in future designs. Afterwards it will be discussed some of the proposals designed to improve the Internet architecture. Some of them will choose to use the "Clean Slate Design" [11] and others will only try to improve existing technologies. Finally it will be discussed some techniques that will try to make the Internet architecture more "green", i.e. to make the Internet more energy efficient.

2.1 Technologies for the future Internet

This section has the goal to provide a brief overview about different types of wireless networks, peer-to-peer systems and security issues. These topics address some important concerns that need to be taken into account in tomorrow's Internet architecture, which are: mobility, scalability and security.

2.1.1 Infrastructured Networks

In infrastructured networks the wireless nodes are connected to an Access Point (AP), forming a wireless network. The AP will act like a coordinator, being responsible for forwarding the packets among the wireless nodes that are connected to it. The major advantage of this kind of wireless networks is that forwarding is a lot easier than the one in ad-hoc networks, because there is a central communication coordinator, the AP. Finally, in this kind of networks the AP may become a bottleneck of the network and also when it fails the communication among the nodes is not possible. These are the major downsides when using an infrastructured network [12].

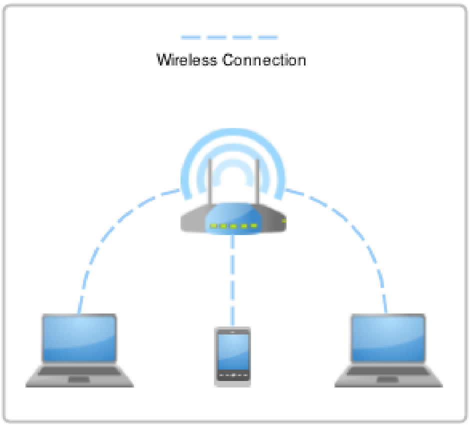


Figure 2.1: Typical infrastructured network.

2.1.2 Ad-Hoc Networks

The ad-hoc networks are constituted by a group of wireless nodes forming a kind of static mesh network, making unnecessary to use a centralized AP. The difference between ad-hoc networks and traditional wireless networks is that ad-hoc networks do not require the use of an established infrastructure. The wireless nodes will communicate with each other by using a multihop approach, in which the nodes that are not able to communicate directly with each other, will make use of the nodes in their neighbourhood to successfully forward the packet to the destination [13].

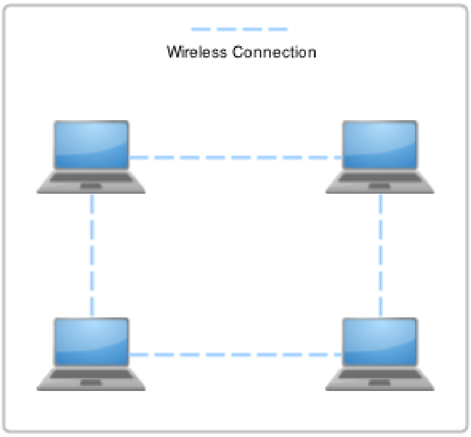


Figure 2.2: Typical ad-hoc network.

2.1.3 Mobile Ad-Hoc Networks (MANETs)

MANETs are constituted by a group of wireless nodes with no fixed network topology, i.e. the topology is changed in real-time due to the constant mobility of the nodes that belong to the network. Like ad-hoc networks, nodes will share network resources in order to successful deliver a packet from a source node to a destination node. Since the nodes are not in a fixed position, traditional ad-hoc routing protocols are no longer applicable. Following, it will be discussed briefly some of the routing protocols for MANETs, such as [14]:

- *Destination-Sequenced Distance-Vector (DSDV):* The routing information is exchanged periodically in the network in order to maintain the routes between nodes. Each node maintains a table telling the number of hops to every possible destination [15].

- *Ad hoc On-Demand Distance Vector (AODV):* Using AODV [16] the routing information will only be exchanged in order to establish routes between nodes that are generating traffic.

- *Anchor Based Routing Protocol (ABRP):* This protocol combines routing tables with geographic routing. It will use the geographic location of the mobile node to define the routes that will be used by the mobile nodes [14].

2.1.4 P2P

The P2P traffic is responsible for a huge part of today's overall Internet traffic, overcoming web traffic. This way implementing P2P natively in the future Internet architecture may be profitable. The idea behind a P2P system is to share the available resources among the participants of the

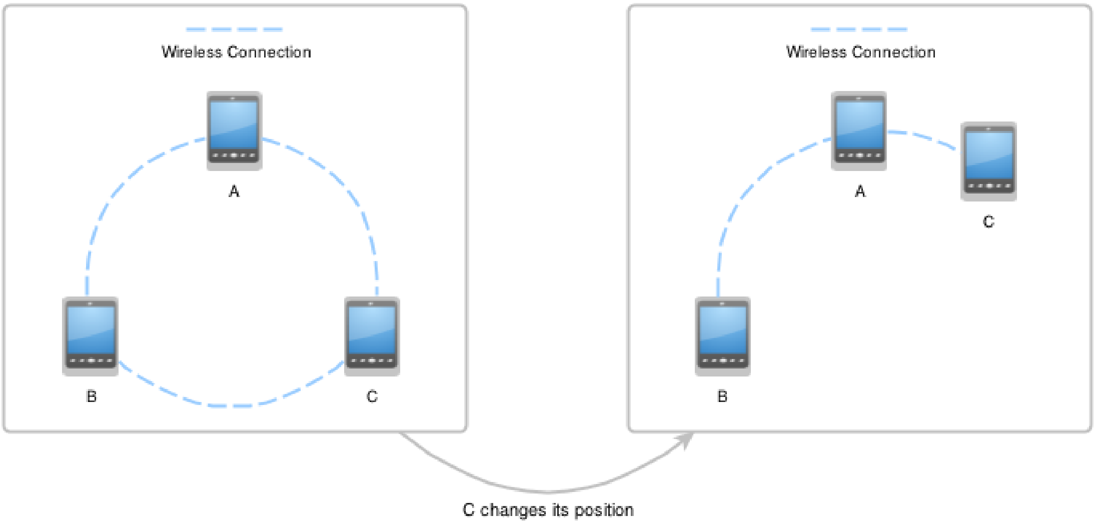


Figure 2.3: Typical MANET.

system in order to maximize performance. Also, the P2P systems are typically decentralized, fault tolerant and scalable.

The P2P systems are built on top of traditional IP networks, like the Internet, forming overlay networks. The overlay networks can be classified as: structured or unstructured. In structured overlays there is a kind of virtual topology, where all the nodes are given unique identifiers. Using this kind of overlay, the messages are routed through a logarithmic number of hops. Some of the most relevant solutions for structured networks are: Chord [17], Pastry [18], Tapestry [19] and CAN [20]. On the other hand unstructured networks do not define a virtual topology to represent the nodes in the network. Using unstructured networks the messages will be propagated using flooding or gossiping [21].

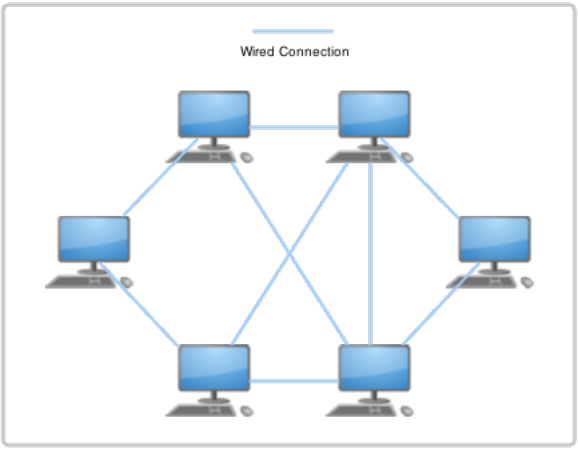


Figure 2.4: Typical decentralized P2P architecture.

2.1.5 Security

The current Internet was designed assuming that all users were truthful and with good intentions. Because of this assumption the current Internet lacks strong security policies. For instance, if no authentication of users or data is performed, an attacker can easily steal information from a user without revealing his identity. This problem as led to the creation of security applications, such as anti-virus and firewalls. Since these kind of applications do not penetrate deeply into the network, attacks are still possible of being successful [22].

Nowadays, most users are very concerned about securing their data, since more and more critical information is put on the Internet. The best example is when a user performs operations over his bank accounts from the Internet, in which he does not want his money to "magically" disappear. But not all people know the Internet dangers, thereby not protecting themselves with firewalls, anti-virus or other security applications. In this sense the network must be capable of offering some level of protection to the people that do not know how they can protect their data from being compromised.

It can be said that security is one of the major concerns that need to be addressed when designing new approaches for the future Internet. In this way, the future Internet implementations must be able to first verify if the source of the information is valid and good intended, before delivering the packet to the destination [23].

2.2 Future Internet Proposals

Despite the tremendous success of the Internet, its current architecture may not be the ideal solution for issues, like: security, mobility, manageability, dependability and scalability [1]. These problems do not have a trivial solution, because it is difficult to address them without increasing the complexity of the architecture. These issues can prevent the achievement of a better performance for some communication technologies, such as fibre optics and radio transmissions [24].

As a consequence of the aforementioned problems, new solutions and even different paradigms are being researched to mitigate them. Following, it will be discussed and analysed some of the solutions for the current and for the future Internet architecture.

2.2.1 4WARD

The Wired and Wireless World Wide Architecture and Design (4WARD) project was created and funded by the EU according to the research and technological development programme, called Seventh Framework Programme (FP7)[1]. The goal of this project *"is to make the development of networks and networked applications faster and easier, leading to both more advanced and more affordable communication services"*[2].

The 4WARD approach is to implement a virtual network over multiple physical infrastructures

[1] Framework Programme for Research and Technological Development. `http://ec.europa.eu/research` (Last access: 06-01-2011)

[2] The FP7 4WARD Project. `http://www.4ward-project.eu` (Last access: 06-01-2011)

achieving some sort of separation between the physical and the logical topology of the network [25]. During research it was followed the "Clean Slate Design", where the architecture is built from scratch taking into account the new requirements that are imposed by today's Internet design issues. The "Clean Slate Design" requires that the new developed system must offer similar functionality as the legacy system, but with improved abstractions and/or performance [11].

The implementation of the 4WARD research into real networks must take a migration approach, since it is not viable to build a completely new Internet architecture from scratch, due to the success of the current Internet. In this sense, the implementation of 4WARD may follow three different approaches, which are: incremental enhancements, utilization of overlay techniques and network virtualization. The incremental approach consists in the development of enhancements or extensions to the existing architecture or protocols with no implications for the current Internet. The other approach is to use overlay techniques which consists on building completely new functionalities above the current architecture. This has been done for example with SIP [26] and P2P applications. The last approach makes use of virtualized networks to allow that a unique physical infrastructure can be used by different networks (sensor networks, enterprise networks, public networks and new network architectures) [24]. Hereafter it will be addressed the two major components of the 4WARD architecture, the generic path and the network virtualization.

The 4WARD connectivity framework makes use of the generic path approach, which is defined as the *"means to organize the accessibility of a sufficient number of parts or copies of information objects stored in a group of hosts"* [1]. The goal of the generic path is to allow for two or more endpoints to communicate with each other, being flexible enough to support the mobility of endpoints. In terms of scalability the generic path can be used from small scale networks, Local Area Networks (LANs), to large scale networks, Wide Area Networks (WANs). A more detailed explanation of the generic path mechanism can be found in [27].

The network virtualization is one of the most important features in the 4WARD project, because it allows different network architectures to coexist with each other, being an important feature for the migration to new evolutionary approaches [1]. The main purpose of virtualization is to easily deploy new communication technologies over an existing physical infrastructure. The 4WARD workgroup proposes a framework that allows the virtualization of individual resources. For example, by virtualizing wireless resources it will be possible to perform a better wireless spectrum resource allocation, thereby achieving a better network efficiency. The developed framework [28] defines four main participants, which are [25]:

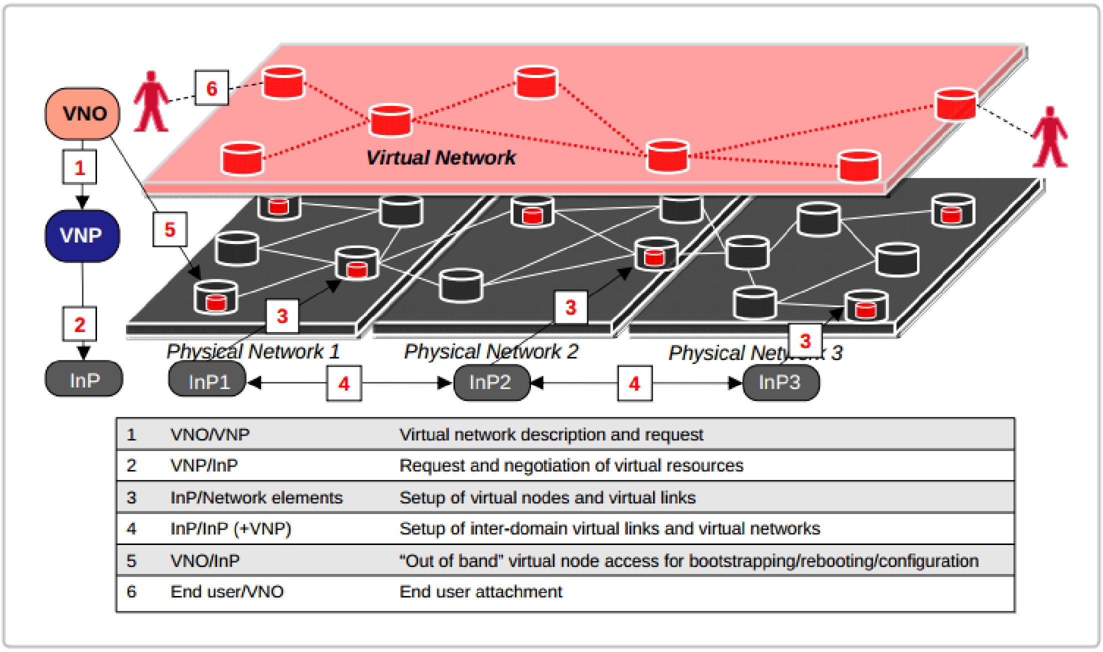
- *Infrastructure Providers (InP):* The infrastructure providers are somewhat similar to the current Internet Service Providers (ISPs), in the sense that they own a physical infrastructure over which they provide virtual resources to be rented. These virtual resources will be used by virtual network providers to build virtual networks.

- *Virtual Network Providers (VNP):* The virtual network providers rent virtual resources, that were made available by infrastructure providers, for the purpose of building virtual networks.

They can move virtual resources, such as virtual nodes, between two physical infrastructures. The virtual networks will be made available to be rented by virtual network operators.

- *Virtual Network Operators (VNO):* The virtual network operators are responsible for operating, managing and monitoring virtual networks. But also to offer interfaces so that end users and service providers can use virtual networks to cover their needs.

- *VNet users:* They are the end users that will use the virtual networks to access legacy or new communication services.



1	VNO/VNP	Virtual network description and request
2	VNP/InP	Request and negotiation of virtual resources
3	InP/Network elements	Setup of virtual nodes and virtual links
4	InP/InP (+VNP)	Setup of inter-domain virtual links and virtual networks
5	VNO/InP	"Out of band" virtual node access for bootstrapping/rebooting/configuration
6	End user/VNO	End user attachment

Figure 2.5: The 4WARD Architecture (figure extracted from the 4WARD website[2]).

The 4WARD project contributes with a completely new design for the future Internet architecture, which is based in the information-centric concept. As just discussed the 4WARD project proposed the use of generic path and network virtualization in its architecture.

2.2.2 ANA

The Autonomic Network Architecture (ANA) project was created and funded by the EU according to the research and technological development programme, called Sixth Framework Programme (FP6)[1]. The expression "Autonomic Networking" stands for self-managing networks without nearly any human intervention. The use of this paradigm has led to the main goal of this project, which *"is to design and develop a novel network architecture that enables flexible, dynamic, and fully autonomic formation of network nodes as well as whole networks"*[3].

The ANA approach avoids building a strict architecture, in which all the communication protocols are imposed by it, thus not allowing for innovation to happen. In this sense, the ANA also uses

[3] The FP6 ANA Project. `http://www.ana-project.org` (Last access: 06-01-2011)

the ”Clean Slate Design” approach, which allows the researchers to be more creative since they do not need to worry about compatibility with the current Internet architecture [2].

The ANA work group defined the necessary network abstractions to allow different networking principles or paradigms to coexist and to offer users a generic way of accessing the services for communication purposes (see Figure 2.6). It was defined the core elements that participate in the system, which led to the following network abstractions [2, 29]:

- *Network compartment and information channel (IC):* The network compartments are a region of the network that have the same addressing, naming, routing, networking mechanisms, protocols, packet formats, etc. Therefore, each network compartment can use any implementation of the aforementioned characteristics. The information channel provides communication services to network compartments and it is used by functional blocks to communicate among each other.

- *Functional Block (FB):* The functional blocks are used for generating, consuming, processing and forwarding information. They are located in the node compartment.

- *Information Dispatch Point (IDP):* The information dispatch points are used to access functional blocks.

- *Node Compartment:* A node compartment is considered a networking node, in which multiple functional blocks are running. In this sense, a node compartment behaves like a network compartment but without providing information channels.

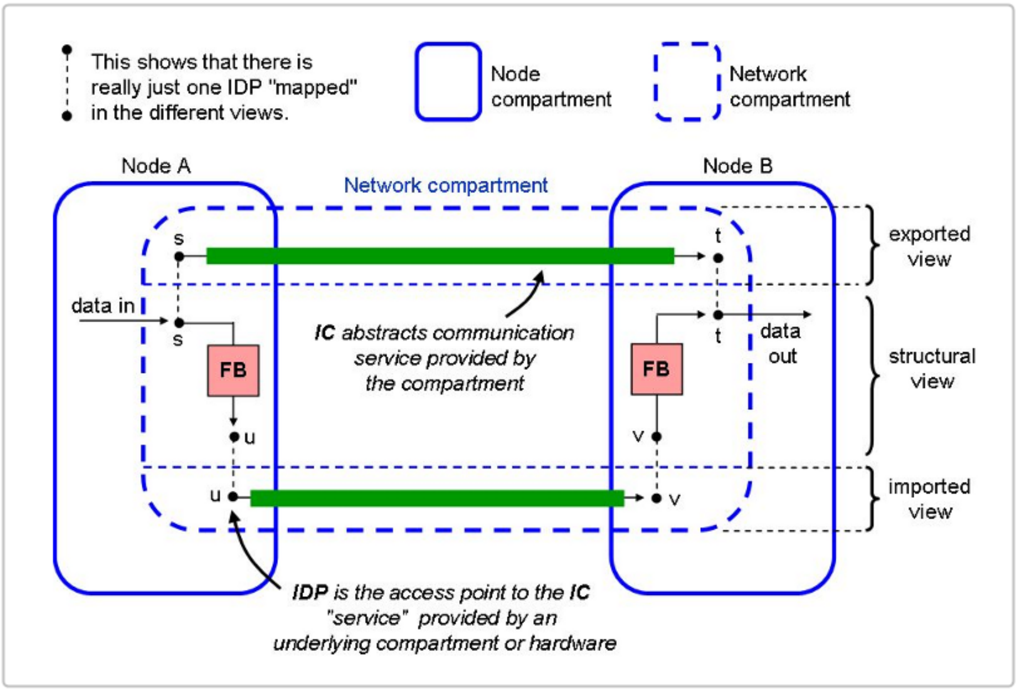

This shows that there is really just one IDP "mapped" in the different views.
Node compartment
Network compartment
Node A
Network compartment
Node B
s
t
exported view
data in
s
t
data out
IC abstracts communication service provided by the compartment
FB
FB
structural view
u
v
u
v
imported view
IDP is the access point to the IC "service" provided by an underlying compartment or hardware

Figure 2.6: The ANA network abstractions (figure extracted from [2]).

From the communication point of view the ANA follows the model publish/resolve where a service is published in a network compartment for further resolution, which will lead to an information channel identified by an information dispatch point. This information channel will allow to send and receive data to/from the resolved service. It is allowed to lookup for services in order to obtain information about them without creating an information channel, e.g. like a DNS lookup [30]. When an information channel is no longer necessary, the corresponding information dispatch point can be released from his duty to save some resources [2].

The ANA approach is very useful for the future Internet since it supports network self management and self optimization. Besides this, it provides good flexibility in terms of deployment and utilization of different networking schemes and protocols. Last but not least it provides good mobility support, allowing a better connectivity and performance when moving between different networks, e.g. wireless networks [31].

2.2.3 FARA

The Forwarding directive, Association, and Rendezvous Architecture (FARA) is part of the New Arch project[4]. The goal of this project is to offer *"an abstract high-level model for the network architecture, which is based upon decoupling of end system names from network addresses"*[5].

Nowadays, the IP addresses are used for identifying both networks and communication points, which provides some security but at the cost of mobility. In this sense, the FARA proposes a solution for solving this problem without the creation of a new identifier name space. This way it is possible to separate entities from their respective location, which offers better support for entity mobility [4, 32].

The key elements of FARA that participate in the communication process are: entities, associations and communication substrate. Using these elements, the FARA defines the communication process as the exchange of data packets over a communication substrate, which is performed between entities using logical links (associations). Following, it will be addressed the FARA key elements in a more detailed way [32].

- *Entities:* An entity represents the end-point of the communication network, which could be a single process, multiple processes, a thread, a machine, etc.

- *Associations:* An association is a logical communication channel, which allows to send and receive data packets from FARA entities. These associations replace the IP and port pair as the destination identity. Also each packet will be associated with only one association, which will be local to entities. The associations are end-to-end which will make them invisible to routers. It will be used an Association Identifier (AId) to uniquely identify the associations of an entity.

