UNIUNEA EUROPEANĂ

Fondul Social European POSDRU 2007-2013

Instrumente Structurale 2007-2013

OIPOSDRU

MINISTERUL EDUCAȚIEI ȘI CERCETĂRII ȘTIINȚIFICE

UNIVERSITATEA TEHNICĂ CLUJ-NAPOCA

**Investeşte în oameni!**

**Proiect cofinanţat din Fondul Social European prin Programul Operaţional Sectorial pentru Dezvoltarea Resurselor Umane 2007 – 2013**
**Axa prioritară:** 1„Educaţia şi formarea profesională în sprijinul creşterii economice şi dezvoltării societăţii bazate pe cunoaştere”
**Domeniul major de intervenţie:** 1.5 „Programe doctorale şi postdoctorale în sprijinul cercetării”
**Titlul proiectului: „Parteneriat inter-universitar pentru excelenta in inginerie - PARTING”**
**Cod Contract: POSDRU/159/1.5/S/137516**
**Beneficiar: UniversitateaTehnică din Cluj-Napoca**

**FACULTATEA DE ELECTRONICĂ, TELECOMUNICAȚII ȘI TEHNOLOGIA INFORMAȚIEI**

***Ing. Iustin-Alexandru IVANCIU***

# REZUMAT

# TEZĂ DE DOCTORAT

# ACTIVE MEASUREMENTS FOR ROUTING IN CLOUD-BASED NETWORKS

**Conducător științific,**

**Prof.dr.ing. Virgil DOBROTĂ**

## Comisia de evaluare a tezei de doctorat:

PREȘEDINTE: - Prof.dr.ing. ***Aurel Vlaicu*** - Universitatea Tehnică din Cluj - Napoca

MEMBRI: - Prof.dr.ing. ***Virgil Dobrotă*** - conducător științific, Universitatea Tehnică din Cluj - Napoca

- Prof.dr.ing. ***Eugen Borcoci*** - referent, Universitatea "Politehnica" din București
- Prof.dr.ing. ***Radu Vasiu*** - referent, Universitatea "Politehnica" din Timișoara
- Conf.dr.ing. ***Daniel Zinca*** - referent, Universitatea Tehnică din Cluj - Napoca

# Table of Contents

# 1 Introduction

Nowadays, content across the Internet is accessed with little reference to the underlying host infrastructure, formed by data centers which are maintained by content providers. In order to provide such a locality transparence, a new model for the provision of computing services, called Cloud Computing, was introduced. In this model, enterprises put their applications in the cloud without having to worry about their location or delivery. By using even a slice of the high computational power over a scalable network of nodes, enterprises can reduce or even eliminate costs associated with the internal infrastructure needed to provide their services. Moreover, cloud computing allows the provision of resources according to the enterprise needs. Consequently, cloud users may rent resources as they become necessary, in a scalable and elastic way.

Routing is defined as the process of selecting the path along which the traffic is sent, or in other words, directing a packet from a source node to a destination node. The routing process involves two steps. The first step is determining the best path by means of routing algorithms, while the second step refers to switching the packet from the input interface to the output interface. Routing is an essential aspect of a fully-functional network, be it cloud-based or not, as it influences not only the stability, but also the performance of the network. Therefore, if the routing process is not performed properly, a decrease in the quality of service will incur. In order to obtain a high routing efficiency it is necessary to increase the throughput as much as possible while maintaining the average packet delay as low as possible. Achieving a tradeoff between these two conflicting requirements calls for a precise knowledge of network parameters such as available transfer rate, end-to-end delay, link quality, etc.

A real-time estimation of the above mentioned parameters is also crucial for the efficient management of network resources and providing QoS (Quality of Service) guarantees to the users. As several users share these resources, the link capacity may prove insufficient and thus cause network congestion. As for the delay, a high value of this parameter leads to significant decrease in the user experience. However important, measuring these parameters is not an easy task. Passive monitoring requires information from all nodes in the path and therefore is not achievable in practice. Consequently, active probing is employed. It is based on injecting probe traffic into the network and analyzing the effect of cross-traffic on these probes. As opposed to passive monitoring, active probing techniques only require access to sender and receiver hosts.