- *Communication Substrate:* For the communication process it is used a connection less scheme to deliver packets using the appropriate addressing and routing. It supports different deliver

[4] NewArch Project: Future-Generation Internet Architecture. `http://www.isi.edu/newarch/` (Last access: 06-01-2011)

[5] The FARA project. `http://www.isi.edu/newarch/fara.html` (Last access: 06-01-2011)

mechanisms, such as: hop to hop delivery, explicit routing and label swapping. The disadvantage in using a connectionless scheme is its lack of reliability. The FARA entities are responsible for offering this reliability. When an entity needs to send a packet to one of its associations, it will first add the destination Forward Directive (FD) to the packet header. The destination FD contains all the information needed to deliver the packet successively to the destination. After that, the entity will hand over the newly created packet to its communication substrate to be sent. Finally, if necessary a reply forward directive may be defined to allow a response packet to be sent to the source.

The main contribution of FARA to the future Internet architecture is the separation between associations and the forward directives. This will give entities the ability to move, since the forward directive may be changed.

2.2.4 NIRA

The New Internet Routing Architecture (NIRA) was designed to allow users the possibility to choose their own domain-level routes. A domain-level route is described as the domains that the packet needs to pass until it reaches its destination, differing from router-level route which is described as the routers that forward the packet to the destination [32]. To provide support for user-controlled routes the design of NIRA needs to address the following issues [5]:

- *How does a user discover a failure free route?*
- *How is a route encoded in a packet header?*
- *How do users pay for ISP's service?*

The NIRA design addressed these issues in different modules, allowing an easy adaptation for future evolution and enhancements. Following, it will be explained the different modules of the New Internet Routing Architecture [5].

- *Route Discovery:* To establish a connection between two hosts a route needs to be set, which represents the path that the packet needs to travel between the sender and the receiver. The creation of a valid route implies that the user must know the routes he can use and if they are failure free. The route discovery is performed by using the Topology Information Propagation Protocol (TIPP) [5], which is composed by two components, the path-vector and the link-state components. The path-vector is responsible for giving the user the direct and indirect service providers and the link-state is responsible for informing the user of current network conditions. The TIPP does not select the path among the possibilities chosen by the path-vector component. Instead he sends that information to the user which will choose the best path.
- *Efficient Route Representation:* When a sender learns an end-to-end route, he will need to encode that information in the packet header. In NIRA it is used a provider-rooted hierarchical

address that represents the route between the user and its core provider. Using this address scheme the user can use the source and destination addresses to represent a valley-free route [33]. They are also used for packet forwarding which limits the possibility of source address spoofing, since the router may not know the source address that was imposed by an attacker. This scheme needs an address which is long enough to represent the provider hierarchy, which in this case it was chosen the IPv6 addressing scheme [34].

- *Bootstrap a Communication:* To bootstrap a communication, the sender needs to know the routes that the destination can use. In the NIRA design it was proposed an infrastructure service, Name-to-Route Lookup Service (NRLS), which maps the destination name to the possible routes that he is allowed to use. This design does not constrain the name space used nor the implementation of the NRLS. The name space can be hierarchical (e.g. DNS [30]), flat [35] or a completely new scheme that may be proposed for the future Internet. When a user wants to communicate with another one, he needs to query the NRLS so that he can obtain the route information of the destination. After that, the user combines his route information with the route information of the destination, choosing the source and destination address to reach the final destination of the packet.

- *Handling Route Failures:* To avoid failures, the NIRA design has a mechanism for discovering route failures, which is based on a combination of proactive and reactive notifications. The TIPP proactively notifies the user of the current conditions of his own routes. This kind of notifications can not be fully trusted because the TIPP messages may not be propagated globally. In this situation, the user will rely on reactive notifications, such as timeout.

- *Forwarding:* In order for a packet to be successfully delivered to its destination, it must primarily be forwarded to all the domains of the source address. After that, the packet will be forwarded to all the domains of the destination address. The top-level providers that belong to the core will choose the path that the packet will follow between the providers of the source and the providers of the destination.

The main contribution of NIRA is to give users the possibility of choosing their domain-level routes without the need to run a global link-state routing protocol. Besides this, it can offer some security because it reduces the possibility of source address spoofing [5].

2.2.5 PSIRP

The Publish-Subscribe Internetworking Routing Paradigm (PSIRP) project was created and funded by the EU according to the research and technological development programme, called FP7[1]. The goal of this project *"is to develop, implement, and validate an information-centric inter-networking architecture based on the publish-subscribe paradigm."*[6].

The current Internet is message oriented, where the communication responsibility is given to the sender and the network will only try to guarantee that the message sent will reach the receiver. This

[6] The FP7 PSIRP Project. `http://www.psirp.org/home` (Last access: 06-01-2011)

approach brings an important security problem, which is giving the full control of the communication to the sender allowing him to easily perform malicious attacks.

The PSIRP approach uses the publish-subscribe paradigm, whose architecture is based in the information and not in the network nodes. This way the receivers have full control of the information that they want to consume. There is a lot of systems that implement the publish-subscribe paradigm, but none of them actually changed the network architecture, since they create an overlay network above the current Internet architecture. This way the PSIRP project aims to design a publish-subscribe architecture that does not rely on the current Internet and makes performance, efficiency and security, key aspects of its architecture.

Most publish-subscribe architectures are composed of three major components, which are: publishers, subscribers and routing nodes (brokers). The publishers are responsible for feeding the network with information to be consumed, i.e. publications. The subscribers are the consumers of information by expressing their interest on some published items using subscription messages. The brokers are responsible for forwarding the data between the publishers and the subscribers by matching the subscriptions with the information published. So the brokers or Rendezvous Points (RPs) have the responsibility to route, forward and allowing the delivery of data from publishers to subscribers. Using this kind of architecture the publishers and subscribers do not need to be aware of the existence of each other [36].

In the PSIRP architecture *"information is everything and everything is information"* [37]. The information is organized hierarchically, ranging from small pieces of data to large documents, audio or video files. As said before the information becomes available for consuming through publications, which are uniquely identified by a pair of identifiers, the rendezvous identifier (RId) and the scope identifier (SId). These publications are arranged into networks, called scopes, which may refer to physical (e.g. IST network) or logical networks (e.g. Facebook) and are identified in a unique way by the SId. The RId has to be unique inside the scope which it belongs to. Not only the scopes are able to locate information, but are also able to offer policies for access control. It will be possible to define the necessary permissions for accessing or publishing information items within a scope (see Figure 2.7). The use of both identifiers, RId and SId, is appropriate for information oriented architectures since they are endpoint independent, which embeds mobility, multicasting and multihoming into the architecture [36, 38].

The PSIRP architecture defined four major operations, which are: topology management, routing, rendezvous and forwarding. As said before the rendezvous is responsible for finding publications in the network and matching them with subscriptions. The topology management module is used for detecting and reacting to changes in the network topology. Also in the same module, the routing function is used to create the delivery paths between the publishers and the subscribers. The last function, forwarding, is responsible for forwarding the information along the delivery paths created by the routing function [38].

The publisher, in order to publish information items in the network needs to know the SId of the scope within which he wants his data to be published. Also the publisher needs to create the RId,

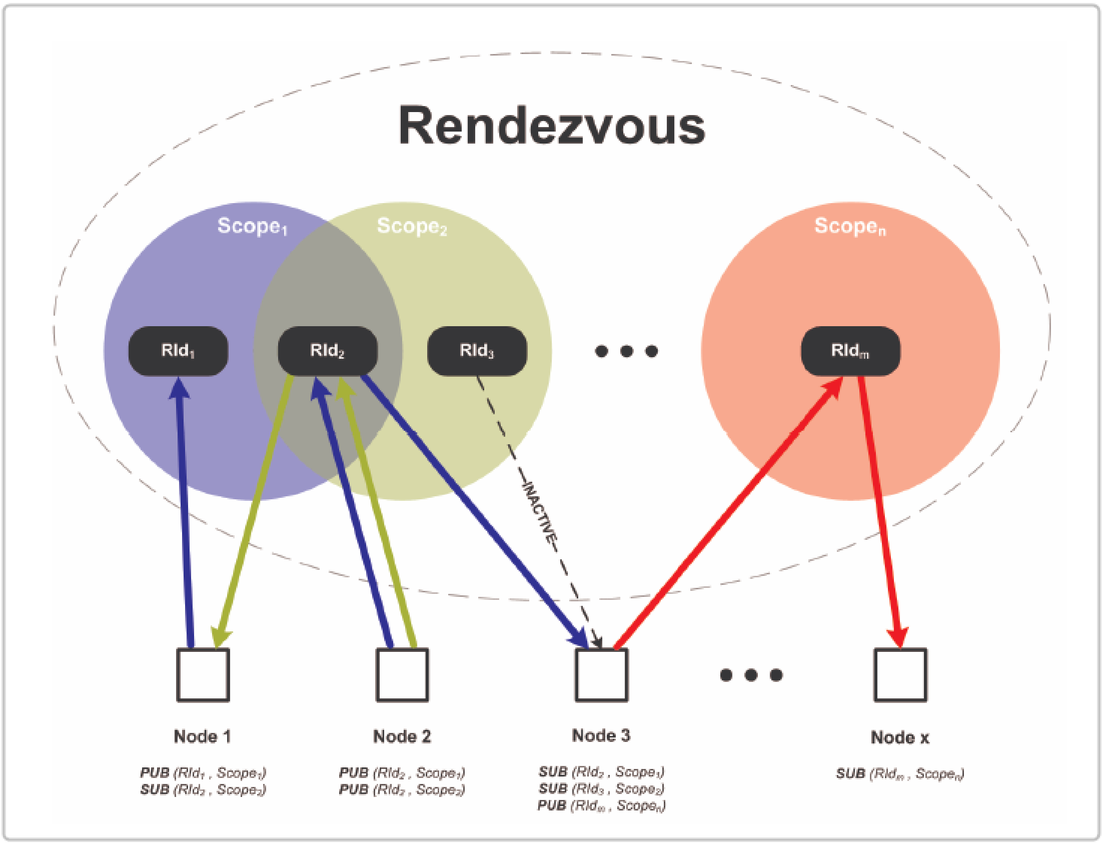

Rendezvous
Scope_1
Scope_2
Scope_n
RId_1
RId_2
RId_3
RId_m
INACTIVE
Node 1
Node 2
Node 3
Node x
PUB (RId_1 , Scope_1)
SUB (RId_2 , Scope_2)
PUB (RId_2 , Scope_1)
PUB (RId_2 , Scope_2)
SUB (RId_2 , Scope_1)
SUB (RId_3 , Scope_2)
PUB (RId_m , Scope_n)
SUB (RId_m , Scope_n)

Figure 2.7: The PSIRP Architecture (figure extracted from [37]).

by typically applying an hash function over his publication. After these steps the publication will be forwarded to the respective rendezvous node within the SId, which is responsible for managing the publications with a specific RId. The rendezvous point only stores the metadata of the publication, which may contain the author, the size or even a brief description of the publication [36].

Similar to the procedure of the publisher, the subscriber must first know the scope identifier and the rendezvous identifier in order to access the information of a specific publication. When a subscriber wants to express his interest on a particular publication, he sends a subscription message to the responsible rendezvous point that belongs to the scope identified by the SId. After receiving the subscription message, the rendezvous point will match it against the stored publications and will forward the publication from the publisher to the subscriber. The forward path created by the rendezvous point between the publisher and the subscriber follows a similar approach as Multiprotocol Label Switching (MPLS) [39]. So for each active publication it will be assigned a forwarding identifier (FId) defining the path that will be followed by the publication [36, 38].

The PSIRP project follows the publish-subscribe networking paradigm, which is a good approach for the future Internet, since it can provide good flexibility, scalability and security. Also due to the increased usage of mobile devices one of the issues that must be addressed by future architectures is mobility. The PSIRP architecture makes the implementation for supporting mobility a lot easier, since it does not depend on the IP protocol and it is not data oriented. This way there is a complete separation between communication and location of the user inside the network. Hence, the user will be able to change his location without losing connectivity.

2.2.6 XCP

The Explicit Control Protocol (XCP) is a window-based protocol, like Transmission Control Protocol (TCP) and Stream Control Transmission Protocol (SCTP), which implements congestion control at the endpoints of a connection, offering high end-to-end throughput. The TCP protocol is commonly used in the Internet for congestion control, but it is not capable of offering high throughput since it is inversely proportional to the packet drop rate. So the packet error rate limits the TCP throughput even in the cases of low error rate links, like fiber, causing a serious limitation to users that nowadays use very fast Internet connections. For these reasons it is needed a new congestion control protocol that can provide better performance than TCP in conventional environments and that can still be efficient, fair, and stable when the delay of the communication increases [32, 40].

The XCP is a generalization of the Explicit Congestion Notification (ECN) [41]. The XCP stores the per-flow state in the packet header, called congestion header, offering good scalability for a different number of flows because the state information does not need to be stored in the routers. In this case the routers will notify the senders about the current congestion level of the network and will be required to perform aggregation operations. Also, it implements a new concept which is the separation between the fairness control and the utilization control. To allow fairness, the XCP will obtain bandwidth from flows whose rate is higher than their fair part, giving it to other flows that need to increase their bandwidth upto the part they are entitled. To perform good utilization of the available network resources, the XCP automatically adjusts its aggressiveness according to the remaining bandwidth and the feedback delay, providing stability in the case of high bandwidth or large delay [32].

To achieve this operation, the XCP maintains an estimate of the Round Trip Time (RTT) and a congestion window, *cwnd*, for the pending packets. So when a packet is to be sent, the sender attaches a congestion header with his current window and RTT. When sending the first packet of a flow, the RTT is set to zero to let the routers know that the source does not have a current estimation of the RTT. The sender needs to initialize a field, called feedback, requesting the desired congestion window increase. The desired increase for the congestion window is calculated using Equation 2.1, where r is the desired rate. In this case it is possible for the sender to achieve the desired rate in only one RTT, if there is enough bandwidth available. When the sender receives an acknowledgement, he will adjust his congestion window according to the received feedback, applying the Equation 2.2. Finally the XCP receiver only differs from the TCP one, since he copies the congestion header in the received packet to the acknowledgement packet to be sent to the source of the original packet [6].

$$increase = \frac{r * RTT - cwnd}{number\ of\ packets\ in\ cwnd} \tag{2.1}$$

$$cwnd = max(cwnd + increase, packetsize) \tag{2.2}$$

The main contribution of the XCP is to allow very large per-flow throughput in comparison with the current TCP. Also the XCP can improve the overall performance of the network by reducing the packet drop rate, increasing utilization, decreasing delay and achieving fairness more quickly. Last but not least, the XCP flexible architecture makes it easier to implement Quality of Service (QoS), offering the possibility to differentiate packet flows and providing proportional resource allocation [32].

2.2.7 Summary

The Internet architecture needs to be greatly enhanced to allow the emergence of new services and applications. The reviewed proposals try to outcome the major concerns about the current Internet architecture. In Table 2.1 is presented a summary of the issues addressed by each proposal.

Proposal	Clean Slate	Congestion	Mobility	Routing	Scalability	Security
4WARD	✓	✓	✓	✓	✓	✓
ANA	✓	✗	✓	✓	✓	✓
FARA	✗	✗	✓	✓	✗	✗
NIRA	✗	✗	✗	✓	✗	✗
PSIRP	✓	✓	✓	✓	✓	✓
XCP	✗	✓	✗	✗	✗	✗

Table 2.1: Future Internet proposals comparison.

2.3 Sustainable Internet Technology

The constant growth of the Internet for several years resulted in a significant increase of the amount of energy required to operate all the network devices, which may be working all day long. This huge energy consumption has become problematic, since the world environmental conditions are becoming more and more unpredictable due to the emission of GHGs to the atmosphere. This leads to the need of finding good energy saving solutions, not only to reduce environmental damages but also to reduce the associated energy costs [42, 43].

Only nowadays the energy consumption has become a priority problem to be solved in future Internet architectures, due to the rapid growth of energy, costs, costumers, broadband accesses and other services offered by the ISPs.

The energy efficiency is a problem that will affect both wired networks and service infrastructures. This is highly dependent on the arrival of new services, because of the traffic increase that may be originated by them [44]. Next it will be discussed some of the work that is being done in the energy efficiency field for the future Internet architecture.

2.3.1 Power Management and Network Design

In legacy networks, energy consumption was not a major concern, not being important enough to be addressed in their design. The major concerns of those systems were mainly: reliability, cost-effectiveness, robustness, service quality and service availability. With the increase of data traffic and new applications, the Internet is consuming more and more energy. To prevent the increasing of the energy consumption it is important to explore new solutions that will allow a better energy management. Hereafter it will be discussed some energy saving solutions [43]:

- *Energy Saving Mode:* The idea of this solution is to put equipments to sleep, since there is no need to waste energy when the equipment is not actually being used. This way it is a good energy saving mechanism to put equipments to sleep when they are idle. This can be done at different levels, which are: at individual level, where switches, routers or other devices are put to sleep; at network level, combining sleep with routing changes and the use of bandwidth aggregation, so that when in low activity only the idle equipments are put to sleep; finally, at Internet level this can be done by changing the network topology, allowing the adaptation of routes to different network loads.

- *Adaptive Link Rate (ALR):* In this approach, the link rate will be dynamically changed according to its utilization. This is done by exploiting the variable periods of idleness between consecutive burst of packets. This way the equipments have the ability to dynamically reduce the link rate, because of lack of utilization. This technique is being adopted by IEEE Energy Efficient Ethernet (EEE)[7] [45].

- *System Redesign:* The idea behind this concept is to design new network architectures and protocols, taking into account the energy consumption constraint. Embedding energy saving mechanisms directly in new architectures has a tremendous impact in reducing the energy consumption. The design of new architectures and protocols must satisfy capacity needs for different network users. One idea may pass by limiting the packet processing that needs more energy to only a group of routers and the creation of new data link and routing protocols that are able to work in on-off networks [43].

- *Reliability and energy consumption:* In [46] it is explored the relationship between reliability and energy consumption. It is defined a tradeoff model between power utilization and performance of the network. In this case reliability and power saving are deeply addressed in this model, with the goal of developing robust and energy efficient networks.

- *Optical technology:* Nowadays, the optical technology is widely used in the backbone of the ISPs networks. The developments in this kind of technology, will lead to all-optical networks that will eliminate the need of optical converters, resulting in an overall reduction of the energy consumption [47].

- *Advanced CMOS technology and superconductors:* The other approach is to develop smaller chips that consume less energy, using CMOS technology and superconductors. The 45 nm architecture in comparison with the 65 nm, is able to reduce energy consumption by around 40% [48].

[7] IEEE 802.3 Energy Efficient Ethernet Study Group. `http://grouper.ieee.org/groups/802/3/eee_study/index.html` (Last access: 13-03-2012)

2.3.2 Virtualization

The current paradigm used by ISPs is to run a single application in one server, due to simplicity reasons, resulting in high resource waste and consequently a lot of energy waste. Using virtualization it is possible to run multiple applications in a smaller number of machines, reducing the necessary hardware to execute those applications. The less hardware is used, less energy will be required to operate that hardware.

Virtualization may be used not only in the server point of view, but also at other levels such as storage, network, platform, application and resource. For instance using server virtualization, the physical server will be separated into multiple virtual servers. This can be achieved using different approaches, such as: Virtual Machine (VM); Paravirtualization or Operating System Virtualization [43].

When using VM or full virtualization technology, multiple VMs will share the same physical machine, called host machine. It is the host operating system or VM monitor that is responsible to allocate the necessary resources for running the VMs. Each VM runs its services on top of a guest operating system, which provides the necessary abstractions for file access and network support for their running applications. This way a VM system may run different VMs with different operating systems, giving the users the flexibility to create, copy, save, read, modify, share, migrate and even roll back execution state of the VM (see Figure 2.8) [49]. For instance the possibility of replicating the same VM image in different hosts in an easy way makes the life for system administrators a lot easier.

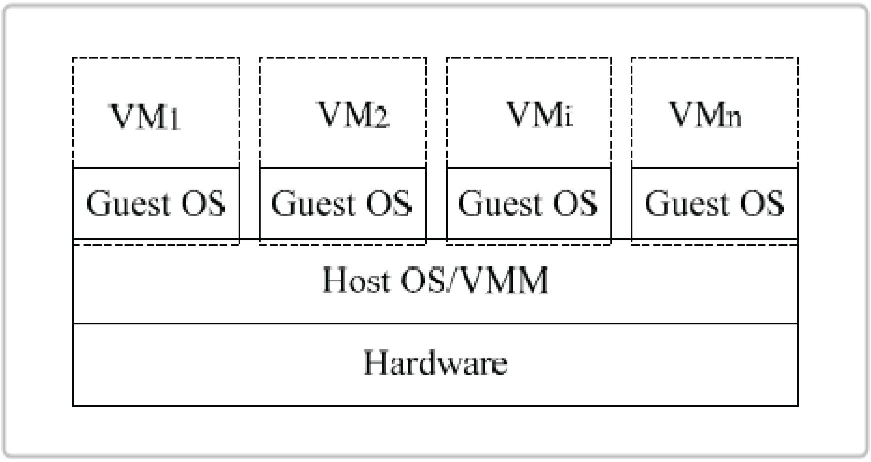


Figure 2.8: The VM Architecture (figure extracted from [49]).

The Paravirtualization technology is used to reduce the performance issues of the full virtualization, since it does not replicate entirely the original guest running environment. In this case the guest operating system must be modified to be able to run in the paravirtualized environment, redirecting all virtualization-sensitive operations to the VM monitor [50]. There is a frontend driver that handles all the guests i/o requests and delivers them to the backend driver, which will interpret these requests and makes a correspondence with the desired physical device (see Figure 2.9) [51].

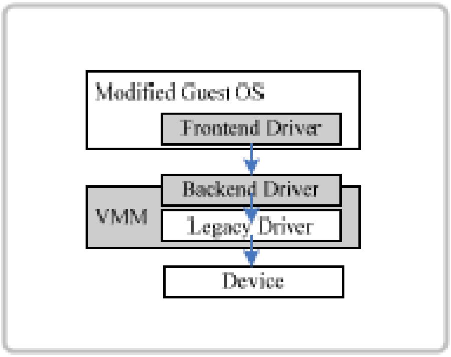


Figure 2.9: The Paravirtualization Architecture (figure extracted from [50]).

The Operating System Virtualization consists on a single operating system kernel running on a server. All the guest environments can only use this specific operating system (see Figure 2.10). On the other hand the networking virtualization will use all the available resources and functionalities, combining them into a virtual network or even sub dividing them into virtual networks. This will allow to optimize the resource utilization of network equipments in order to reduce their energy consumption.

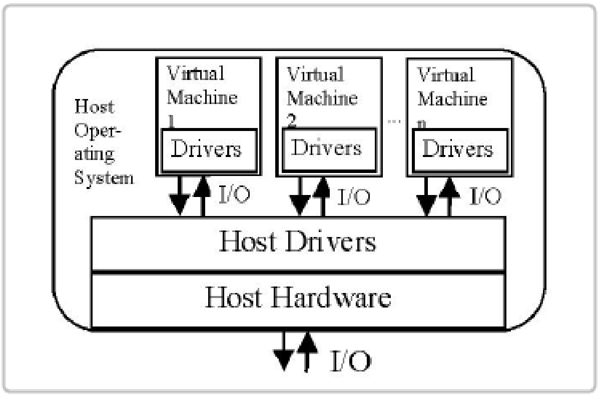


Figure 2.10: The Operating System Virtualization Architecture (figure extracted from [52]).

The use of virtualization in future Internet architectures can play a big role in the energy saving field. There is still the need to evaluate which type of virtualization will allow a better energy management [43].

2.3.3 Pipeline Forwarding

The pipeline forwarding mechanism is a packet-scheduling technique that combines simplicity and effectiveness using a global Common Time Reference (CTR), in order to perform network traffic shaping. It does not need a large amount of network resources and offers good performance. It is also capable of offering QoS and good scalability [53, 54]. The pipeline forwarding is used in some architectures, which are designed to reduce the overall network energy consumption in the future Internet, e.g. the Greener Internet proposed in [42].

Using this technique, switches will be synchronized through the utilization of a time period, Time Frame (TF), which can be assumed as a sort of virtual container for IP packets. The duration of the TF can be obtained by using external sources, e.g. get the Universal Time Coordinated (UTC) from GPS or Galileo positioning systems, or it can also be distributed throughout the network. To allow QoS, the transmission capacity can be partially or totally allocated to one or more flows during the resource allocation period [42]. The pipeline forwarding behaviour is managed by two simple rules, which are:

1. The packets that will be sent in the TF t by some node n, must be put in their output ports buffer in the TF $t-1$.

2. When a packet p is transmitted in the TF t by a node n, it must be also transmitted by the node $n+1$ in the TF $t+d_p$, where d_p is the forwarding delay.

The forwarding delay is calculated during the resource allocation period, which involves scheduling techniques. The pipeline forwarding uses a predefine schedule, Synchronous Virtual Pipe (SVP), for forwarding a pre-allocated number of bytes during one or more TFs along a path of subsequent UTC based switches [42]. There are two implementations of the pipeline forwarding, which are [42]:

- *Time-driven switching:* Using this technique all the packets belonging to the same TF will be switched to the same output port. Therefore, it will not be necessary to perform header processing, resulting in low complexity and possible optical implementation.

- *Time-driven priority:* This technique is suitable for optical backbones, arranging the traffic in large capacity SVPs that are handled by high speed switches. If more flexibility is necessary, the time-driven priority will combine pipeline forwarding with IP routing. This way, packets that enter in the same switch input port during the same TF can be sent to different output ports, according to the established rules in the IP routing.

2.3.4 Selectively Connected End Systems

The selectively connected end systems can manage their own network connectivity in response to internal or external events. This way it is possible for them to predict changes in connectivity, reacting in accordance. For example, the end systems may predict the loss of connectivity just by knowing that they are moving to an area that has few layer-two connectivity. So using selective connectivity will allow hosts to go to sleep, achieving a substantial power saving without sacrificing their place in the network.

In terms of power management the end system may have three different states, which are: on, off and sleep. Analogously to networking, end systems will be characterized by having connectivity, no connectivity or operating in selectively connected mode. The end system can enter in the sleep state without loosing its place in the network. Occasionally, it may be required that an end system in the sleep state to go back on, in order to perform some specific tasks. Following it will be discussed the architectural concepts and components of this solution [55].

- *Assistants:* An assistant is a generic mechanism which helps the host while he is in sleep mode, by performing the routine operations that normally are assigned to end systems. For instance the assistant will allow the host to keep his connectivity by responding to keep-alive messages on his behalf.

- *Exposing Selective Connectivity:* It is important for the host to expose his level of connectivity throughout the different layers of his protocol stack and to inform possible peers with which he may want to communicate. For reasons concerning the energy management, it is important for end systems to know each others state. For example, an active end system may be induced to enter in sleep mode when he wants to communicate with a sleeping end system.

- *Evolving Soft State:* There is the need to evolve soft state, since it is difficult to renew the state for sleeping end systems. There is two solutions for resolving this problem, which are:

  - *Proxyable State* - Using this state it will be the assistant who will be responsible for managing the soft state.

  - *Limbo State* - This state is in the middle between soft state and statelessness. Soft state assumes that a host is not available, but there is the need to know if the host is completely turned off or only sleeping. This way when the renovation of the state expires, the host will enter in the limbo state allowing only the necessary information, used for distinguishing the two states, to be exchanged among the participants.

- *Host-based Control:* The end system has control of how the other ones in the network will react to his selective connectivity. When a host moves to the selective connected mode it is necessary to delegate his tasks to the other participants.

2.3.5 Ranking Network Elements

In order to efficiently choose which network elements to be turned off, it is important to rank each one according to its importance in the network. This can be done by looking to the network topology or to the traffic volume passing through the network element.

The most widely used topology based rankings are: Degree centrality; Betweenness centrality; Closeness centrality; Eigenvector centrality. The Degree centrality is defined as the number of links connected to each node. The Betweenness centrality represents the number of shortest paths in which a node participates. The Closeness centrality gives the average distance between a node and all the other ones, in which the more critical nodes are the ones with the lowest closeness centrality. Lastly, the Eigenvector centrality corresponds to the influence of a node in the network by taking into account the importance level of its neighbours [56].

The traffic volume based rankings only take into consideration the amount of traffic that is routed by the network elements. An example of the application of this principle is presented in [57] and can be called Load. In Table 2.2 it is presented a summary of the characteristics of the different rankings that were described.

Ranking	Topology Aware	Traffic Aware
Degree Centrality	✓	✗
Betweenness Centrality	✓	✗
Closeness Centrality	✓	✗
Eigenvector Centrality	✓	✗
Load	✗	✓

Table 2.2: Summary of the different types of rankings.

2.3.6 Energy Saving Models

In order to achieve an important reduction in the energy consumption of networks, it must be explored the possibility of making routing and traffic engineering decisions based on the utilization and criticality of the network elements. This way, it is possible to achieve a reduction of the overall energy consumption of the network by dynamically turning off network nodes and links when their resources are not required. Hereafter, it will be discussed some solutions based on the aforementioned concepts.

2.3.6.A Dynamic Link Metric

The basic idea of the algorithm presented in [58] is to aggregate traffic to the most used links. The links with no traffic load will be turned off, allowing some energy savings. Also, it will be defined a threshold to avoid traffic congestion in a link by restraining the allowed amount of traffic that may pass in it. A link will be considered to be congested when its traffic load exceeds the threshold, making it necessary to switch back on some other link to carry the remaining traffic. This will be achieved by dynamically changing the weight of the link, transferring the traffic load to most commonly used links. The weight of the link is modified according to Equation 2.3.

$$weight' = \begin{cases} k * weight, & load \leq threshold \\ weight/k, & load > threshold \end{cases} \tag{2.3}$$

The previous equation will update the weight of the link based on the traffic load, the desired threshold and a configuration coefficient k. When the traffic load exceeds the threshold, the weight of the link will be raised in order to reduce its traffic load. When the traffic load is below the threshold, the weight of the link will be decreased in order to increase its traffic load. In this case, it will only be decreased the weight of the link with the higher utilization. The changes made to the weight of the links must be communicated to all nodes in the network topology. Finally, to power off a link it must be taken into account if the link has no traffic load and if the network remains fully connective without the link.

This approach explores the redundancy in the core networks to encounter the minimum set of links that need to be power on in order to successfully route all the traffic, allowing a reduction in energy consumption by powering-off the unused links. The downside of this approach is the decrease of network performance, especially in high-peak traffic hours, due to the increase of the packet delay.

2.3.6.B Green Open Shortest Path First Protocol

In [59] it is proposed a solution that uses the topological information advertised by routers using the Open Shortest Path First (OSPF) protocol. The focus of this study is on making the OSPF protocol more "green", i.e. energy aware. The OSPF protocol specifies that each router computes its own Shortest Path Tree (SPT) by applying the Dijkstra algorithm. Hence only the links that belong to at least one SPT will be used to route data in the network.

The proposed algorithm, Energy-Aware Routing (EAR), defines two sets of routers, the exporters and the importers. The exporters are responsible for computing the shortest routing paths and the importers will then compute their own SPTs based on the SPTs calculated by the exporters, selecting the routing paths to be used. This way it will be possible to reduce the number of links used for routing traffic. The EAR algorithm is composed of three phases, which are:

1. *Exporter Router (ER) selection:* As previously mentioned the ERs will be responsible for computing their SPT by applying the Dijkstra algorithm. The neighbours, Importer Routers (IRs), of the selected ERs will use these SPTs to identify possible links that can be switched off. The selection of the ERs is based on the information contained in the LSA database of each router. With this information the routers with the highest degree will be selected. It will only be selected the routers which are not neighbours of another ER.

2. *Modified Path Tree (MPT) evaluation:* In this phase it will be determined the links to be switched off, according to the MPTs computed by the IRs. The IRs will use a modified version of the Dijkstra algorithm, in which the root node will be his associated ER, instead of the node itself. The SPT computed by the IR will be the same as the one computed by his ER. After this the IR will insert itself as root in the computed routing tree, creating a new routing tree called Modified Path Tree (MPT).

3. *Routing path optimization:* After the completion of phase 2, each IR will have a list of links to be switched off. Turning off these links will generate a new network topology which must be propagated to all network nodes. So the IRs with at least one link to be switch off must send LSA messages to the network. At the end of this procedure each network router will have the current information about the network topology.

The EAR algorithm is a solution that addresses the problem of energy efficiency in today's IP networks without taking into account QoS constraints. The main advantage of EAR is the full compatibility with the OSPF protocol, allowing energy to be saved in low traffic periods. It is still being researched the best criteria for the selection of the exporter routers and its respective number to avoid network congestion, especially in high-peak traffic hours. Hence it is possible to extend this algorithm to take into account the QoS constraints.

2.3.6.C Sleep Coordination in Wired Core Networks

In [60] it is proposed a distributed routing protocol, General Distributed Routing Protocol for Power Saving (GDRP-PS), in which the goal is to put routers into sleep mode without compromising the QoS and network connectivity. This protocol will offer similar operation as other distributed routing protocols in high traffic hours, and in low traffic hours it will put some routers into sleep mode to save energy, taking into account network connectivity and QoS.

In this protocol it is used two types of routers: the power saving routers (PSRs) and the traditional routers. The traditional routers will use the OSPF protocol. These types of routers are always powered on, even when no packets are to be processed. Instead, the PSRs will have two different states: working and sleeping.

In the beginning of the algorithm it is randomly chosen one coordinator, which will record the information about available PSRs and will also be responsible for coordinating the operations of the PSRs. Due to the constant monitoring of the PSRs, the coordinator will never be put to sleep. Furthermore, after a predefined period of time a new coordinator will be randomly chosen, giving the opportunity for all the PSRs to be coordinators (fairness).

To change from the working state to the sleeping state, the PSR must detect that the network is idle by measuring the maximum utilization of all the links that are connected to it, U_{max}. A network is considered to be idle if the U_{max} is below a determined threshold, T_1. If the network is idle then the PSR will verify if the network connectivity can be maintained in his absence. If so, the PSR will recompute his routing table and will send a message to the coordinator to get permission for entering in the sleeping state, since it is not allowed more than one PSR in the sleep state. This is necessary to guarantee that the remaining PSRs will not be overloaded. In case of a positive response from the coordinator, the PSR will broadcast the rebuilt routing table and will enter in the sleeping state for a certain period of time. If not, it will remain in the working state.

After waking up from the sleeping period, the PSR will rejoin the network by using the existing routing protocol and the routing tables of all the network nodes will be rebuilt. When the coordinator gets aware of the waking up of the PSR, it will verify if its own maximum link utilization is greater than the threshold, i.e. the network loading is high (high-peek hours). If so, the coordinator will send a wakeup message to the PSR, otherwise it will do nothing. The PSR is expecting a confirmation message from the coordinator in a certain period of time. If it receives the confirmation message the PSR will remain in the working state, otherwise it will go back to the sleeping state for another period of time.

In this protocol it was defined the process of putting routers in sleep mode to achieve energy savings. According to the results presented in [60], the GDRP-PS is able to achieve a reduction of approximately 18% in the total energy consumption of the network.

2.3.6.D Switching Off Network Elements

In [57] it is explored the possibility of switching off not only network links but also network nodes. The goal of the proposed algorithm is to find the minimum set of routers and links that must be powered on so that the total energy consumption of the network can be reduced. Hereafter it will be explained the proposed heuristics to solve the aforementioned energy consumption problem, taking into account the parameters described in Table 2.3.

Parameter	Description
x_{ij}	Link ij is on or off
y_i	Node i is on or off
$\mathcal{PL}_{ij}$	Power consumption of the link ij
$\mathcal{PN}_i$	Power consumption of the node i
f_{ij}^{sd}	Traffic from source to destination that is routed through the link ij
f_{ij}	Total amount of traffic that is routed through the link ij
t_{sd}	Average amount of traffic going from source to destination
c_{ij}	Capacity the link ij

Table 2.3: Parameters used in the problem formulation.

To reduce the total energy consumption, calculated using the Equation 2.4, of the network it must be discovered the routers and links that can be turned off without jeopardizing the network connectivity.

$$\mathcal{P}_{tot} = \sum_{i=1}^{N}\sum_{j=1}^{N} x_{ij}\mathcal{PL}_{ij} + \sum_{i=1}^{N} y_i \mathcal{PN}_i \tag{2.4}$$

In this solution it is assumed a complete knowledge of the network topology and of the average amount of traffic that is exchanged between all node pairs. Taking this into account, some constraints have been set (see Equation 2.5 to Equation 2.8).

$$\sum_{j=1}^{N} f_{ij}^{sd} - \sum_{j=1}^{N} f_{ji}^{sd} = \begin{cases} t^{sd}, & \forall s,d,i=s \\ -t^{sd}, & \forall s,d,i=d \\ 0, & \forall s,d,i \neq s,d \end{cases} \tag{2.5}$$

$$f_{ij} = \sum_{s=1}^{N}\sum_{d=1}^{N} f_{ij}^{sd} \quad \forall i,j \tag{2.6}$$

$$f_{ij} \leq \alpha c_{ij} x_{ij} \quad \forall i,j \tag{2.7}$$

$$\sum_{j=1}^{N} x_{ij} + \sum_{j=1}^{N} x_{ji} \leq M y_i \tag{2.8}$$

The Equation 2.5 describes the flow conservation constraints, whereas Equation 2.6 gives the total traffic flow that is routed on each link. The Equation 2.7 forces an upper limit in the traffic load that will be allowed on each link, while Equation 2.8 specifies that a node can only be turned off when all links that are connected to it are also turned off. Also, the M parameter in Equation 2.8 forces the following constraint, $M \geq 2N$.

The proposed algorithm will iteratively try to switch off a network element (node or link). In this case, at each iteration the network element will be disabled and all the shortest paths will be recomputed. After this, it will be verified if the network remains its connectivity and that the traffic demand can be satisfied.

2.3.6.E Dijkstra-based Power-aware Routing Algorithm

The DPRA [61] is an heuristic algorithm that consists in the partitioning of the traffic demand, δ, from a source node to a destination node. Then it will be computed the path that consumes the minimum power for the specified traffic demand, taking into account the resources that are already allocated. This will be executed for all node pairs and until all the traffic demand is allocated.

Each link of the network will be associated with a cost equal to the increase of the power consumption of the destination node, which can be calculated taking into account the traffic of the link and the energy profile of the destination node. Afterwards, it will be calculated the maximum resources in use by each node and consequently by each link, excluding the nodes and links whose available resources are not enough for the allocation of more traffic. Finally, the Dijkstra algorithm will be executed taking into account the newly calculated costs and the disabled network elements.

2.3.6.F Green-Game

The Green-Game [56] proposes a model that tries to solve a resource consolidation problem by taking into account both the traffic load and the network topology. Using this information it will be possible to rank the contribution of each node in the packet delivery process. This can achieve a good tradeoff between performance and energy savings, since the ranking combines traffic awareness and topology awareness. Taking this into consideration the Green-Game will try to find the set of nodes that can safely be turned off on low load networks.

The ranking of each node will be obtained by computing the Shapley Value [62]. The Shapley Value will rank nodes with a higher value when their absence disconnects the network and when their presence its very important in the packet forwarding process. In combination with the traffic load, the Sapley Value can efficiently distinguish the network nodes by their importance. Hence the network nodes with the lowest Shapley Value will be possibly turned off.

The high complexity in the computation of the Shapley Value makes it unsuitable for being applied in real networks. This way, in the Green-Game was proposed some optimizations to reduce the computation complexity of the Shapley Value, so it can become practical in real networks.

The work developed in the Green-Game provides an efficient way of choosing which network elements to be turned off. The higher ranked network elements will most likely be the most used ones. By using this measure it will be possible to reduce the impact of the energy saving mechanism in the network performance.

2.3.6.G Summary

The reviewed energy saving models provide a mechanism that can put network elements in a power saving mode. In Table 2.1 is presented a summary of the main characteristics of each energy saving model.

Energy Saving Model	Link Control	Node Control	Offline Scheme
Dynamic Link Metric	✓	✗	✗
Green OSPF	✓	✗	✗
GDRP-PS	✗	✓	✗
Switching Off Network Elements	✓	✓	✓
DPRA	✓	✓	✓
Green-Game	✓	✓	✗

Table 2.4: Energy saving models summary.

3
Architecture

Contents

In this chapter it will be described the proposed solution, which seeks to reduce the energy consumption of networks by enabling network elements to be turned off. Throughout this chapter it will be detailed the implemented simulation environment, the energy consumption model and the energy saving algorithm.

The proposed solution embeds energy awareness to the IP network architecture and to the PSIRP network architecture. This is done by controlling the working state of network elements and by exploring their idleness periods. This way, it will be possible to turn off the unused network elements. It will also be used traffic aggregation to give an opportunity for underused network elements to be turned off.

Finally, the proposed solution will allow a complete evaluation of both network architectures, in terms of performance and energy consumption. Making it possible to accurately evaluate the tradeoff between energy savings and network performance.

3.1 Architecture Model

The architecture of the proposed solution is composed of three major modules, which are: the simulation environment; the energy consumption model and the energy saving algorithm. The simulation environment will be responsible for introducing traffic in the implemented architectures, IP and PSIRP. In both architectures, it was implemented an energy saving module that tries to reduce the overall network energy consumption.

It was also implemented an energy consumption model that estimates the overall network energy consumption in both architectures. In Figure 3.1 it is shown the basic interactions between the different modules. Hereafter, it will be explained in more detail the modules that compose the proposed solution.

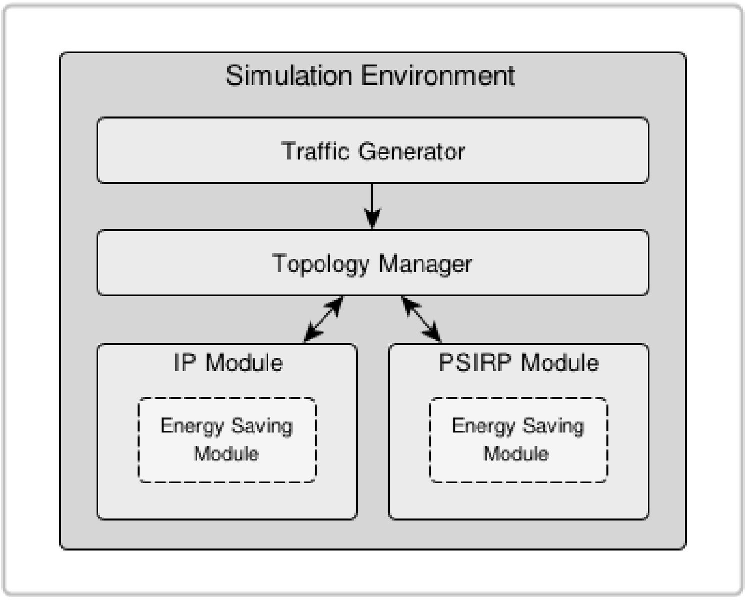


Figure 3.1: The basic architecture of the proposed solution.

3.1.1 Simulation Environment

This environment is responsible for simulating the traffic flow in the implemented network architectures, with and without the energy saving algorithm, and for collecting the correspondent statistical data. The flexibility provided by simulation will allow to test the implemented modules in different network conditions and topologies.