Traditional routing algorithms may not be suitable for cloud-based networks. Therefore, a new approach, tailored for the requirements of such networks, is needed. The solution is to develop custom versions of the routing algorithms which take into account these new parameters. Moreover, the decision to use one algorithm or another should be performed dynamically by means of software.

The main characteristics of routing in cloud-based networks are:

- The need for adaptation – the condition of the nodes is dynamic and heterogeneous.
- The need for interoperability – vertical interoperability is used for cross-layer information exchange while horizontal interoperability is necessary for intra and inter-cloud communication.
- The need for optimization – the optimization process is dynamic and specific for each type of network.

A lot of industry and research communities are fascinated nowadays by the huge potential of Software-Defined Networking (SDN) and its applications for the Future Internet. SDN was developed to facilitate innovation by enabling a simple, programmatic control of the network data-path. The separation of the hardware responsible for forwarding from the control logic not only allows for the easier deployment of new protocols and applications but also enables network management and the consolidation of middle boxes into software control. Rather than enforcing policies and running protocols on numerous individual hosts, the entire network now comes down to simple forwarding hardware and decision-making controllers. OpenFlow-based technologies are a good example of how an innovative solution could be rapidly prototyped and deployed. This thesis presents an adaptive routing solution in a private cloud-based testbed with Open vSwitches controlled by software. Implementation issues are discussed and a novel solution for automating the deployment of applications, based on SPLAY, is introduced.

## 1.1 Active Measurements

Available Transfer Rate (ATR) and One-Way Delay (OWD) are fundamental metrics when describing the performance of a network path. The capability of measuring them proves important in several contexts such as server selection, network monitoring or service level agreement verification. Different tools have been developed in order to measure these parameters and these tools are either passive or active. Passive monitoring does not create or modify network traffic. Instead, it captures existing traffic and analyzes it in order to provide the required information. The advantage of this method is that the measuring process is non-intrusive. The downside is that no measurements can be performed in the absence of network traffic. Active measurements rely on injecting customized probes into the network and observing their behavior. As such, active probing techniques can be used on links with no existing traffic. However, the amount of probing traffic must be controlled in order not to interfere with the measurement process.

ATR can be defined as the minimum available transfer rate for all links $j$ along the path: $ATR = min_j (C_j – X_j)$ and it is determined by the tight link or bottleneck link which is the link with the smallest value of the available transfer rate. $C_j$ is the throughput capacity of each link $j$ in a network path and represents the amount of bytes successfully transmitted over the link per unit time while $X_j$ is the cross traffic or link load.

Some of the ATR estimation tools rely on a technique called the Probe Gap Model (PGM). It consists in sending a pair of packets with a predefined gap between them. This gap is measured at the end of the path and compared to the initial gap. If the network is slow or cross traffic is high the second packet in the pair will be delayed and the gap will increase. The difference between the initial gap and the gap measured at the end of the path is called dispersion and it increases with every queuing delay. PGM can be used to measure both the capacity and the available transfer rate. When measuring the capacity, the effect of cross traffic is minimized by using a small initial gap and the capacity is then determined by observing the dispersion. If the ATR is estimated, the dispersion is used to measure the rate of cross traffic on the tight link and then subtract it from the capacity of the bottleneck. The downside of this method is that it assumes the tight link occurs at the bottleneck which is not always the case.

OWD of packets is also an important metric for the performance of a network path as it provides information about both the state of the network and application performance. In the first case, the values of transmission, propagation and queuing delays can be inferred from this metric. OWD variations are also an indicator of congestion or route changes within a network. Regarding application performance, OWD needs to be taken into account when sizing the buffers for real time communication. A large value of the OWD might mean the loss of interactivity in such applications.

Nowadays, more and more researchers deal with different approaches regarding the measurement of OWD. Active measurement techniques are based on sending sequences of probe packets from one end of the monitored network to the other end. Each probe packet is marked with a timestamp immediately before its departure from the source. After its arrival to the destination, the OWD can be computed as the difference between the sender timestamp and the time measured by the receiver. However, the previous method is valid if and only if the two end-to-end hosts are perfectly synchronized [Wan03]. Unfortunately a perfect synchronization between source and destination cannot be guaranteed, although research in this field in in progress.

Other factors which influence the accuracy of the OWD measurement are the operating systems (OSs, interruptions), packetization and packet compression [Sim01]. For example, if the OS of the source has a tick period of 10 ms, this adds 10 ms of uncertainty to any time value measured. Moreover, as this time measurement is performed in software, the times collected are referred to as “host times”. However, the definition of OWD mandates the use of “wire times”, which represent the time instances when the test packet leaves and arrives at the network interfaces. The differences between the “host” and “wire” times can only be estimated and compensated to some extent, thus adding to the uncertainty of the measurement. Finally, in the case of an overloaded link, an increasing OWD trend will occur due to network congestion. During congestion, the packets will suffer extra delay as the packets that were previously queued to that interface need to be transmitted first.

A common way to avoid the previously described issues is to estimate the OWD as one half of the Round Trip Time (RTT). RTT is defined as the time interval between the packet transmission and the sender receiving an acknowledgement message for the same packet from the receiver. The measurement of the RTT can be easily obtained by means of ICMP Request/Reply packets. Synchronization between the two hosts is no longer required as the measurements is only performed at one end. The main drawback of this method consists in the asymmetry of the network paths. In packet-switched networks, the path from source to destination may be different than the path from the destination back to source. Moreover, application performance usually depends on the characteristics of a network in one direction. Therefore, deriving one-way metrics from round trip measurements is not a correct approach [Gur01].

## 1.2 Cloud-Based Networks

Cloud computing is a model used to enable ubiquitous, convenient, on-demand network access to a variety of shared and configurable computing resources (e.g., networks, servers, storage, applications, and services). Not only can the cloud be rapidly provisioned and released, but this can be done with minimal management effort or service provider interaction [Pat12] [Zho10] [Mel11].

The essential features of cloud computing refer to:

- **On-demand self-service** – consumers of cloud-computing services are provided with on-demand and instant access to resources. Therefore, the request, payment, and use of services must be possible without human operator intervention.
- **Broad network access** – shared resources are available over the network and accessed through different client platforms like mobile phones, tablets, laptops, and workstations.

- **Resource pooling** – the physical or virtual resources are dynamically assigned and reassigned according to consumer demand. A sense of location independence is introduced as the consumer has no knowledge of the exact location of these resources.
- **Rapid elasticity** – capabilities can be provisioned or released automatically making the model very flexible.
- **Measured service** – cloud systems automatically control and optimize resource use by means of a metering capability, thus assuring transparency for both provider and users.
- **Availability of computing resources** – computing resource provisioning does not have to be planned ahead since access to these resources is available anytime by means of demand.
- **No up-front commitment** – hardware and software resources may be increased only when needed thus eliminating heavy, upfront investments.
- **Short-term pay for use** – cloud services may be paid for on a short-time basis.

According to the abstraction level of the capability provided and the service model of providers, cloud computing services are divided into three layers: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS).

Infrastructure as a Service (IaaS) offers virtualized resources (computation, storage and communication) on demand. The underlying cloud infrastructure is nor managed nor controlled by the consumer. However, the consumer may have control over the operating system, deployed applications and some of the networking components such as firewalls. Rather than selling raw hardware infrastructure, IaaS providers typically offer virtualized infrastructure as a service. Hardware level resources are abstracted, encapsulated and exposed to upper layer and end users through a standardized interface as a unified resource.

Platform as a Service (PaaS) provides an abstraction layer between the software applications (Saas) and the virtualized infrastructure (IaaS). Consumers may deploy onto the cloud infrastructure applications that were created using programming languages, libraries, services, and tools supported by the provider without having to worry about hardware requirements. Users only have control over the deployed applications and some configuration settings for the application-hosting environment.

Software as a Service (SaaS) allows consumers to use provider applications running on the cloud infrastructure. These applications are accessible from various client devices through web browsers or a program interface. As in the previous case, the consumer does not manage or control the underlying cloud infrastructure be it the software platform which the application is based on (PaaS) or the actual hardware infrastructure (IaaS). However, a SaaS application can be developed on an existing platform and run on infrastructure of a third party.