3.1.1.A Traffic Generator

The traffic generator will create the packets to be routed over the network. Upon the formation of a new packet it will be randomly chosen its source and destination nodes. A node will only be selected as source if it is not in the sleep mode. Also, a new packet will be generated from time to time. The packet size, the number of packets and the inter-packet interval are the configuration parameters provided by the traffic generator, which will allow to simulate different traffic demands.

3.1.1.B Topology Manager

The topology manager will have at all times an overview of the network conditions. It will provide the necessary abstractions for controlling the simulation environment and for managing the network. With these abstractions it will be possible to check for partitions in the network when turning off a link or node, which is of great importance to the energy saving algorithm.

3.1.2 Network Energy Optimization

In this section it will be described the network architectures, in which the energy saving algorithm will be applied. The proposed solution comprises two different architectures, one that replicates the current Internet model (IP) and another that is based on the PSIRP model. As previously mentioned, both of these architectures will be implemented with a mechanism for enabling/disabling the energy saving algorithm. Lastly, the evaluation of the proposed solution will be performed on these two architectures.

3.1.2.A Energy optimization in the IP network

In this section it will be described the architecture that replicates the current Internet. The implemented architecture is based on the IP stack and on the Dijkstra's algorithm to find the shortest path between a pair of nodes in the network. In this sense, this architecture is an extension of the current IP architecture with an energy saving module.

This new module will allow the unused network elements, nodes or links, to be turned off in order to achieve a reduction in the overall network energy consumption. Also, the energy saving module was implemented in the network node. The components belonging to this architecture are detailed below.

In Figure 3.2 it is described a basic network architecture using the IP stack in conjunction with the energy saving module. The architecture is composed of some network nodes which are connected by communication links, being the network node responsible for sending and receiving information.

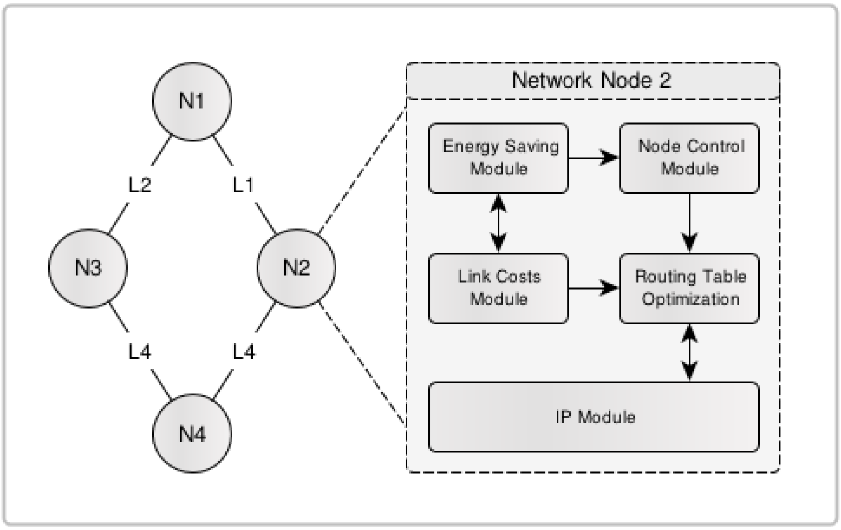


Figure 3.2: Module interaction in a network node with the IP stack.

As said before, the selection of the shortest path between a pair of nodes will be made by computing the Dijkstra's algorithm over the costs of links. To allow this functionality and the possibility to reduce the network energy consumption, several modules were implemented on the network node. The implemented modules are as follows:

- **Energy saving module:** This module is responsible for applying the energy saving algorithm to the network node. It will make decisions concerning the traffic aggregation to allow the possibility of redirecting traffic to most used network elements. Also, it will be responsible for the decisions regarding the working state (wake and sleep) of the network node. This way, it will try to achieve the best network configuration that will allow a reduction in the overall network energy consumption.

- **Node control module:** This module is responsible for controlling the working state of the network node. It will allow the network node to be waked up and be put to sleep according to the decisions made by the energy saving algorithm. Also, it will allow to individually turn off each of the interfaces belonging to the network node.

- **Link costs module:** This module is responsible for applying the OSPF routing protocol to gather/update all the network link costs and status to be used in the computation of the routing tables, since each network element will need to be aware of the changes that may occur in the network topology due to the decisions made by the energy saving algorithm.

- **Routing table optimization:** This module is responsible for updating the routing tables of the network node according to the shortest path principle. This is done by computing the Dijkstra's algorithm over the weights of links that belong to the network. When updating the routing tables it will also be considered the state of each network element, node or link.

- **IP Module:** This module uses the IP stack to provide communication capabilities to the network node. This way, the network node will have the ability to send and receive data over its communication channels.

3.1.2.B Energy optimization in the PSIRP network

In this section it will described the architecture that represents a proposal for the future Internet, which does not rely on the current IP stack. The implemented architecture uses the PSIRP design specifications. In extension to the PSIRP architecture, it will also be used the Dijkstra's algorithm to solve the shortest path between two pair of nodes. The implemented architecture will not follow thoroughly all of the PSIRP specifications, being only implemented the forwarding functionality.

Finally, this model will be extended to allow the integration of an energy saving module. This new module will allow the unused network elements, nodes or links, to be turned off in order to achieve a reduction in the overall network energy consumption.

The PSIRP module was implemented inside the network node with the energy saving model as an extension. The components belonging to this new architecture are detailed below.

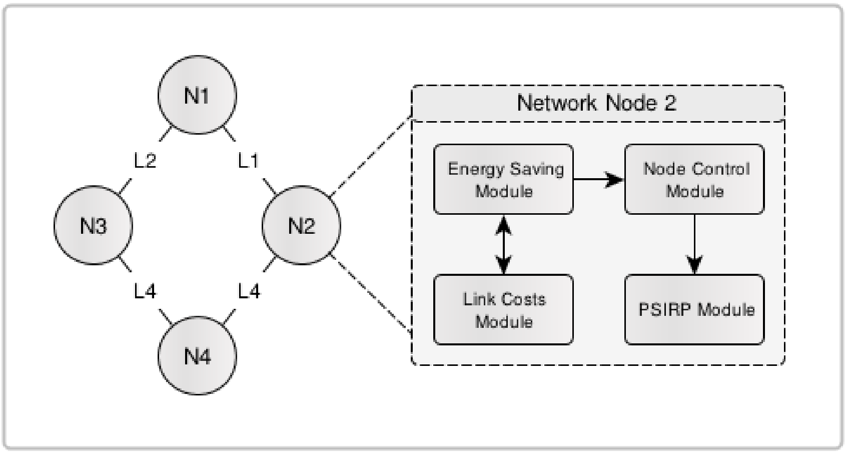


Figure 3.3: Module interaction in a network node with the PSIRP module.

In Figure 3.3 it is described a basic network architecture using the developed PSIRP module in conjunction with the energy saving module. The PSIRP architecture shares the same principles as the IP architecture, in which the network nodes are connected through communication links, allowing them to send and receive information. To provide these functionalities it was implemented the following modules:

- **Energy saving module:** This module has the same functionality as the one implemented in the IP architecture.

- **Node control module:** This module has the same functionality as the one implemented in the IP architecture.

- **Link costs module:** This module is different from the one implemented in the IP architecture, because it will not implement the OSPF protocol. Instead, it will gather/update the link costs by exchanging information with the topology manager. This way, every network node in the network will have access to the new topology information.

- **PSIRP module:** This module will give the communication capabilities to the network nodes. It was implemented a simplified version of the PSIRP architecture. The rendezvous system and the forwarding mechanism were the only considered functionalities in the implemented architecture. Following, it will be described the implemented functionalities.
  - *Rendezvous System:* The rendezvous system is responsible for managing the link identifiers used by each link that connects a pair of nodes. For each direction of the link it will be given a different link identifier. It is also responsible for the creation of the zFilter, which represents the routing path between a pair of nodes. The zFilter will contain the link identifiers of each hop that belongs to the selected routing path. The zFilter is the result of performing an OR operation over the link identifiers. Finally, the rendezvous system will select the routing path between two nodes according to the shortest path. To perform this computation it will be used the Dijkstra's algorithm.
  - *Forwarding Mechanism:* The forwarding mechanism represents the packet delivery process. When a network node wants to send a packet, it will check which of its interfaces belong to the given zFilter. After that, the packet will be sent through the selected interface. The matching of the link identifier against the zFilter has a low computational complexity, since it is based in performing an AND operation between the zFilter and link identifier. The forwarding process of the PSIRP is described in Algorithm 3.1. More details and possible optimizations on the forwarding process are presented in [63].

Algorithm 3.1 The PSIRP forwarding process.

```
for all $LinkID's$ do
  if $zFilter \,\&\, LinkID == LinkID$ then
    Forward the packet through the link
  end if
end for
```

3.1.3 Network Energy Consumption Model

In this section it will be described the energy consumption model used for calculating the overall network energy consumption of the implemented architectures. This will allow to evaluate the possible energy savings that may be achieved when enabling the energy saving module.

To accurately evaluate the energy savings, it is desirable that the energy consumption model can represent a good estimation of the energy consumed by real network devices. In this sense, it was implemented the energy consumption model that was proposed in [64].

3.1.3.A Energy model parameters

The following parameters were considered in the formulation of the energy consumption model. This includes the energy consumption of each network element (see Table 3.1) and a brief description of the parameters used in the overall network energy consumption modelling (see Table 3.2).

Network element	E_0 [Watt]	M [Watt]
Nodes	$0.85C^{2/3}$	$C^{2/3}$
(0-100) Mbps links	0.48	0.48
(100-600) Mbps links	0.90	1.00
(600-1000) Mbps links	1.70	2.00

Table 3.1: Energy consumption of the network elements.

Parameter	Description
a	Network element (node or link)
x_a	Defines the network element state (on or off)
c_a	Capacity of the network element
α_a	Number of bits sent over the link a
β_a	Number of dropped bits at link a
τ	Time window for calculating the network element utilization

Table 3.2: Paremeters used in the energy consumption formulation.

3.1.3.B Link Utilization

The link load can be estimated by knowing the number of bits sent over the channel/link in a certain period of time. This can be done by recording the packet transmission start time, transmission end time and the size of the packet. It must be taken into account the fact that a packet can be retained in the transmission queue, because of the occupation of the channel by another started transmission. Hence the start time of the packet corresponds to the instant in which it was dequeued. The end time is the instant in which the last bit of the packet was pushed into the channel. With this information it is possible to know which packets were sent during a certain period of time. If a packet is not totally sent during one time window, then it will be estimated the number of bits that can be sent during the remaining time. It is assumed the maximum load per link when the start time belongs to a previous time window and the packet does not finish in the current time window. Taking this into account the link utilization, lu, can be defined as the ratio between the load and the maximum capacity/bandwidth of the link in a certain period of time τ. The link utilization estimation can be summarized by Equation 3.2.

$$l\left(i,j\right) = \left(\alpha_{ij} - \beta_{ij}\right) \tag{3.1}$$

$$lu\left(i,j\right) = \frac{l_{ij}}{c_{ij} * \tau} \tag{3.2}$$

3.1.3.C Node utilization

The node utilization, nu, can be estimated by performing a ratio between the node load and its respective switching capacity during a certain period of time τ. The node load will be assumed to be equal to the traffic load that enters and leaves the node (see Equation 3.3). The overall switching capacity of the node will be considered to be proportional to the capacity of each link that is connected to it (see Equation 3.4). Taking this into account, the node utilization in a certain period of time can be calculated using the Equation 3.5.

$$l\left(n\right) = \sum_{(i,n)\in\mathcal{L}} l_{in} + \sum_{(n,i)\in\mathcal{L}} l_{ni} \tag{3.3}$$

$$c\left(n\right) = 2 \sum_{(i,j)\in\mathcal{L}} c_{ij} \tag{3.4}$$

$$nu\left(n\right) = \frac{l_n}{c_n * \tau} \tag{3.5}$$

3.1.3.D Network energy consumption

Taking into account the previous constraints it is possible to define the total network energy consumption as the amount of energy spent by all nodes and links that belong to the network topology and that are powered on. Also, the energy consumption of each network element varies according to its utilization. When a network element is powered on it consumes a constant amount of energy, E_0, even when its utilization is zero. If its utilization is greater than zero then the energy consumption of the network element will be increased by a fraction (link/node utilization) of the difference between M and E_0, also denoted as E_{fa}. The M parameter is the energy consumption of an element when fully used and the E_0 parameter is the energy consumption when the network element is idle.

According to this model, the total network energy consumption can be represented by Equation 3.6. Since links are full duplex, the sum of the energy consumption of all links will be divided by two to avoid counting twice the energy spent by them. Finally, in the proposed energy consumption model it is assumed that the necessary energy for switching between the states on and off is equal to zero.

$$E_T = \frac{1}{2} \sum_{(i,j)\in\mathcal{L}} \left(\left(lu_{ij} + lu_{ji}\right) E_{fij} + x_{ij} E_{0ij}\right) + \sum_{n\in\mathcal{N}} \left(nu_n E_{fn} + x_n E_{0n}\right) \tag{3.6}$$

3.1.4 Energy Saving Algorithm

The proposed algorithm aims to reduce the overall network energy consumption by exploring the possibility of turning off the network elements, nodes or links, which are not being used. This has to be done taking into account the resulting impact in the performance of the network. To achieve the best tradeoff between energy consumption and performance, the following questions were taken into consideration during the design of the energy saving algorithm.

- *How to aggregate traffic to most frequently used links?*
- *Which are the network elements that can be turned off? In which sequence?*
- *When to turn back on links, in order to reduce the impact in the network performance?*

This solution is not initially aware of the network traffic patterns, trying to dynamically adapt to the changes in the traffic demand. Hereafter it will be detailed the procedures applied by the algorithm to reduce the overall network energy consumption without jeopardizing too much network performance.

3.1.4.A Description

For the purpose of saving energy, the algorithm will turn off the network elements that are not being used. The network elements to be turned off must be carefully selected in order to reduce the inevitable impact in the network performance. Hereafter it will be explained how the algorithm will try to achieve the best tradeoff between energy savings and network performance.

The algorithm is divided in two main functions, which are: the traffic aggregation and the selection of the network elements to be turned off. Firstly, the use of traffic aggregation will allow the possibility of turning off the underused links by transferring their traffic to other links which have higher utilization. Using this mechanism it will be possible to induce the underused links to an idle mode, allowing for a greater number of network elements to be turned off. Lastly, the selection algorithm will choose the network elements to be turned off and in which order. This will greatly affect energy savings and network performance. The detailed steps of the algorithm are enumerated below:

1. **Check the utilization of every link:** For each node in the network, it will be analysed the utilization of all its links. When the utilization of a link exceeds the threshold, its cost will be increased to avoid congestion. The links with a utilization below the threshold will become candidates for a cost decrease. Among the candidates, only the link with higher utilization will have a decrease in its cost to allow the aggregation of some traffic into it. Lastly, the links that are not being used will be chosen as candidates for the turn off procedure.

2. **Ranking the network elements:** In this step it will be assigned a ranking to each network element that has been selected as candidate to be turned off. This ranking will reflect the importance of the network element in the network. This way it will be possible to specify the sequence in which the network elements will be turned off, starting with the least important ones.

3. **Turn off the network elements:** With the output of the previous step, each of the chosen network elements will be possibly turned off. If a network node can not be put to sleep then it will be verified if any of its links can be safely disconnected. This is done because the network nodes are the network elements which most influence the energy consumption and the performance of the network.

Finally, the algorithm may be forced to turn back on some links to avoid the possibility of network congestion in the case of high traffic demand. These links will be reconnected taking into account their significance in the network, starting with the most important ones.

3.1.4.B Traffic Aggregation

The basic mechanism of the energy saving algorithm is to turn off the unused links (idle links). But this is not always possible, since links may be underused because of low traffic demand. These links may also be turned off if their utilization drops to zero, which can be done by aggregating the traffic to other links that have higher utilization. The aggregation will be achieved by dynamically changing the cost of the links, which are used by the Dijkstra algorithm to compute the shortest path between a pair of network nodes.

It will be established a threshold to avoid congestion in the links that will carry the aggregated traffic. Taking this into account, the cost of a link will be decreased when its utilization rate, lu, is below the threshold and it will be increased when the utilization rate exceeds the threshold (see Equation 3.8). The modification of the link cost takes into account the remaining traffic, λ, that can be allocated to it (see Equation 3.7).

$$\lambda\left(i,j\right) = \left|1 - lu_{ij}\right| \tag{3.7}$$

$$cost'\left(i,j\right) = \begin{cases} \lambda_{ij} * cost_{ij}, & lu_{ij} \leq threshold \\ \dfrac{cost_{ij}}{\lambda_{ij}}, & lu_{ij} > threshold \end{cases} \tag{3.8}$$

When the utilization of a link is below the threshold its cost will not be immediately decreased, becoming only a candidate for this specific change. After knowing the candidate links, it will be chosen the one with the highest utilization and its cost will be decreased. This will be done independently by each node. This procedure is summarized by Algorithm 3.2.

Algorithm 3.2 Traffic aggregation to most used links.

```
for $i = 1 \to N$ do
  for $j = 1 \to L$ do
    $u \leftarrow LinkUtilization\left(i,j\right)$
    if $u > threshold$ then
      $IncreaseCost\left(i,j\right)$
    end if
    $j \leftarrow j + 1$
  end for
  Decrease cost of the most used link of the node below the threshold
  $i \leftarrow i + 1$
end for
```

In Figure 3.4 it is illustrated a scenario where the traffic is being routed from nodes 0 and 1 to node 3. In this situation the only network element that can be turned off is the link that connects nodes 0 and 1.

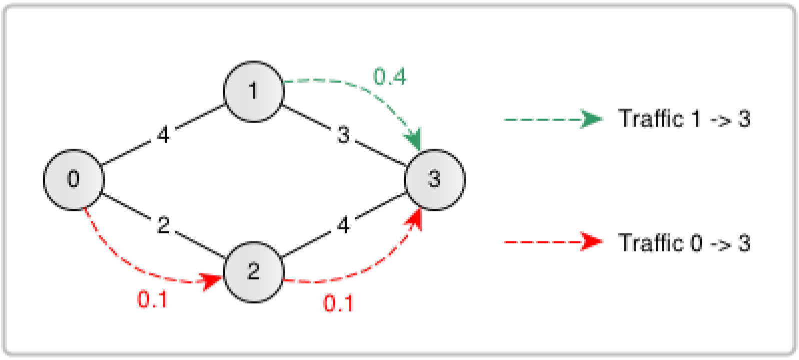


Figure 3.4: Traffic demand before changing the link costs.

The traffic from node 0 to node 3 has to pass necessarily through node 2, because it makes part of the shortest path computed by the Dijkstra's algorithm. In this case it is possible to better manage the used resources by making the traffic between the nodes 0 and 3 pass through node 1, instead of node 2. To achieve this result the cost of the link that connects the nodes 1 and 3 will be decreased. After applying these changes the node 2 can go to sleep since it will no longer be used for routing traffic (see Figure 3.5).

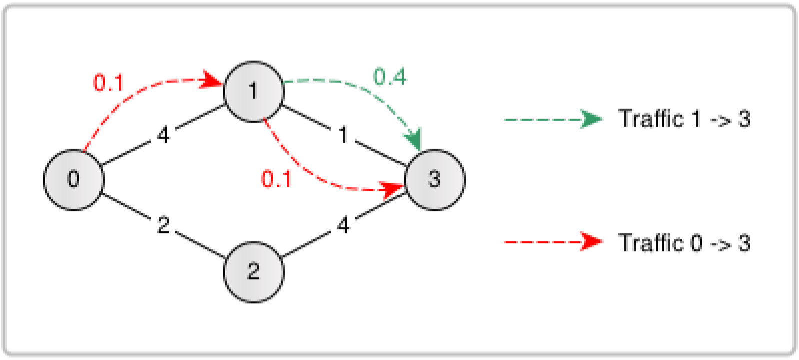


Figure 3.5: Traffic demand after changing the link costs.

3.1.4.C Ranking Network Elements

The ranking will use the local centrality measure [65] to classify the importance of each network element in the network topology. It was chosen this measure because of its low complexity and because it is more accurate than degree centrality. The local centrality of the network node v, $C_L(v)$, is then defined as

$$Q(u) = \sum_{w \in \Gamma(u)} N(w) \tag{3.9}$$

$$C_L(v) = \sum_{u \in \Gamma(v)} Q(u) \tag{3.10}$$

where $\Gamma(u)$ is the set of the nearest neighbours of node u and $N(w)$ is the number of the nearest and the next nearest neighbours of node w.

The network elements will be ranked by mixing its history of utilization and its local centrality. It will be considered a sliding time window for the utilization history. In other words, the history of utilization will be reset after a certain period of time. Taking this into account, the ranking of nodes and links will be calculated as follows:

- Links: The ranking of links will be calculated using Equation 3.11, where H_{lu} is the history of utilization of a link. This computation will then be used to order the links by ranking. The links with the same ranking will be reordered by their local centrality, C_L.

$$\mathcal{R}_L(i,j) = (C_L(i) + C_L(j)) \times H_{lu}(i,j) \tag{3.11}$$

- Nodes: The ranking of nodes will be calculated using Equation 3.12, where H_{nu} is the history of utilization of a node. As for the links, this computation will allow to order the network nodes according to their ranking. The nodes with the same ranking will be reordered by their local centrality, C_L.

$$\mathcal{R}_N(n) = C_L(n) \times H_{nu}(n) \tag{3.12}$$

3.1.4.D Turning Off Network Elements

The Algorithm 3.3 briefly describes the procedure for turning off network elements. At each iteration of the energy saving algorithm it will be checked which network elements can be turned off.

Algorithm 3.3 Turning off the network elements.

```
$SortByRank(nodes)$
for $n = 1 \rightarrow N$ do
    if $CanGoToSleep(n)$ then
        $Sleep(n)$
    end if
    $n \leftarrow n + 1$
end for
$SortByRank(links)$
for $l = 1 \rightarrow L$ do
    if $CanBeTurnedOff(l)$ then
        $TurnOff(l)$
    end if
    $l \leftarrow l + 1$
end for
```

The links that are not in use will become candidates to be turned off. The candidate links will be ordered according to their ranking, starting with the lowest ones. The links with the same ranking will be ordered by their local centrality. After ordering the links, each one will be turned off if the network remains fully connective without it.

Turning off a node is not a trivial operation as it could easily lead to the partition of the network, i.e. the loss of communication between the remaining nodes of the network. Also, there is the possibility of losing the packets destined to the node during the time it is turned off. Hence it must be explored the possibility of turning off the node without jeopardizing the network connectivity and without losing too many packets that would otherwise be sent to it.

A node will only be put to sleep rather than turned off completely to give it the possibility of waking up at a later time. It is assumed that a sleeping node will have a very low energy consumption which may be ignored. Also, it will be considered that the energy consumption is zero when switching between the states awake and asleep. A node can only be put to sleep in the following conditions:

1. No remaining traffic in any of its links.
2. The remaining nodes of the network can still communicate with each other.
3. All of its neighbours which are in sleep mode can rejoin the network in its absence.

If all the above conditions are met, the network node can safely enter in the sleep mode. When a node goes to sleep all of its active links will be turned off. When the sleeping node wakes up, it will signal its neighbour nodes in order to receive the packets that possibly were stored during its sleeping period. If no packets were generated during its sleeping period the node can go to sleep again, otherwise it will remain awake until the algorithm decides to put it to sleep. When waking up the node, only the links which are connected to active nodes will be restored, avoiding the possibility of waking up the neighbours that are sleeping. The state transitions of a network node are shown in Figure 3.6.

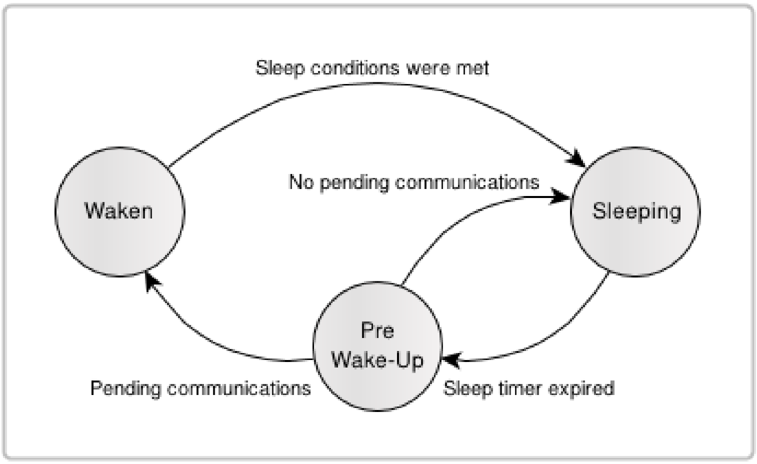


Figure 3.6: Switching between the on/off states.

3.1.4.E Turning Back On Links

To prevent a large decrease in the network performance some of the links that are turned off will be turned on again. At each iteration, only one link can be reconnected. This is done to prevent a peek in the energy consumption, maintaining good energy savings. Also, only the turned off links that connect two active nodes can be turned on, otherwise a sleeping node would be forced to wake up again.

The link with the highest local centrality will be chosen to be turned on. The local centrality of a link is the sum of the local centrality of the nodes that are connected by it. Lastly, if two links have the same local centrality then the link to be turned on will be randomly chosen.

3.1.4.F Network Connectivity

The verification of the network connectivity is necessary because the process of turning off network elements can lead to isolated nodes in the network, i.e. network partitions. Before turning off a node, the algorithm will verify if the remaining active nodes will be able to communicate among each other. This completely eliminates the possibility of occurring network partitions.

For each node it will be created a set containing its active links and then all the sets will be intersected. The sets in which the same node participates will be merged. Afterwards, the network will be considered to be fully connective if the resulting set contains all the active nodes of the network, as shown in Figure 3.7. Otherwise, there will be active nodes that will not be able to participate in the network as shown in Figure 3.8 and Figure 3.9.

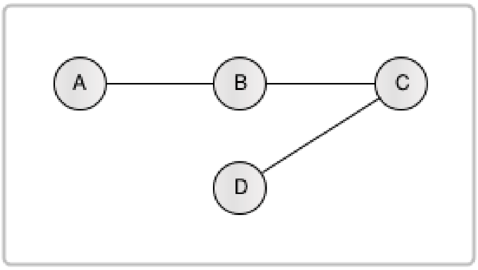


Figure 3.7: Network with no partitions.

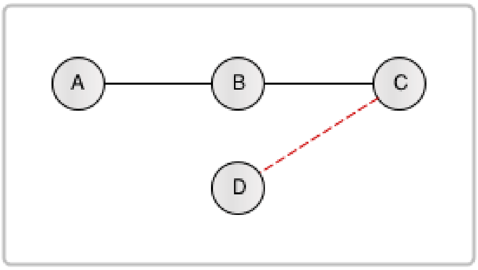


Figure 3.8: Partitioned network due to a turned off link.

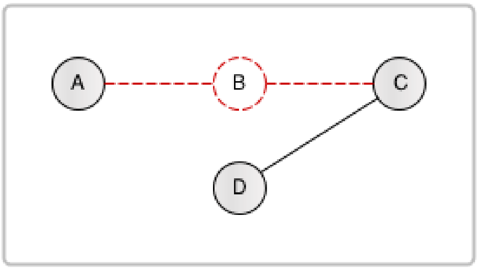


Figure 3.9: Partitioned network due to a sleeping node.

3.2 Limitations

In this section it will be discussed the limitations that were found during the implementation and testing of the proposed solution. Unfortunately, it was not possible to solve them all in due time. Following, it will be enumerated the limitations that were not addressed in the solution.

- **Energy consumption when waking up:** It is assumed that the transition between the sleeping and the working states does not consume any energy, which is not true in a real live situation. With this assumption, it is not possible to know the impact on energy savings when waking up a node. Ideally, the energy saving algorithm should try to reduce at maximum the number of wakeups, in order to achieve good energy savings.

- **Fixed sleeping period:** The energy saving algorithm assumes a fixed sleeping period for the network nodes, which may generate unnecessary wakeups on some of them. The sleeping period should take into consideration the number of transitions between the pre-wakeup and the working states. So, the nodes that after the expiration of the sleeping period do not pass consecutively from the sleep state to the working state, should see their sleeping expiration period increased, otherwise it should be decreased. Ideally, the sleeping expiration period should be dynamic to allow further energy savings and to possibly reduce the delay introduced by putting an important network element into sleep.

- **TCP Traffic:** Ideally, the energy saving module should be tested in the IP architecture with both UDP and TCP traffic. Unfortunately, the TCP traffic flowing does not work well when the energy saving module is enabled in the IP architecture. This may happen because of the loss of the ack packets during simulation. Also, in the NS3 bug tracker their are some open bugs for TCP to be resolved in the next version (3.15).

- **Publish/Subscribe data:** It was not implemented the mechanism for publishing and subscribing data in the PSIRP architecture. As said before, it was only implemented the forwarding procedure, which means that the publisher and the subscriber are well known. Because of this, it is not possible to evaluate the overhead in the network that is caused by the publish and subscribe messages.

- **LIT implementation:** The utilization of bloom filters in the PSIRP architecture, can cause a lot of false positives when matching the LID's against the zFilter. This is very problematic in very dense networks where the probability in generating a false positive is very high. The false positive probability is given by Equation 3.13, where m is the size in bits of the bloom filter, k is the number of bits set to one in the bloom filter and n is the number of links in the network. In [63], it is proposed the usage of LIT to reduce the probability of false positives, which was not implemented in the solution.

$$fpb = \left[1 - \left(1 - \frac{1}{m}\right)^{k*n}\right]^k \tag{3.13}$$

4
System Evaluation

Contents

In this chapter it will be discussed the experimental results of the implemented solution, with the main goal of verifying the tradeoff between energy savings and the performance of the network.

The system evaluation was carried on the IP and PSIRP architectures. Both of the architectures were evaluated using three network topologies with different traffic demands. It was also evaluated the behaviour of the energy saving algorithm when disabling the node sleeping mechanism and when disabling the traffic aggregation mechanism.

4.1 Evaluation Goals and Settings

In this section it will be detailed the goals and the configurations of the performed evaluation. This evaluation will try to answer the major questions that led to this work, which are summarized as follows:

- How much energy it can be saved when using the energy saving module?
- What is the impact of the energy saving module in the performance of the network?
- What is the tradeoff between network performance and energy savings? Is it worth?

To answer the above questions, it was gathered a wide range of data that allows to study the implemented solution in terms of network performance and energy savings. In Table 4.1 it is presented the metrics used to perform this evaluation.

Metric	Description
Throughput	Gives the packet delivery average rate.
Delay	Gives the average time that a packet needs to go from the source to the destination.
Link Utilization	Gives the average link utilization of the network.
Energy	Gives the overall energy consumption of the network.

Table 4.1: The evaluation metrics.

4.1.1 Used Tools

The proposed solution was implemented, tested and evaluated using the NS3[1], which is a discrete event network simulator mostly used for research and education purposes. It is open-source, being licensed under the GNU GPLv2 license[2]. Despite of being the natural successor of the NS2[3], it does not offer backward compatibility with it. Furthermore, it provides full support to the C++ and Python programming languages, which are used for describing models and program flow.

4.1.2 Network Topologies

For the evaluation it was considered three different network topologies to be used in the experimental scenarios, which allows to verify the adaptation of the energy saving algorithm to different networks.

[1] The Network Simulator - NS3. `http://www.nsnam.org/` (Last access: 13-03-2012)

[2] GNU General Public License, version 2. `http://www.gnu.org/licenses/gpl-2.0.html` (Last access: 13-03-2012)

[3] The Network Simulator - NS2. `http://www.isi.edu/nsnam/ns` (Last access: 13-03-2012)

To provide a more realistic scenario it was chosen three different topologies from real networks: Abilene[4], COST 239[5] and Géant[6]. These topologies are widely used for research purposes.

The Abilene network is a high performance backbone network that is used by universities and research laboratories in the USA to deploy newly developed applications and network services. In Figure 4.1 it is shown the topology of the Abilene network.

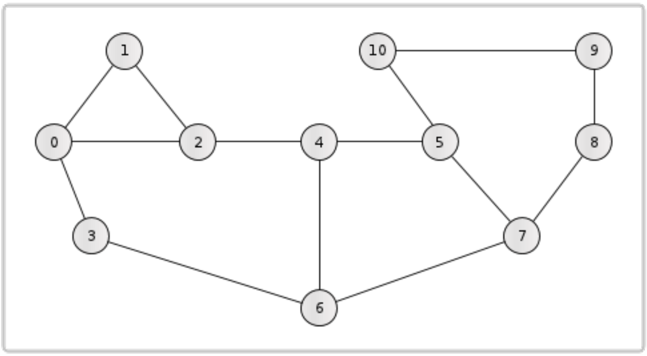


Figure 4.1: The topology of the USA Abilene network.

The COST 239 network is an European optical network, which is widely used as reference in the evaluation of numerous studies in the networking field. In Figure 4.2 it is shown the topology of the COST 239 network.

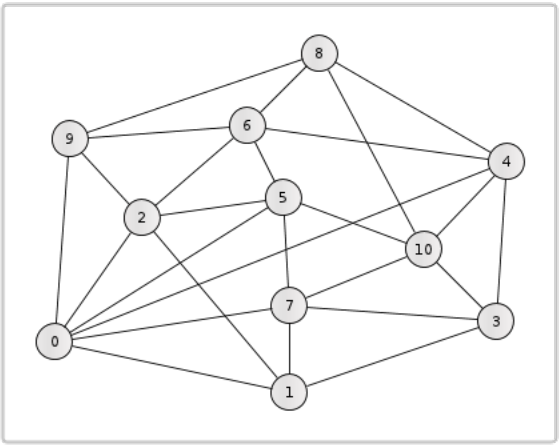


Figure 4.2: The topology of the Pan-European COST 239 network.

The Géant network is a high bandwidth backbone network dedicated for education and research purposes. This network interconnects the National Research and Education Networks (NRENs) across Europe. In Figure 4.3 it is shown the topology of the Géant network.

The chosen network topologies vary in the number of nodes and links. The weight of the links were randomly chosen between the values 1 and 10. Also, it will be used links with a capacity of 10 Mbps. The main characteristics of the topologies are summarized in Table 4.2.

[4] The Abilene network. `http://www.internet2.edu/network` (Last access: 23-04-2012)

[5] European Cooperation in Science and Technology (COST). `http://http://www.cost.eu` (Last access: 23-04-2012)

[6] The Géant network. `http://www.geant.net/pages/home.aspx` (Last access: 23-04-2012)

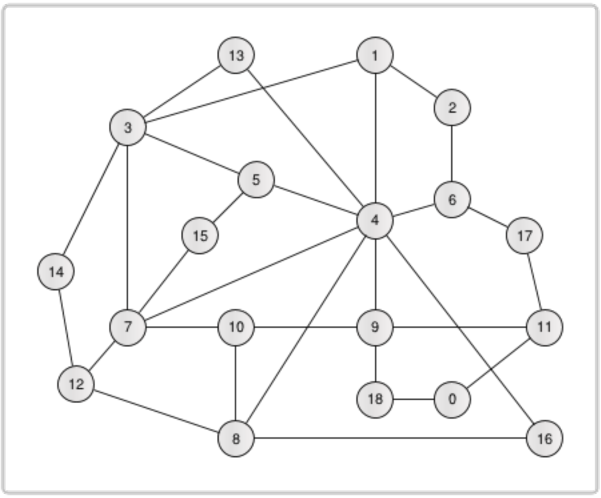


Figure 4.3: The topology of the Pan-European Géant network.

Network	Nodes	Links	Average Degree	Reference
Abilene	11	14	2.55	[66]
COST 239	11	26	4.73	[67]
Géant	19	30	3.16	[68]

Table 4.2: The network topologies used in evaluation.

4.1.3 Experimental Scenarios

It was chosen different experimental scenarios that will allow to study the adaptation of the energy saving algorithm to different network conditions. To perform this study, it was introduced different traffic loads into the network. It was evaluated the energy savings and network performance, when the network is lightly loaded and when the network is heavily loaded.

It was also evaluated two scenarios in which the energy saving module is partially enabled. In these two scenarios, it will firstly be disabled the node sleeping mechanism and then it will be disabled the traffic aggregation mechanism. Following, it will be described the goals of each experimental scenario.

- **Light Traffic Scenario:** The goal of this scenario is to test the energy saving algorithm in lightly loaded networks. In this situation, it will be possible to achieve large energy savings because there will be some unused network elements which will be turned off. Due to the low traffic demand it is not expected a major performance reduction of the network.

- **High Traffic Scenario:** The goal of this scenario is to test the energy saving algorithm in heavy loaded networks. In this situation, the solution will be tested in a more realistic scenario, being expected a low reduction in the energy consumption. Some network elements may be turned off, but due to performance constraints they will eventually be turned on again.

- **Only Turn Off Links Scenario:** The goal of this scenario is to test the energy saving algorithm without enabling the node sleeping mechanism. In this scenario, only the network links are allowed to be turned off.

- **No Traffic Aggregation Scenario:** The goal of this scenario is to test the energy saving algorithm without enabling the traffic aggregation mechanism. Without this mechanism there will be fewer network elements that can be turned off.

These scenarios try to cover the best and worst situations that can happen in real networks. In Table 4.3 it is described the traffic conditions used in each of the chosen experimental scenarios.

Scenario	Traffic Size	Inter-Packet Interval
Light Traffic	100 KB	880 μs
Heavy Traffic	1 MB	400 μs
Only Turn Off Links	100 KB	880 μs
No Traffic Aggregation	100 KB	880 μs

Table 4.3: The traffic conditions in the different scenarios.

4.2 Evaluation Results

This section is dedicated to the presentation of the evaluation results. These results were obtained by running the simulation 40 times for each experimental scenario, being calculated the average and the standard deviation for each of the chosen evaluation metrics. This procedure was used to evaluate the system with and without the energy saving algorithm.

With the collected results is possible to analyse the tradeoff between network performance and energy savings in each of the chosen experimental scenarios.

4.2.1 Light Traffic Scenario

This section describes the evaluation results of the solution in the presence of low traffic demand. In this experiment it will be tested how much energy it can be saved in a best case scenario, in which the network is lightly loaded. Hereafter, it will be presented the evaluation results of this experimental scenario.

4.2.1.A Without the energy saving algorithm

Table 4.4 shows the results obtained by injecting a small amount of traffic in the different network topologies and by using the IP/PSIRP architectures. In this situation, the energy saving algorithm was disabled, serving as base for the analysis of the previously defined metrics in a low traffic scenario.

4.2.1.B With the energy saving algorithm

Table 4.5 shows the results obtained by injecting a small amount of traffic in the different network topologies and by using the IP/PSIRP architectures. In this situation, the energy saving algorithm was enabled.

Abilene Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\tilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\tilde{x}$)**	**STD (σ)**
Throughput (Mbps)	8.536	0.166	Throughput (Mbps)	8.830	0.165
Delay (ms)	2.530	0.192	Delay (ms)	2.340	0.189
Link Utilization (%)	17.822	0.011	Link Utilization (%)	16.601	0.010
Energy (W)	12.318	0.198	Energy (W)	12.267	0.127
COST 239 Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\tilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\tilde{x}$)**	**STD (σ)**
Throughput (Mbps)	8.852	0.099	Throughput (Mbps)	9.375	0.075
Delay (ms)	1.646	0.097	Delay (ms)	1.407	0.068
Link Utilization (%)	6.872	0.003	Link Utilization (%)	5.951	0.002
Energy (W)	18.369	0.152	Energy (W)	18.349	0.130
Géant Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\tilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\tilde{x}$)**	**STD (σ)**
Throughput (Mbps)	8.584	0.135	Throughput (Mbps)	9.062	0.132
Delay (ms)	2.884	0.190	Delay (ms)	2.577	0.162
Link Utilization (%)	9.724	0.005	Link Utilization (%)	8.864	0.004
Energy (W)	24.110	0.274	Energy (W)	24.118	0.308

Table 4.4: The system evaluation without the energy saving algorithm in a low traffic scenario.

Abilene Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\tilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\tilde{x}$)**	**STD (σ)**
Throughput (Mbps)	7.650	0.244	Throughput (Mbps)	8.009	0.275
Delay (ms)	3.137	0.326	Delay (ms)	2.972	0.271
Link Utilization (%)	18.909	0.013	Link Utilization (%)	18.028	0.010
Energy (W)	8.735	0.331	Energy (W)	8.614	0.341
COST 239 Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\tilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\tilde{x}$)**	**STD (σ)**
Throughput (Mbps)	7.793	0.223	Throughput (Mbps)	8.594	0.162
Delay (ms)	2.132	0.237	Delay (ms)	1.745	0.109
Link Utilization (%)	7.302	0.006	Link Utilization (%)	6.280	0.003
Energy (W)	8.493	0.430	Energy (W)	8.034	0.280
Géant Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\tilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\tilde{x}$)**	**STD (σ)**
Throughput (Mbps)	7.901	0.210	Throughput (Mbps)	8.446	0.207
Delay (ms)	2.980	0.258	Delay (ms)	2.723	0.251
Link Utilization (%)	8.911	0.006	Link Utilization (%)	8.401	0.006
Energy (W)	12.857	0.556	Energy (W)	12.559	0.538

Table 4.5: The system evaluation with the energy saving algorithm in a low traffic scenario.

4.2.1.C Comparison

In Table 4.6 is given the results of the comparison made between disabling and enabling the energy saving algorithm (see Appendix A). These results reflect the variation of each metric after applying the energy saving algorithm. Following, it will be given a brief overview of the results for each metric.

- **Throughput:** As shown in Table 4.6, the throughput of the network decreases when the energy saving algorithm is enabled. It suffers a regression from $\sim 7\%$ to $\sim 12\%$ in the tested networks. These percentages represent a reduction in the network throughput of about 1 Mbps, in relation to the base system.
- **Delay:** As shown in Table 4.6, the packet delay of the network increases when activating the energy saving algorithm. It is introduced an extra delay that lies between $\sim 3\%$ and $\sim 30\%$ in the tested networks. These percentages represent an increase in delay which ranges from 0.1 to 0.6 ms, in relation to the base system.
- **Link Utilization:** As shown in Table 4.6, the average utilization of each link increases when the energy saving algorithm is enabled. This is mostly caused by the traffic aggregation algorithm, contributing with an increase of about $\sim 6\%$. As opposed to this, in the Géant network topology the average utilization of each link is decreased by $\sim 6\%$, because in this case there will be a lot of turned off links whose utilization will be zero.
- **Energy:** As shown in Table 4.6, the overall network energy consumption is reduced in no less than 29%. The results represent a decrease in the power consumption, which ranges from 4 W to 12 W in comparison to the base system.

Abilene Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-10.386	-9.294
Δ_{Delay} (%)	24.013	27.013
$\Delta_{\text{LinkUtilization}}$ (%)	6.103	8.597
Δ_{Energy} (%)	-29.089	-29.783
COST 239 Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-11.97	-8.33
Δ_{Delay} (%)	29.53	24.06
$\Delta_{\text{LinkUtilization}}$ (%)	6.26	5.54
Δ_{Energy} (%)	-53.77	-56.22
Géant Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-7.95	-6.80
Δ_{Delay} (%)	3.34	5.66
$\Delta_{\text{LinkUtilization}}$ (%)	-8.36	-5.23
Δ_{Energy} (%)	-46.68	-47.93

Table 4.6: The impact of the energy saving algorithm in a low traffic scenario.