Regardless of its service class, a cloud can be classified as public, private, community or hybrid. The public cloud may be owned, managed and operated by a business, academic or government organization. It resides on the premises of the cloud provider and is exposed to the general public via the Internet. The main characteristics of a public cloud refer to scalability and flexibility as users can easily add or drop capacity. However, security remains a debatable issue considering the fact that the exact whereabouts and the users who are granted access to data are not known thus making it more prone to hacks.

A private cloud however allows the exclusive use by a single organization comprising multiple consumers. The cloud may be owned, managed and operated by either the organization, a third party or even a combination of the two and may reside on or off premises. The same goes for the community cloud, as its infrastructure is provisioned for exclusive use by a community of customers sharing the same concerns (mission, security requirements, policy, and compliance considerations). This allows a company to select the most suitable cloud model while relying on secure third-party help for its maintenance. As opposed to public clouds, private clouds may provide more control and higher reliability by means of strong service level agreements. Moreover, a private cloud may prove more customizable as storage and network components can be tailored for specific needs. On the other hand, the increased management responsibilities and required expertise will raise the costs. Therefore, it is important to thoroughly compare the advantages and drawbacks of each solution before making a decision.

The infrastructure of a hybrid cloud comprises two or more distinct cloud infrastructures (private, community or public) which are bound together to enable data and application portability. For example, an internally operated private cloud may be bridged with several public clouds by means of standardized or proprietary technology in order to better meet business requirements. This binding however maintains the autonomy of the infrastructures [Pat12]. A hybrid cloud makes the most of both worlds by combining the advantages of private and public clouds. Besides offering scalability and flexibility, a hybrid cloud is also a cost-effective secure solution thus making it increasingly popular among enterprises.

## 1.3 Motivation of the Thesis

The starting point of this thesis was represented by a measurement tool for the Available Transfer Rate and One-Way Delay, previously developed within UC Labs. The tool performed measurements on top of the MAC Sublayer and was therefore restricted to local area networks. Moreover, the estimation of the ATR relied on a passive technique which made it unsuitable for measuring the high data rates found in cloud-based networks. Another drawback referred to the need of synchronization for precise measurements of the OWD, thus requiring expensive equipment. It was therefore decided to build two new measurement tools, relying on active techniques. The novelty consists in the use of Kalman filtering for ATR which enables more flexibility and cyclic-path delays estimation for the OWD thus mitigating the need for precise node synchronization.

These new tools were to be used for making routing decisions in a SDN-controlled cloud-based network orchestrated by OpenStack. Again, the initial testbed was previously developed within UC Labs with the purpose of demonstrating the feasibility of a gearbox-like routing algorithm selection in runtime [Rus11]. The testbed however was limited by a number of factors and did not provide good scalability or flexibility. Consequently, the testbed was virtualized using OpenStack as a hypervisor. The SDN controller was also changed, the now obsolete Beacon being replaced by newer solutions such as Pyretic.

In January 2015, a collaborative research work started, in the form of the CHIST-ERA "DIONASYS" project, a joint initiative of four research institutions (Universities of Neuchâtel, Bordeaux, Lancaster and Technical University of Cluj-Napoca) in four countries, funded by the CHIST-ERA ERA-NET. The goal of DIONASYS is "to make the programming of complex and heterogeneous Systems-of-Systems simpler and more straightforward by allowing a higher level of abstraction and allowing advanced features such as automatic adaptation, automatic interoperation, and support of programmable networks for these tasks". The OWD estimation tool and the OpenStack orchestrated testbed previously discussed were developed as part of the work in this project.

This thesis reflects both personal considerations and others in accordance to the partners within the CHIST-ERA "DIONASYS" project. The results were validated by several papers presented at various conferences: IEEE LANMAN 2016, IEEE ISETC 2012 and 2014, IEEE RoEduNet 2015 and 2016, IEEE COMM 2014 and 2016.

## 1.4 Structure of the Thesis

This thesis is organized as follows:

**Chapter 2 "Active Measurements of the Available Transfer Rate"** presents the implementation of an Available Transfer Rate estimation tool, based on active measurements and Kalman filtering. Experiments performed on both wired and wireless testbeds emphasized the effect of the process noise covariance matrix Q on the estimation accuracy and tracking ability of the tool.

**Chapter 3 "Active Measurements of the One-Way Delay"** is dedicated to an active measurement tool for estimating one-way delays from cyclic-path delay measurements. The novelty of the mechanism is that it does not require synchronization between the nodes of the testbed nor does it rely on ICMP for performing the measurements. Tests carried out in an OpenStack orchestrated testbed showed that the estimation results may have a precision down to a level of nanosecond.

**Chapter 4 "Automatic Control of an OpenFlow-Based Network Using Lua Scripts and SPLAY"** focuses on a mechanism for the automatic control of a virtualized testbed orchestrated by OpenStack. The SPLAY open-source framework, based on Lua, was chosen for this implementation and the main enhancement refers to eliminating the sandboxing concept, integrated in the default SPLAY distribution.

**Chapter 5 "Cloud-Based Networks Composability"** illustrates how multiple sites or hybrid clouds may be composed based on a holonic model. The feasibility of this abstract model was investigated by applying it onto real-time composing OpenStack-based clouds. While an exact mapping of the two frameworks proved unpractical, several guidelines which can be further built upon were provided.

**Chapter 6 "Routing in Software Defined Cloud-Based Networks"** describes the virtualization of the physical testbed used to implement the GRAS routing mechanism, previously developed within UC Labs. The Pyretic SDN Controller was investigated. Also, a method for minimizing the latency in wireless sensor network was introduced, alongside with a mechanism for estimating the power consumption of a Raspberry Pi model 2B device. The feasibility of the global mechanism proposed in this thesis was demonstrated, with emphasis on scalability and composability, key requirements of the CHIST-ERA "DIONASYS" project.

**Chapter 7 "Contributions to Active Measurements for Routing in Cloud-Based Networks"** provides a summary of the five contributions presented in the previous chapters. It also includes some final remarks, awards, and a list of personal publications.

## 1.5 Conclusions

Routing in cloud-based networks displays several characteristics which make legacy routing mechanisms unsuitable. Not only are the nodes dynamic and heterogeneous but the entire network topology may change abruptly. Therefore, a stringent need for interoperability and composability of systems arises. This request can be met by providing an appropriate abstract model which allows the virtualization of the underlying network resources. In turn, this provisions a range of reusable network services which can be composed with existing systems.

The solution proposed in this PhD thesis aims at simplifying the tasks of infrastructure and service providers, allowing them to face the dynamicity of new demands in Future Internet. This is achieved by offering a level of abstraction which makes the processes less vulnerable to the evolution of the underlying cloud infrastructure. Most importantly, the measurement tools and the proposed routing mechanism can be deployed automatically in both traditional and cloud-based networks.

# 2 Contributions to Active Measurements for Routing in Cloud-Based Networks

## 2.1 Contributions Summary

This section presents a synthesis of the contributions which were discussed in the previous chapters.

# 1. Active Measurements of the Available Transfer Rate

The work described in Chapter 2 refers to the implementation of an Available Transfer Rate estimation tool, called ATRAM, based on active probing and Kalman filtering. The efficiency of Kalman filtering relies on its ability to estimate the state of a dynamic system from a set of noisy measurements. The performance of ATRAM was evaluated through several experiments, performed on two different testbeds. For the first testbed the bottleneck link is a wired link while for the second testbed the bottleneck link is a wireless one. The main purpose of these experiments was to observe the impact of the process noise covariance matrix $Q$ on the estimation performance and tracking ability of ATRAM. Several configurations of the matrix have been studied, under various levels of cross-traffic. Results show that the optimal choice is dependent on the characteristics of the bottleneck-link capacity and the cross-traffic. When configuring ATRAM for estimation in wired networks, it is better to choose a small value for $Q_{11}$, since this element is a measure of the expected variations of the bottleneck-link capacity, typically constant in wired networks. Instead, the Available Transfer Rate fluctuates due to cross-traffic variations. Therefore, $Q_{22}$ is the significant element to be tuned. The experimental results indicate that the capacity of wireless links is highly dependent on the radio-channel conditions, which may vary abruptly on a short time scale. $Q_{11}$ becomes the decisive element of the $Q$ matrix.