4.2.2 High Traffic Scenario

This section describes the evaluation results of the solution in the presence of high traffic demand. In this experiment it will be tested how much energy it can be saved in a worst case scenario, in which the network is heavily loaded. Hereafter, it will be presented the evaluation results of this experimental scenario.

4.2.2.A Without the energy saving algorithm

Table 4.7 shows the results obtained by injecting a large amount of traffic in the different network topologies and by using the IP/PSIRP architectures. In this situation, the energy saving algorithm was disabled, serving as base for the analysis of the previously defined metrics in a high traffic scenario.

Abilene Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\widetilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\widetilde{x}$)**	**STD (σ)**
Throughput (Mbps)	6.184	0.327	Throughput (Mbps)	6.309	0.415
Delay (ms)	4.992	0.889	Delay (ms)	5.219	1.753
Link Utilization (%)	39.427	0.006	Link Utilization (%)	37.080	0.007
Energy (W)	56.184	0.475	Energy (W)	56.369	1.316
COST 239 Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\widetilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\widetilde{x}$)**	**STD (σ)**
Throughput (Mbps)	8.616	0.076	Throughput (Mbps)	8.895	0.056
Delay (ms)	1.755	0.056	Delay (ms)	1.554	0.032
Link Utilization (%)	14.694	0.003	Link Utilization (%)	13.271	0.002
Energy (W)	83.777	0.140	Energy (W)	83.610	0.152
Géant Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\widetilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\widetilde{x}$)**	**STD (σ)**
Throughput (Mbps)	7.497	0.122	Throughput (Mbps)	7.881	0.122
Delay (ms)	3.621	0.131	Delay (ms)	3.254	0.123
Link Utilization (%)	21.164	0.003	Link Utilization (%)	19.984	0.004
Energy (W)	108.843	0.391	Energy (W)	108.604	0.289

Table 4.7: The system evaluation without the energy saving algorithm in a high traffic scenario.

4.2.2.B With the energy saving algorithm

Table 4.8 shows the results obtained by injecting a large amount of traffic in the different network topologies and by using the IP/PSIRP architectures. In this situation, the energy saving algorithm was enabled.

4.2.2.C Comparison

In Table 4.9 is given the results of the comparison made between disabling and enabling the energy saving algorithm (see Appendix B). These results reflect the variation of each metric after applying the energy saving algorithm. Following, it will be given a brief overview of the results for each metric.

- **Throughput:** As shown in Table 4.9, the throughput of the network decreases when the energy saving algorithm is enabled. It suffers a regression from $\sim 9\%$ to $\sim 12\%$ in the tested

Abilene Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\widetilde{\text{x}}$)**	**STD (σ)**	**Metric**	**AVG ($\widetilde{\text{x}}$)**	**STD (σ)**
Throughput (Mbps)	5.647	0.201	Throughput (Mbps)	5.685	0.233
Delay (ms)	5.588	0.629	Delay (ms)	5.706	0.753
Link Utilization (%)	39.250	0.006	Link Utilization (%)	37.829	0.007
Energy (W)	52.379	0.855	Energy (W)	52.096	0.968
COST 239 Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\widetilde{\text{x}}$)**	**STD (σ)**	**Metric**	**AVG ($\widetilde{\text{x}}$)**	**STD (σ)**
Throughput (Mbps)	7.688	0.092	Throughput (Mbps)	7.949	0.097
Delay (ms)	2.154	0.076	Delay (ms)	2.008	0.072
Link Utilization (%)	14.973	0.002	Link Utilization (%)	13.927	0.002
Energy (W)	56.955	0.741	Energy (W)	54.849	0.698
Géant Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\widetilde{\text{x}}$)**	**STD (σ)**	**Metric**	**AVG ($\widetilde{\text{x}}$)**	**STD (σ)**
Throughput (Mbps)	6.752	0.170	Throughput (Mbps)	6.981	0.184
Delay (ms)	4.037	0.282	Delay (ms)	3.925	0.326
Link Utilization (%)	19.604	0.003	Link Utilization (%)	18.995	0.004
Energy (W)	79.806	1.345	Energy (W)	79.373	1.489

Table 4.8: The system evaluation with the energy saving algorithm in a high traffic scenario.

networks. These percentages represent a reduction in the network throughput of about 1 Mbps, in relation to the base system.

- **Delay:** As shown in Table 4.9, the packet delay of the network increases when activating the energy saving algorithm. It is introduced an extra delay that lies between ~ 9% and ~ 30% in the tested networks. These percentages represent an increase in delay which ranges from 0.4 to 0.7 ms, in relation to the base system.

- **Link Utilization:** As shown in Table 4.9, the average utilization of each link is not affected significantly by the energy saving algorithm when the network is heavily loaded, only causing a small fluctuation of about 0.6% in relation to the base system. This fluctuation can cause a small increase or decrease in the average link utilization.

- **Energy:** As shown in Table 4.9, the overall network energy consumption is reduced in a maximum of 35%. The results represent a decrease in the power consumption, which ranges from 4 W to 30 W in comparison to the base system. Another important aspect is the huge difference between the energy savings in the Abilene network and the ones in the other two. This difference occurs because the Abilene has fewer network elements, being that in a heavily loaded network the opportunities for putting nodes to sleep are scarce.

4.2.3 Only Turn Off Links Scenario

This section describes the evaluation results of the solution when enabling only certain parts of the energy saving module. In this case, the node sleeping mechanism has not been enabled. Because

Abilene Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-8.678	-9.892
Δ_{Delay} (%)	11.947	9.336
$\Delta_{\text{LinkUtilization}}$ (%)	-0.448	2.021
Δ_{Energy} (%)	-6.773	-7.580
COST 239 Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-10.77	-10.63
Δ_{Delay} (%)	22.72	29.22
$\Delta_{\text{LinkUtilization}}$ (%)	1.90	4.94
Δ_{Energy} (%)	-32.02	-34.40
Géant Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-9.94	-11.41
Δ_{Delay} (%)	11.49	20.61
$\Delta_{\text{LinkUtilization}}$ (%)	-7.37	-4.95
Δ_{Energy} (%)	-26.68	-26.92

Table 4.9: The impact of the energy saving algorithm in a high traffic scenario.

of this, only the network links were allowed to be turned off.

4.2.3.A Without the node sleeping mechanism

Table 4.10 shows the results obtained by injecting a small amount of traffic in the different network topologies and by using the IP/PSIRP architectures. In this situation, the energy saving algorithm was enabled without the node sleeping mechanism.

4.2.3.B Comparison

In Table 4.11 is given the results of the comparison made between disabling and partially enabling the energy saving algorithm without the node sleeping mechanism (see Appendix C). These results reflect the variation of each metric after applying the energy saving algorithm. Following, it will be given a brief overview of the results for each metric.

- **Throughput:** As shown in Table 4.11, the throughput of the network suffers from a very small regression. This is the expected result, since only the links that are not used for the forwarding procedure will be turned off. The maximum throughput decrease is around 600 Kbps.

- **Delay:** As shown in Table 4.11, the packet delay of the network increases in this situation. The extra delay introduced lies between $\sim$ 3% and $\sim$ 82% in the tested networks. These percentages represent an increase in delay which ranges from 0.1 to 1.1 ms, in relation to the base system. This regression mostly happens because turning off links may cause an increase in the number of hops necessary to successfully deliver a packet.

- **Link Utilization:** As shown in Table 4.11, the average utilization of each link increases. This is mostly caused by the traffic aggregation algorithm, contributing with an increase of about

Abilene Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\tilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\tilde{x}$)**	**STD (σ)**
Throughput (Mbps)	8.214	0.171	Throughput (Mbps)	8.407	0.199
Delay (ms)	2.994	0.208	Delay (ms)	3.040	0.235
Link Utilization (%)	19.845	0.011	Link Utilization (%)	19.944	0.011
Energy (W)	12.247	0.206	Energy (W)	12.306	0.248
COST 239 Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\tilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\tilde{x}$)**	**STD (σ)**
Throughput (Mbps)	8.459	0.138	Throughput (Mbps)	8.657	0.181
Delay (ms)	2.506	0.161	Delay (ms)	2.562	0.160
Link Utilization (%)	9.457	0.004	Link Utilization (%)	9.372	0.004
Energy (W)	18.151	0.327	Energy (W)	18.214	0.352
Géant Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\tilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\tilde{x}$)**	**STD (σ)**
Throughput (Mbps)	8.503	0.175	Throughput (Mbps)	8.735	0.169
Delay (ms)	2.962	0.172	Delay (ms)	2.965	0.188
Link Utilization (%)	9.756	0.004	Link Utilization (%)	9.647	0.006
Energy (W)	23.684	0.300	Energy (W)	23.844	0.647

Table 4.10: The system evaluation with the energy saving algorithm when disabling the node sleeping mechanism.

$\sim 3\%$.

- **Energy:** As shown in Table 4.11, the overall network energy consumption is negligibly reduced. This confirms the importance of the node sleeping mechanism in the reduction of the overall network energy consumption.

Abilene Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-3.782	-4.790
Δ_{Delay} (%)	18.341	29.897
$\Delta_{\text{LinkUtilization}}$ (%)	11.351	20.137
Δ_{Energy} (%)	-0.579	0.312
COST 239 Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-4.44	-7.66
Δ_{Delay} (%)	52.28	82.07
$\Delta_{\text{LinkUtilization}}$ (%)	37.63	57.50
Δ_{Energy} (%)	-1.19	-0.74
Géant Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-0.95	-3.61
Δ_{Delay} (%)	2.69	15.05
$\Delta_{\text{LinkUtilization}}$ (%)	0.33	8.83
Δ_{Energy} (%)	-1.77	-1.14

Table 4.11: The impact of the energy saving algorithm with the node sleeping mechanism disabled.

4.2.4 No Traffic Aggregation Scenario

This section describes the evaluation results of the solution when enabling only certain parts of the energy saving module. In this case, the traffic aggregation mechanism has not been enabled. Because of this, the underused network elements will have less opportunities to be turned off.

4.2.4.A Without the traffic aggregation mechanism

Table 4.12 shows the results obtained by injecting a small amount of traffic in the different network topologies and by using the IP/PSIRP architectures. In this situation, the energy saving algorithm was enabled without the traffic aggregation mechanism.

Abilene Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\widetilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\widetilde{x}$)**	**STD (σ)**
Throughput (Mbps)	7.552	0.277	Throughput (Mbps)	8.052	0.327
Delay (ms)	3.269	0.347	Delay (ms)	2.960	0.406
Link Utilization (%)	19.388	0.013	Link Utilization (%)	18.012	0.015
Energy (W)	8.747	0.302	Energy (W)	8.620	0.368
COST 239 Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\widetilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\widetilde{x}$)**	**STD (σ)**
Throughput (Mbps)	7.699	0.188	Throughput (Mbps)	8.494	0.195
Delay (ms)	2.217	0.204	Delay (ms)	1.834	0.185
Link Utilization (%)	7.405	0.004	Link Utilization (%)	6.482	0.005
Energy (W)	8.538	0.389	Energy (W)	8.106	0.391
Géant Network Topology					
IP			**PSIRP**		
Metric	**AVG ($\widetilde{x}$)**	**STD (σ)**	**Metric**	**AVG ($\widetilde{x}$)**	**STD (σ)**
Throughput (Mbps)	7.884	0.171	Throughput (Mbps)	8.412	0.225
Delay (ms)	2.980	0.226	Delay (ms)	2.748	0.274
Link Utilization (%)	8.848	0.005	Link Utilization (%)	8.354	0.006
Energy (W)	12.891	0.507	Energy (W)	12.629	0.525

Table 4.12: The system evaluation with the energy saving algorithm when disabling the traffic aggregation mechanism.

4.2.4.B Comparison

In Table 4.13 is given the results of the comparison made between disabling and partially enabling the energy saving algorithm without the traffic aggregation mechanism (see Appendix D). These results reflect the variation of each metric after applying the energy saving algorithm. Following, it will be given a brief overview of the results for each metric.

- **Throughput:** As shown in Table 4.13, the throughput of the network suffers a regression between 8% and 13%. These percentages represent a reduction in the network throughput of about 1.1 Mbps, in relation to the base system. Also, when the energy saving module is fully active, this regression is typically 1% smaller.

- **Delay:** As shown in Table 4.13, the packet delay of the network increases in this situation. The extra delay introduced lies between ∼ 3% and ∼ 35% in the tested networks. These percentages represent an increase in delay which ranges from 0.1 to 0.8 ms, in relation to the base system. Also, when the energy saving module is fully active, this regression is typically 4% smaller.

- **Link Utilization:** As shown in Table 4.13, the average utilization of each link increases when the energy saving algorithm is enabled. This is mostly caused by the fact that the less important network elements are the ones that will be turned off, thereby raising the average link utilization on the remaining network elements in ∼ 8.5%. As opposed to this, in the Géant network topology the average utilization of each link is decreased by ∼ 7%, because in this case there will be a lot of turned off links whose utilization will be zero. Also, when the energy saving module is fully active, this regression is typically 2% smaller.

- **Energy:** As shown in Table 4.13, the overall network energy consumption is reduced in no less than 28%. The results represent a decrease in the power consumption, which ranges from 3.5 W to 11.5 W in comparison to the base system. Also, when the energy saving module is fully active the reduction in the energy consumption is 0.3% higher.

Abilene Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-11.530	-8.810
Δ_{Delay} (%)	29.200	26.494
$\Delta_{\text{LinkUtilization}}$ (%)	8.787	8.500
Δ_{Energy} (%)	-28.985	-29.729
COST 239 Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-13.03	-9.40
Δ_{Delay} (%)	34.72	30.38
$\Delta_{\text{LinkUtilization}}$ (%)	7.75	8.94
Δ_{Energy} (%)	-53.52	-55.82
Géant Network Topology		
Metric	**IP**	**PSIRP**
$\Delta_{\text{Throughput}}$ (%)	-8.16	-7.17
Δ_{Delay} (%)	3.33	6.66
$\Delta_{\text{LinkUtilization}}$ (%)	-9.01	-5.75
Δ_{Energy} (%)	-46.53	-47.64

Table 4.13: The impact of the energy saving algorithm with the traffic aggregation mechanism disabled.

4.3 Evaluation Summary

In this section it will be given a brief overview and discussion over the most relevant evaluation results. In general, the evaluation results confirm that is possible to effectively reduce the overall network energy consumption by turning off the unused network elements.

As expected, there will be greater energy savings when the network is lightly loaded. But, even when it is heavily loaded it is possible to achieve a significant reduction in the power consumption without a major decrease in the performance of the network.

The results also show that without the node sleeping mechanism, the energy savings can be considered negligible. The reduction in the energy consumption caused by turning off the links does not compensate the loss of network performance.

The evaluation also proves that by using the traffic aggregation mechanism, it can be achieved a better tradeoff between energy savings and network performance. According to the evaluation results, when this mechanism is enabled the energy savings are higher and at the expense of less performance (throughput and delay).

Finally, by analyzing the obtained results it is now possible to answer to the questions that were enumerated in the beginning of this chapter.

- **How much energy it can be saved when using the energy saving module?** The results indicate that activating the energy saving module effectively lead to a reduction in the network energy consumption. The experimental results show that the base system in a low traffic scenario consumes from ∼12 W (4.32 kWh) to ∼25 W (9 kWh) of energy. In this situation, the proposed solution achieves energy savings from 3 W (1.08 kWh) to 12 W (4.32 kWh) of energy. On the other hand, in the heavy traffic scenario the base system consumes from ∼56 W (20.16 kWh) to ∼109 W (39.24 kWh) of energy. In this situation, the proposed solution achieves energy savings from 3 W (1.08 kWh) to 30 W (10.8 kWh) of energy. Finally, it is important to note that the overall energy consumption is highly dependent on the network topology.

- **What is the impact of the energy saving module in the performance of the network?** Applying energy saving mechanisms will inevitably reduce the performance of the network by decreasing the throughput and increasing the delay. The experimental results show that the base system in a low traffic scenario has a throughput that goes from 8.4 Mbps to 9.4 Mbps and delay from 1.4 ms to 2.9 ms. When enabling the energy saving module, the throughput is reduced between 600 kbps and 1 Mbps. Also, the delay will be increased from 0.1 ms to 0.65 ms. On the other hand, in the heavy traffic scenario the base system has a throughput that ranges from 6.1 Mpbs to 8.9 Mbps and the delay from 1.5 ms to 5.2 ms. With the energy saving module enabled, the throughput is reduced between 500 kbps and 1 Mbps. Also, the delay will be increased from 0.4 ms to 0.7 ms.

- **What is the tradeoff between network performance and energy savings? Is it worth?** As expected, the network performance decreases after enabling the energy saving module. The experimental results show that in a low traffic scenario the energy savings will be $\sim 45\%$ in average and at the cost of $\sim 9\%$ of network throughput. On the other hand, in a heavy traffic scenario the energy savings are $\sim 23\%$ in average and at the cost of $\sim 10\%$. Also, it is important to note that the energy consumption reduction in the Abilene network is much lower than the reduction in the other networks, since there is less network elements to disconnect. Finally, these results demonstrate that in fact it is possible to reduce the overall energy consumption without excessively affecting the performance of the network. The resulting impact caused by the energy saving mechanism in the network performance is not very meaningful, specially in the absence of very strict QoS requirements.

5 Conclusions and Future Work

Contents

5.1 Future Work

In future work, the remaining limitations (see section 3.2) of the system will be addressed. It should also be performed a further evaluation of the system in a wider set of network topologies and traffic conditions, giving a lot of focus to the robustness in the case of failures.

It will also be studied the applicability of the energy saving algorithm in conjunction with a distance-vector routing protocol. As opposed to link-state routing protocols, each network node only has partial knowledge of the network topology.

Finally, the energy saving algorithm will be implemented and evaluated in real networks. In this work, the energy saving algorithm was only evaluated by making use of network simulation.

5.2 Conclusions

There is a growing concern on avoiding network overload and ensuring security, which may lead to the rethinking of the current Internet paradigm. Nowadays the Internet is node-centric, e.g. to allow remote connections to a specific node in the network. The new trend is to focus on information instead of network nodes. In this new paradigm the major concern is to exchange information without worrying about the network nodes that are used to exchange it. Today's users are only concerned on getting the information that they want as fast as possible. Because of this, a significant amount of research is being done to design new approaches for the Internet architecture, which are based on the information-centric paradigm, e.g. 4WARD and PSIRP.

The current Internet not only faces the problems of network overload and security, but also mobility and energy consumption. All of these issues are very important and need to be addressed in the Internet architecture. The problem of the current Internet architecture is the lack of flexibility, making it very difficult to address these new concerns without increasing the complexity of the architecture itself. In this context the only solution is to develop a completely new architecture, based on the "Clean Slate Design" principle, which allows a better integration of these concerns in a future Internet architecture.

Lately, the increase of the energy consumption is receiving a lot of attention because of environmental and economical issues. Because of this, it is being researched energy saving techniques that will allow a reduction in the energy consumption. The most widely adopted energy saving technique is to put network elements into sleep mode.

This work makes a review on current proposals for the future Internet, including new architectural and energy saving solutions. Based on this study, it was implemented a solution which contemplates two architectures, one that is based on the current Internet model and the other that is based on the PSIRP architecture. It was also implemented an energy saving module which embeds energy awareness in both implemented architectures by turning off unused network elements.

The implemented energy saving solution makes traffic engineering decisions to aggregate traffic to most used links, which will allow the possibility of inducing underused links to an idle mode. The unused network elements, nodes or links, can be turned off if all the remaining network nodes are still reachable without them. It is also proposed a ranking mechanism that classifies the importance of a network element. This mechanism is extremely important to achieve a good trade off between energy savings and network performance, mainly because it will turn off in first place the network elements that are less important to the packet delivery process.

The evaluation of the solution shows that significant energy savings can be achieved in a low traffic scenario without too much impact in the network performance. In a heavy traffic scenario, the proposed solution also manages to reduce the energy consumption but in a smaller percentage. Finally, the evaluation demonstrates that using traffic aggregation will allow to achieve a better tradeoff between energy savings and network performance. It is also shown that without the node sleeping mechanism it is not possible to significantly decrease the overall energy consumption of the network.

Bibliography

[1] N. Niebert, S. Baucke, I. EI-Khayat, M. Johnsson, B. Ohlman, H. Abramowicz, K. Wuenstel, H. Woesner, J. Quittek, and L. Correia. The way 4ward to the creation of a future internet. *IEEE*, 2008.

[2] G. Bouabene, C. Jelger, C. Tschudin, S. Schmid, A. Keller, and M. May. The autonomic network architecture. *IEEE Journal on Selected Areas in Communications*, 28(1), January 2010.

[3] G. Tselentis. *Towards the Future Internet*, pages 75 –84. IOS Press, 2010.

[4] D. Clark, R. Braden, A. Falken, and V. Pingali. Fara: Reorganizing the addressing architecture. *ACM Sigcomm Workshops*, August 2003.

[5] X. Yang, D. Clark, and A. Berger. Nira: A new inter-domain routing architecture. *IEEE/ACM Transactions on Networking*, 15(4), August 2007.

[6] D. Katabi, M. Handley, and C. Rohrs. Congestion control for high bandwidth-delay product networks. *ACM Sigcomm*, August 2002.

[7] C. Jinzhou, W. Chunming, J. Ming, and Z. Dong. A review of future internet research programs and possible trends. *IEEE*, 2010.

[8] D. Meyer, L. Zhang, and K. Fall. Report from the iab workshop on routing and addressing. *RFC 4984*, September 2007.

[9] A. Bianzino, C. Chaudet, D. Rossi, and J. Rougier. A survey of green networking research. *Communications Surveys Tutorials, IEEE*, PP(99):1 –18, 2010.

[10] Global e-Sustainability Initiative (GeSI). Smart 2020 report: Global ict solution case studies. Technical report, http://www.smart2020.org/publications/, 2008.

[11] A. Feldmann. Internet clean-slate design: What and why? *ACM Sigcomm Computer Communication Review*, 37(3), July 2007.

[12] N. Ahmed and S. Keshav. A successive refinement approach to wireless infrastructure network deployment. *IEEE*, 2006.

[13] S. Srinivasan and A. Azadmanesh. Data aggregation in static adhoc networks. *IEEE*, 2008.

[14] H. Li and M. Singhal. A scalable routing protocol for ad hoc networks. *IEEE*, 2005.

[15] E. Mahdipour, A. Rahmani, and E. Aminian. Performance evaluation of destination-sequenced distance-vector (dsdv) routing protocol. *IEEE International Conference on Future Networks*, 2009.

[16] C. Perkins and E. Royer. Ad-hoc on-demand distance vector routing. *IEEE*.

[17] I. Stoica, R. Morris, D. Karger, M. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. *ACM Sigcomm*, August 2001.

[18] A. Rowstron and P. Druschel. Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems. *ACM International Conference on Distributed Systems Platforms*, 2001.

[19] B. Zhao, L. Huang, J. Stribling, S. Rhea, A. Joseph, and J. Kubiatowicz. Tapestry: A resilient global-scale overlay for service deployment. *IEEE Journal on Selected Areas in Communications*, 22(1), January 2004.

[20] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content addressable network. *ACM Sigcomm*, August 2001.

[21] B. Maniymaran, M. Bertier, and A. Kermarrec. Build one, get one free: Leveraging the coexistence of multiple p2p overlay networks. *IEEE*, 2007.

[22] H. Bos, E. Jonsson, E. Djambazova, K. Dimitrov, S. Ioannidis, E. Kirda, and C. Kruegel. Anticipating security threats to a future internet. 2008.

[23] P. Stuckmann and R. Zimmermann. European research on future internet design. *IEEE Wireless Communications Magazine*, October 2009.

[24] G. Tselentis. *Towards the Future Internet*, pages 91 –101. IOS Press, 2009.

[25] M. Achemlal, J. Pailles, and C. Gaber. Building trust in virtualized networks. *IEEE*, 2010.

[26] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. Sip: Session initiation protocol. *RFC 3261*, June 2002.

[27] P. Aranda, T. Biermann, D. Bursztynowski, T. Janaszka, L. Gomez, A. Gunnar, N. Hegde, F. Idzikowski, H. Karl (UPB), K. Miller (TUB), A. Neto, Z. Polgar, P. Poyhonen, I. Psaras, S. Randriamasy, J. Roberts, N. Wang, and H. Woesner. Description of generic path mechanism. *4WARD Deliverable D-5.2.0*, 2007.

[28] P. Gutierrez, S. Baucke, R. Bless, M. Bourguiba, J. Cabero, L. Caeiro, J. Carapinha, M. Dianati, C. Gorg, I. Houidi, L. Izaguirre, Y. Lemieux, W. Louati, O. Maennel, L. Mathy, M. Melo, J. Nogueira, P. Papadimitriou, S. Sánchez, A. Serrador, I. Seskar, J. Tiemann, A. Udugama, C. Werle, F. Wolff, A. Wundsam, Y. Zaki, D. Zeghlache, and L. Zhao. Virtualisation approach: Evaluation and integration. *4WARD Deliverable D-5.3.1*, 2007.

[29] S. Dudler. New protocols and applications for the future internet. Master's thesis, Swiss Federal Institute of Technology (ETH), Zurich, March 2008.

[30] P. Mockapetris and K. Dunlap. Development of the domain name system. *ACM Sigcomm*, pages 123 –133, 1988.

[31] M. Siekkinen, V. Goebel, T. Plagemann, K. Skevik, M. Banfield, and I. Brusic. Beyond the future internet – requirements of autonomic networking architectures to address long term future networking challenges. *Proceedings of the 11th IEEE International Workshop on Future Trends of Distributed Computing Systems*, 2007.

[32] D. Clark, K. Sollins, J. Wroclawski, D. Katabi, J. Kulik, X. Yang, R. Braden, T. Faber, A. Falk, V. Pingali, M. Handley, and N. Chiappa. New arch: Future generation internet architecture. Technical report, Air Force Research Laboratory, Rome, NY, December 2003.

[33] L. Gao. On inferring autonomous system relationships in the internet. *IEEE/ACM Transactions on Networking*, 9(6):733 –745, December 2001.

[34] M. Haq, M. Perwaz, and K. Ahmed. Compact and user-friendly representation of ipv6 addressing approach and masking. *IEEE*, 2009.

[35] H. Balakrishnan, K. Lakshminarayanan, S. Ratnasamy, S. Shenker, I. Stoica, and I. Walfish. A layered naming architecture for the internet. *ACM Sigcomm*, pages 343 –352, 2004.

[36] N. Fotiou, D. Trossen, and G. Polyzos. Illustrating a publish-subscribe internet architecture. *Telecommunication Systems, Springer*, 2010.

[37] M. Ain, S. Tarkoma, D. Trossen, P. Nikander, T. Burbridge, A. Zahemszky, J. Rajahalme, D. Lagutin, M. Särelä, J. Riihijärvi, and T. Rinta-aho. Conceptual architecture of psirp including subcomponent descriptions. *FP7-INFSO-ICT-216173-PSIRP-D2.2*, 2008.

[38] N. Fotiou, P. Nikander, D. Trossen, and G. Polyzos. Developing information networking further: From psirp to pursuit. *International ICST Conference on Broadband Communications*, October 2010.

[39] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol label switching architecture. *RFC 3031*, January 2001.

[40] Y. Zhang and T. Henderson. An implementation and experimental study of the explicit control protocol (xcp). *IEEE*, 2:1037 –1048, 2005.

[41] K. Ramakrishnan and S. Floyd. Proposal to add explicit congestion notification to ip. *RFC 2481*, January 1999.

[42] M. Baldi and Y. Ofek. Time for a "greener" internet. *IEEE*, 2009.

[43] H. Mellah and B. Sansò. Review of facts, data and proposals for a greener internet. *ICST Broadnets*, 2009.

[44] R. Bolla, R. Bruschi, F. Davoli, and F. Cucchietti. Energy efficiency in the future internet: A survey of existing approaches and trends in energy-aware fixed network infrastructures. *IEEE Communications Surveys*, 2010.

[45] C. Gunaratne, K. Christensen, B. Nordman, and S. Suen. Reducing the energy consumption of ethernet with adaptive link rate (alr). *IEEE Transactions on Computers*, 57(4), April 2008.

[46] B. Sansò and H. Mellah. On reliability, performance and internet power consumption. *IEEE*, 2009.

[47] A. Gladisch, C. Lange, and R. Leppla. Power efficiency of optical versus electronic access networks. *IEEE*, 2(143), September 2008.

[48] G. Papadimitriou, C. Papazoglou, and A. Pomportsis. Optical switching: Switch fabrics, techniques, and architectures. *IEEE Journal of Ligthwave Technology*, 21(2), February 2003.

[49] Y. Li, W. Li, and C. Jiang. A survey of virtual machine system: Current technology and future trends. *IEEE*, 2010.

[50] M. Anisetti, V. Bellandi, A. Colombo, M. Cremonini, E. Damiani, F. Frati, J. Hounsou, and D. Rebeccani. Learning computer networking on open paravirtual laboratories. *IEEE Transactions on Education*, 50(4), November 2007.

[51] B. Zhang, X. Wang, R. Lai, L. Yang, Y. Luo, X. Li, and Z. Wang. A survey on i/o virtualization and optimization. *IEEE*, 2010.

[52] C. Messom, A. Sarrafzadeh, A. Gerdelan, M. Johnson, and J. Shanbehzadeh. Operating system virtualization to support e-learning with affective intelligent tutoring systems. *IEEE*, 2008.

[53] M. Baldi and G. Marchetto. Pipeline forwarding of packets based on a low-accuracy network-distributed common time reference. *IEEE/ACM Transactions on Networking*, 17(9), December 2009.

[54] M. Baldi, J. Martin, E. Masala, and A. Vesco. Quality-oriented video transmission with pipeline forwarding. *IEEE Transactions on Broadcasting*, 54(3), September 2008.

[55] M. Allmany, K. Christensenz, B. Nordman, and V. Paxsony. Enabling an energy-efficient future internet through selectively connected end systems. *ACM Sigcomm Hotnets*, November 2007.

[56] A.P. Bianzino, C. Chaudet, D. Rossi, J. Rougier, and S. Moretti. The green-game: Striking a balance between qos and energy saving. In *Teletraffic Congress (ITC), 2011 23rd International*, pages 262 –269, September 2011.

[57] L. Chiaraviglio, M. Mellia, and F. Neri. Reducing power consumption in backbone networks. In *Communications, 2009. ICC '09. IEEE International Conference on*, pages 1 –6, June 2009.

[58] S. Gao, J. Zhou, T. Aya, and N. Yamanaka. Reducing network power consumption using dynamic link metric method and power off links. June 2009.

[59] A. Cianfrani, V. Eramo, M. Listanti, M. Marazza, and E. Vittorini. An energy saving routing algorithm for a green ospf protocol. *IEEE*, 2010.

[60] K. Ho and C. Cheung. Green distributed routing protocol for sleep coordination in wired core networks. In *Networked Computing (INC), 2010 6th International Conference on*, pages 1 –6, May 2010.

[61] R. Garroppo, S. Giordano, G. Nencioni, and M. Pagano. Energy aware routing based on energy characterization of devices: Solutions and analysis. *IEEE*, June 2011.

[62] S. Moretti and F. Patrone. Transversality of the shapley value. *TOP*, 16:1 –41, 2008.

[63] P. Jokela, A. Zahemszky, C. E. Rothenberg, S. Arianfar, and P. Nikander. Lipsin: line speed publish/subscribe inter-networking. *ACM SIGCOMM Computer Communication Review*, 39(4):195 –206, August 2009.

[64] A. Bianzino, C. Chaudet, F. Larroca, D. Rossi, and J. Rougier. Energy-aware routing: A reality check. In *GLOBECOM Workshops, 2010 IEEE*, pages 1422 –1427, December 2010.

[65] D. Chen, L. Lü, M. Shang, Y. Zhang, and T. Zhou. Identifying influential nodes in complex networks. *Physica A: Statistical Mechanics and its Applications*, 391(4):1777 –1787, February 2012.

[66] M. Yu, M. Thottan, and L. Li. Latency equalization as a new network service primitive. *Networking, IEEE/ACM Transactions on*, 20(1):125 –138, February 2012.

[67] P. Batchelor, B. Daino, P. Heinzmann, D. R. Hjelme, R. Inkret, H. A. Jager, and et al. Study on the implementation of optical transparent transport networks in the european environment - results of the research project cost 239. *Photonic Network Communications*, 2:15 –32, 2000.

[68] Tarik Čičić. On basic properties of fault-tolerant multi-topology routing. *Computer Networks*, 52(18):3325 –3341, December 2008.

A

System evaluation in a light traffic scenario

Abilene Network Topology

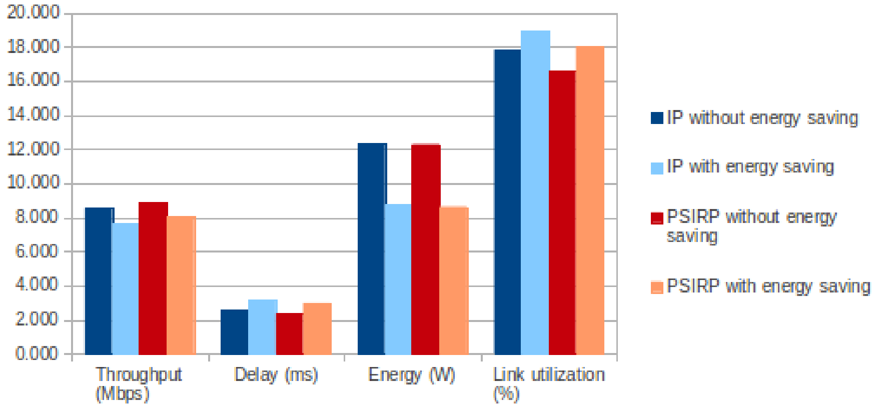


Figure A.1: Metric evaluation in a light traffic scenario using the Abilene topology.

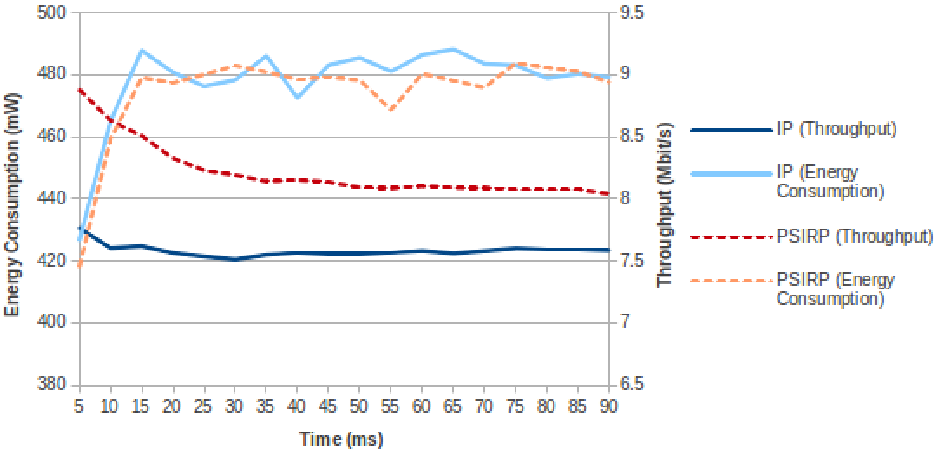


Figure A.2: Energy consumption and throughput variation over time using the Abilene topology.

COST-239 Network Topology

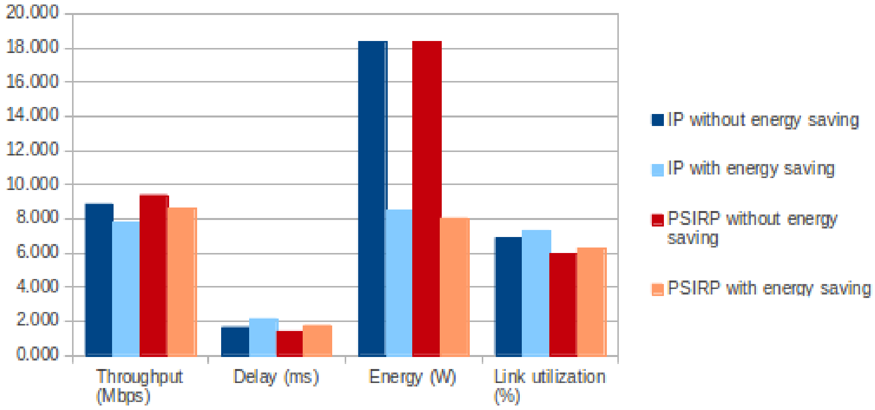


Figure A.3: Metric evaluation in a light traffic scenario using the COST-239 topology.

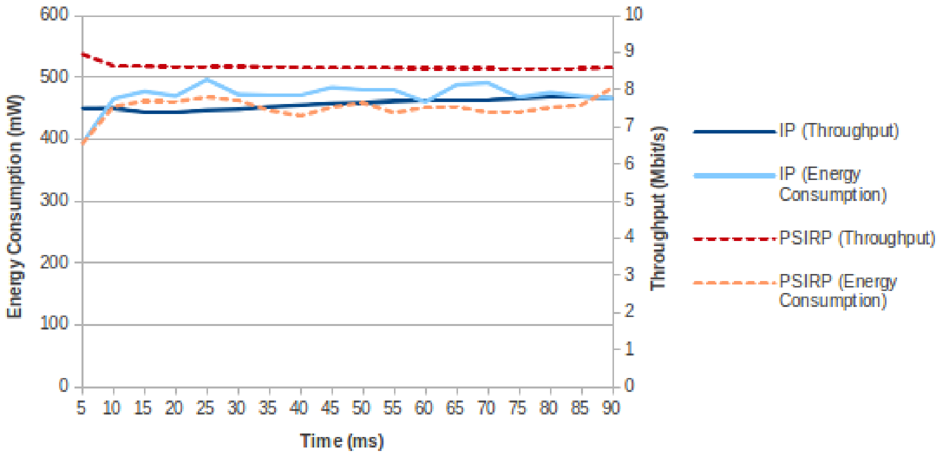


Figure A.4: Energy consumption and throughput variation over time using the COST-239 topology.

Géant Network Topology

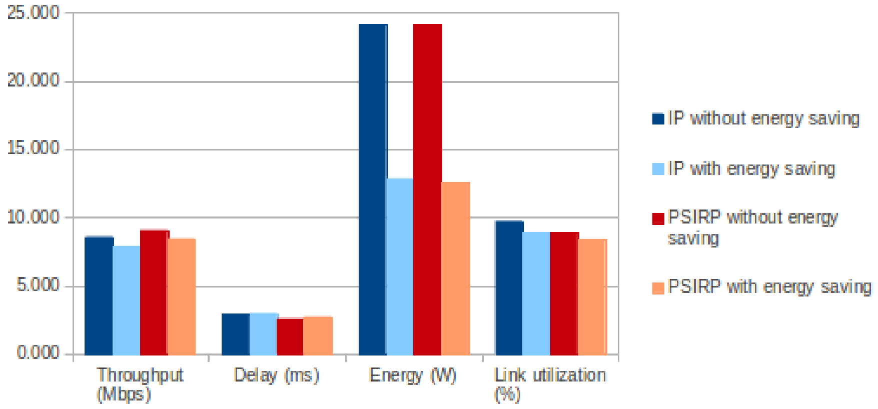


Figure A.5: Metric evaluation in a light traffic scenario using the Géant topology.

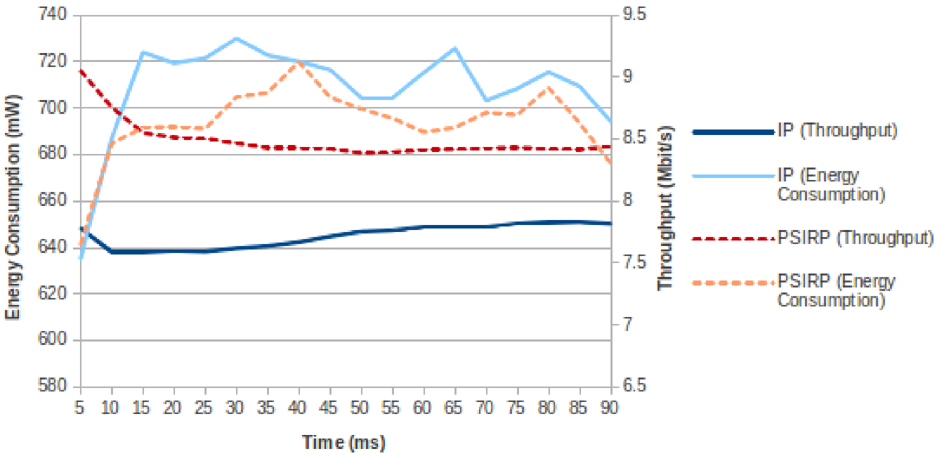


Figure A.6: Energy consumption and throughput variation over time using the Géant topology.

B

System evaluation in a high traffic scenario

Abilene Network Topology

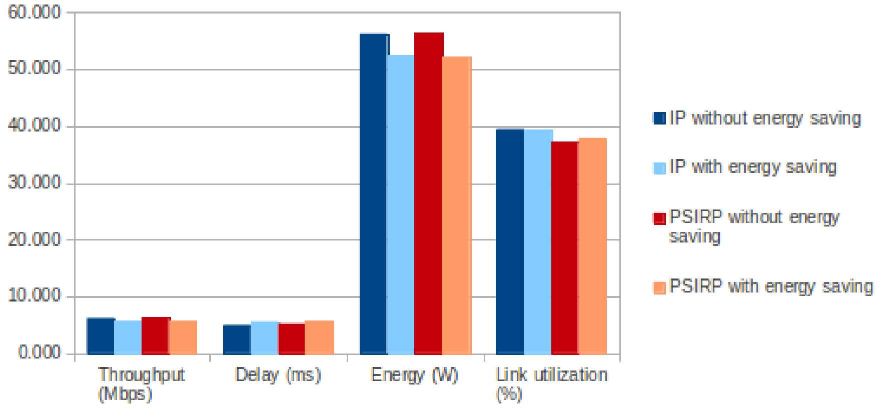


Figure B.1: Metric evaluation in a high traffic scenario using the Abilene topology.

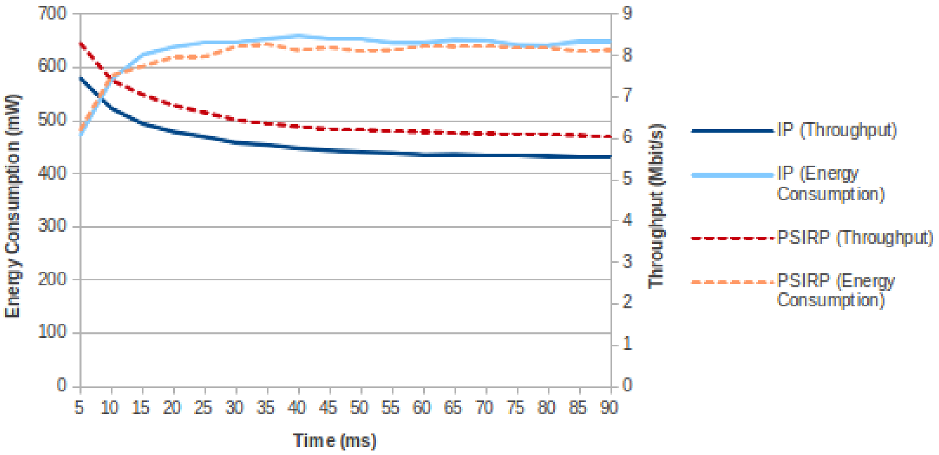


Figure B.2: Energy consumption and throughput variation over time using the Abilene topology.

COST-239 Network Topology

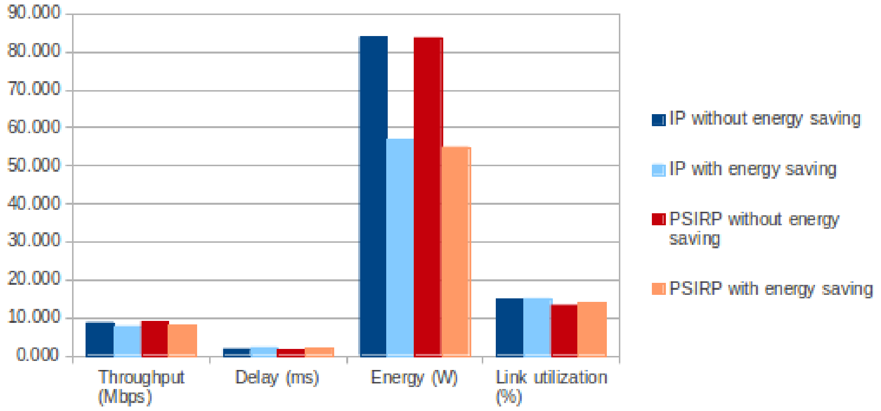


Figure B.3: Metric evaluation in a high traffic scenario using the COST-239 topology.

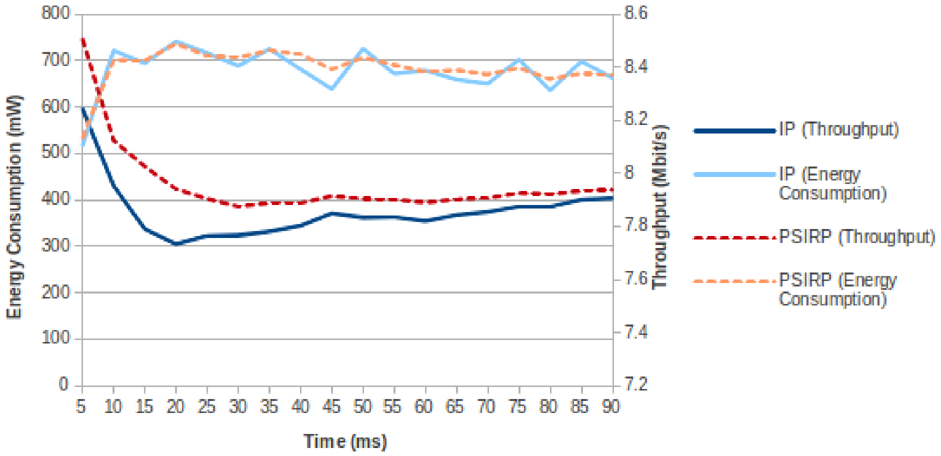


Figure B.4: Energy consumption and throughput variation over time using the COST-239 topology.

Géant Network Topology

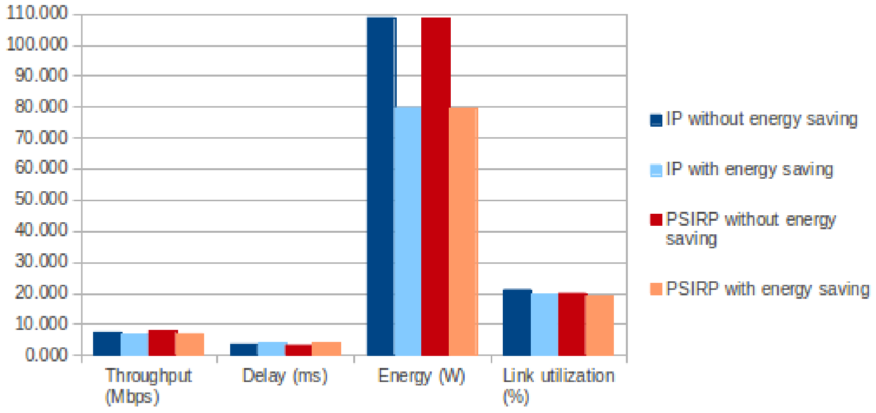


Figure B.5: Metric evaluation in a high traffic scenario using the Géant topology.

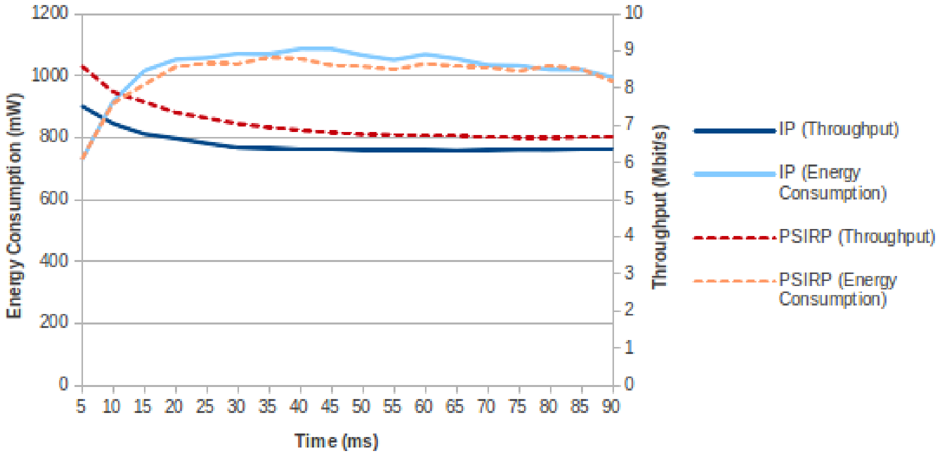


Figure B.6: Energy consumption and throughput variation over time using the Géant topology.

C

System evaluation with the node sleeping mechanism disabled

Abilene Network Topology

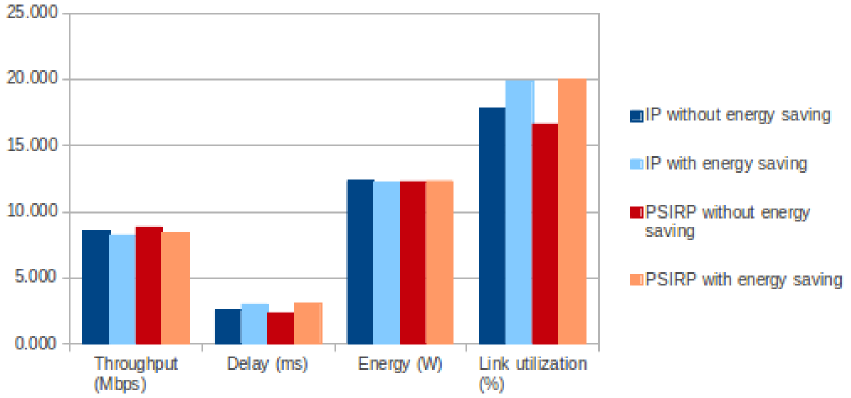


Figure C.1: Metric evaluation in a light traffic scenario using the Abilene topology and with the node sleeping mechanism disabled.

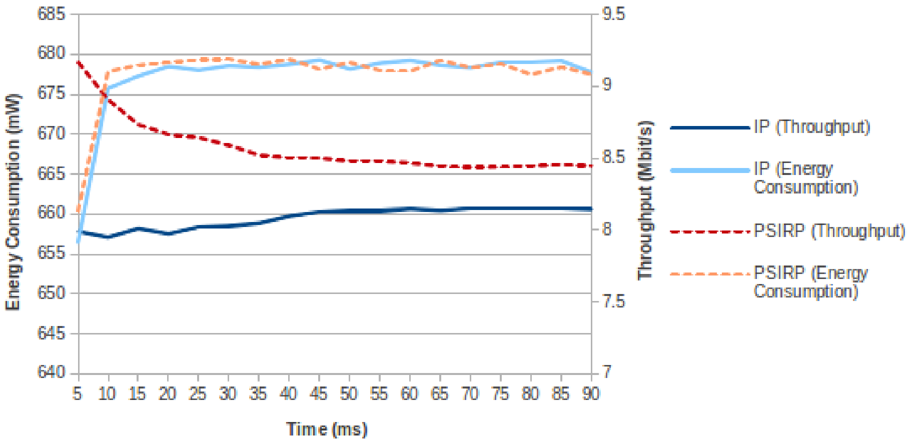


Figure C.2: Energy consumption and throughput variation over time using the Abilene topology.

COST-239 Network Topology

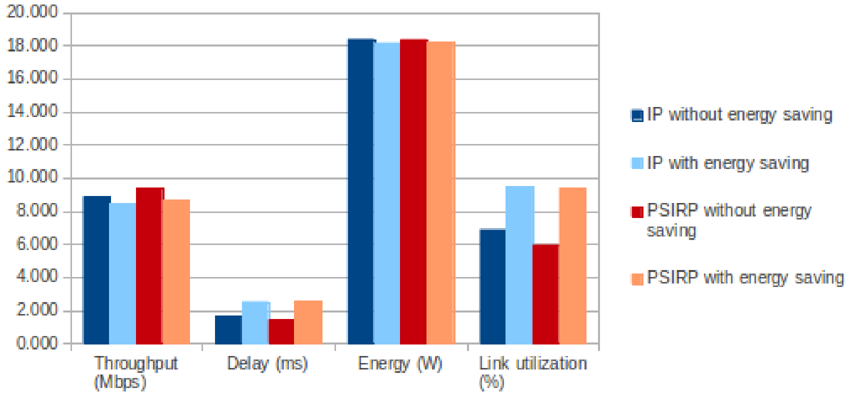


Figure C.3: Metric evaluation in a light traffic scenario using the COST-239 topology and with the node sleeping mechanism disabled.

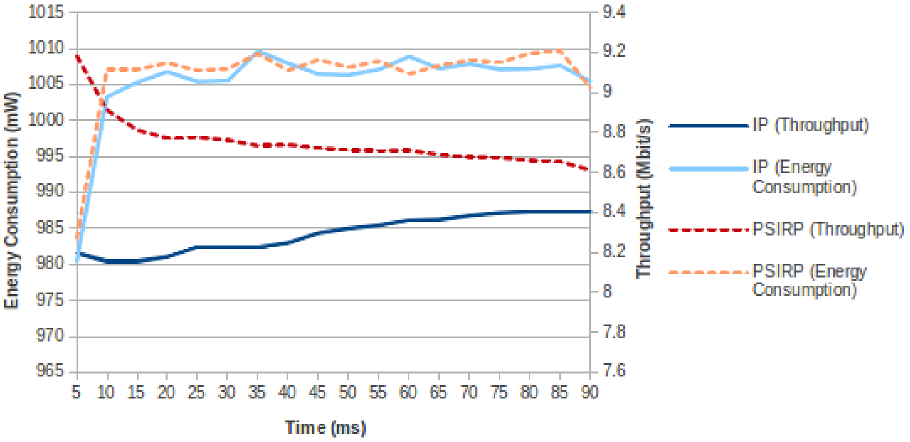


Figure C.4: Energy consumption and throughput variation over time using the COST-239 topology.

Géant Network Topology

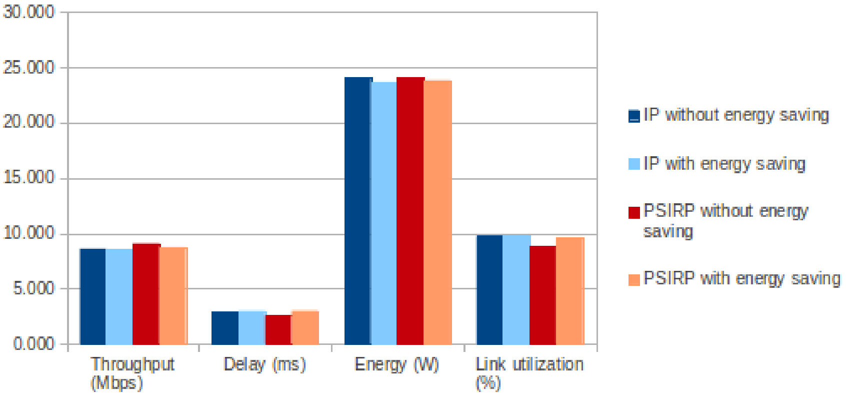


Figure C.5: Metric evaluation in a light traffic scenario using the Géant topology and with the node sleeping mechanism disabled.

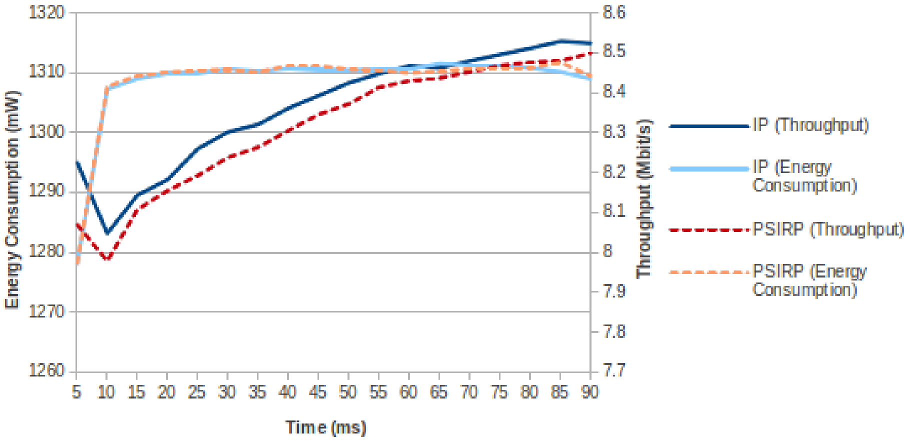


Figure C.6: Energy consumption and throughput variation over time using the Géant topology.

D

System evaluation in a scenario with the traffic aggregation mechanism disabled

Abilene Network Topology

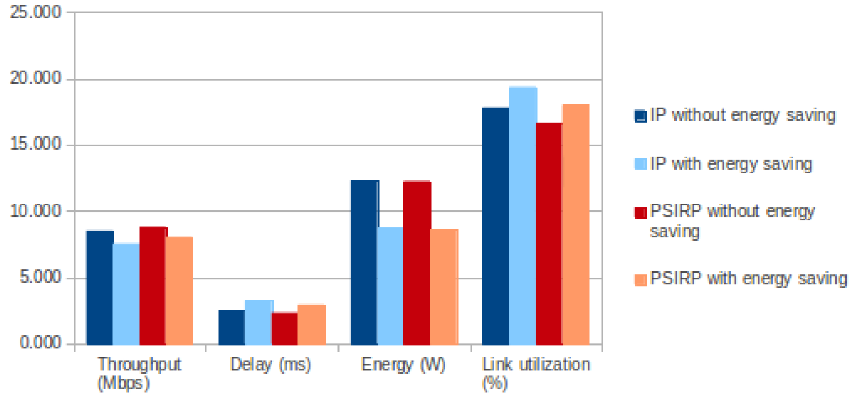


Figure D.1: Metric evaluation in a light traffic scenario using the Abilene topology and with the traffic aggregation mechanism disabled.

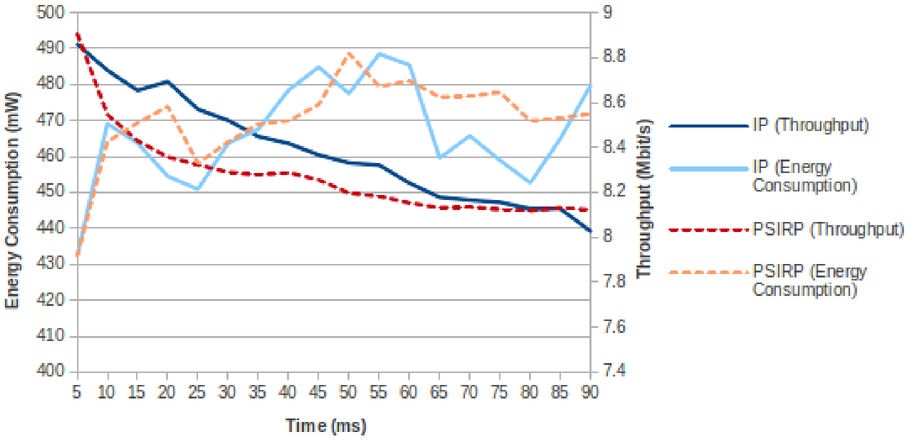


Figure D.2: Energy consumption and throughput variation over time using the Abilene topology.

COST-239 Network Topology

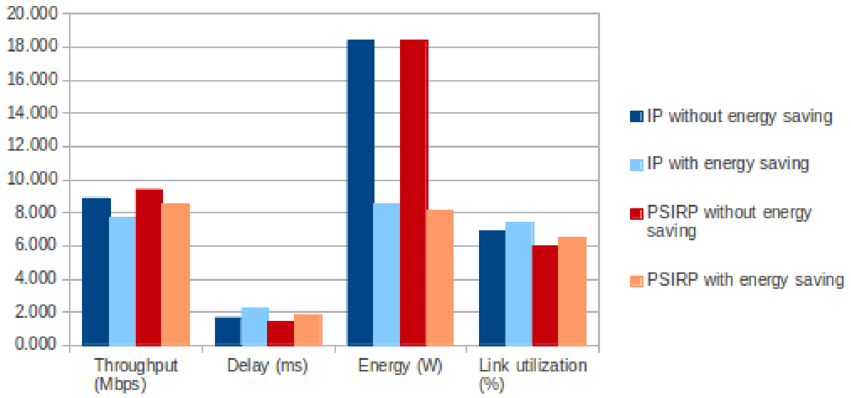


Figure D.3: Metric evaluation in a light traffic scenario using the COST-239 topology and with the traffic aggregation mechanism disabled.

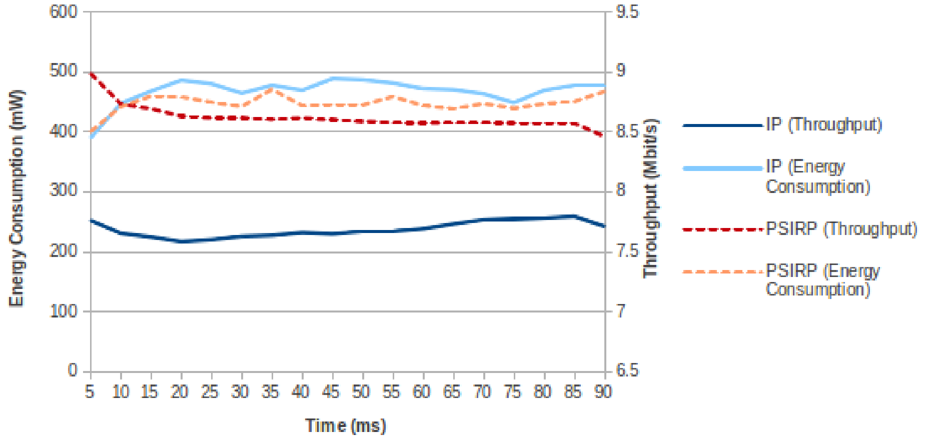


Figure D.4: Energy consumption and throughput variation over time using the COST-239 topology.

Géant Network Topology

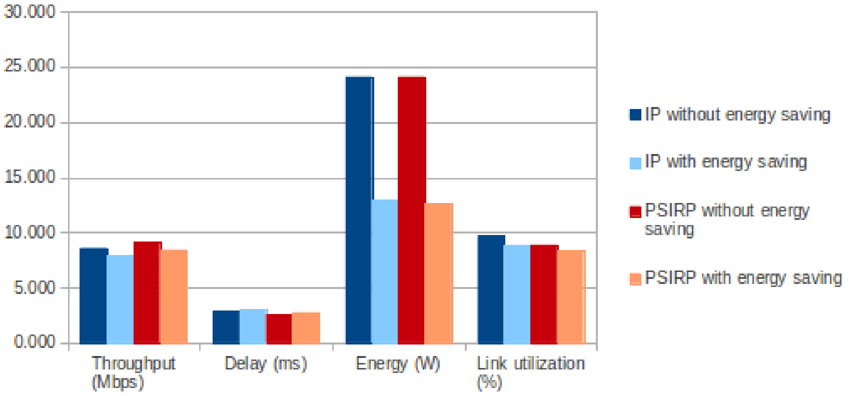


Figure D.5: Metric evaluation in a light traffic scenario using the Géant topology and with the traffic aggregation mechanism disabled.

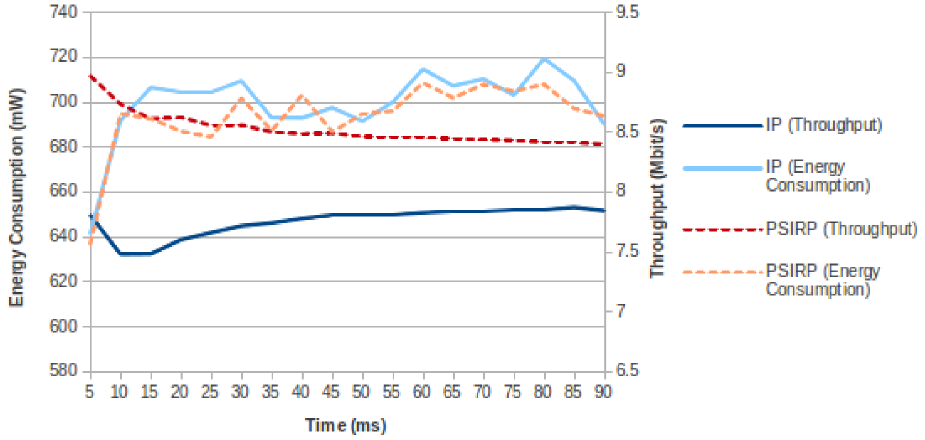


Figure D.6: Energy consumption and throughput variation over time using the Géant topology.