| | |
|---|---|
| **Contributions can be found in:** | Chapter 2 |
| **Publications:** | [Iva14a], [Iva14b], [Iva14c], [Hos14], [Iva15a] |

# 2. Active Measurements of the One-Way Delay

Chapter 3 presents a solution for the estimation of one-way delays from cyclic-path delay measurements. The cyclic-path delays measurements are performed by a C language software program using a source node that forwards a packet in the network. This packet is then multiplied and passes through all nodes in the network. Finally the packet returns to the source node, where the cyclic-path delays are measured by subtracting the departure time from the arrival time. These measurements are then expressed in terms of one-way delay variables. Since the resulting equation system is underdetermined, an estimation of one-way delay is performed via MATLAB in order to output a final solution of one-way delays on each link of a network. The estimation problem is formulated as a constrained optimization problem with the constraints derived from the cyclic-path delay measurements. Thus, the estimation of the one-way delays can be performed with a precision of nanoseconds. The experiments conducted in an OpenStack-based testbed showed that the estimation is performed more often when there are fewer links thus resulting in a higher accuracy and a better tracking ability.

**Contributions can be found in:** Chapter 3
**Publications:** [Iva15b], [Tau16], [Dio16b]

## 3. Automatic Control of an OpenFlow-Based Network Using Lua Scripts and SPLAY

Chapter 4 introduces the SPLAY open-source framework that facilitates the design, deployment and testing of large-scale distributed applications. The main contribution refers to the implementation of a mechanism which fully automates the control of a virtualized testbed created in OpenStack running Ubuntu 14.04.3. The SPLAY controller was installed on the Infrastructure Provider and the daemons were integrated on the testbed nodes. As a proof of concept, measuring scripts and Open vSwitch scripts (in Lua) were deployed through SPLAY agents, thus avoiding the time consuming and cumbersome processes of manually configuring the testbed. The measuring tools providing necessary information for the network administrator about the QoS parameters of the links (ATR and OWD). This data can further be used to make routing decisions in the SDN Controller, which are sent back to the previously configured Open vSwitches. Compared to legacy solutions such as bash scripting, this approach not only provides more scalability as the code is automatically distributed to a large number of nodes but also a higher degree of flexibility since the code can be tailored for specific nodes.

**Contributions can be found in:** Chapter 4
**Publications:** [Dio15], [Iva16a], [Pad16]

## 4. Cloud-Based Networks Composability

This contribution aims to tackle the composing of complex systems such as merging multiple sites or hybrid clouds based (a request in the CHIST-ERA “DIONASYS” project) on a holonic model. Holons are compositional systems entities, recently proposed as a general framework for the programming and deployment of complex systems of systems. This approach deals with the increasing management and operation complexity by leveraging overlay networks and the higher level of abstraction they allow for distributed operations. In particular, the virtualization of underlying network resources allows for the provision of a range of reusable network services and compose them with existing systems. The main task was to investigate whether this abstract, holonic model can be applied to the complex process of real-time composition of OpenStack-based IaaS clouds. In this context, issues related to the energy-performance tradeoff using techniques similar to those involving dynamic service consolidations were also addressed. The proposed solution aims at simplifying the tasks of infrastructure and service providers, allowing them to face the dynamicity of new demands in Future Internet. This is achieved by offering a level of abstraction which makes the processes less vulnerable to the evolution of the underlying cloud infrastructure. However, this is not a trivial task since a mere mapping of the OpenStack components into the holon vision is not enough. Therefore, a set of preliminary, conceptual recommendations were provided, which can be further built upon for practical use.

**Contributions can be found in:** Chapter 5
**Publications:** [Dio16a], [Iva16b]

# 5. Routing in Software Defined Cloud-Based Networks

This contribution presents an adaptive routing solution in a private cloud-based testbed with Open vSwitches controlled by software. Legacy routing algorithms, such as Dijkstra's, can only detect congestion if it was triggered by either link failure or changes occurring in Data Link layer technology. Consequently, these algorithms are not suitable for routing in cloud-based networks where there is a stringent need for adaptability, interoperability and optimization. A new routing mechanism called GRAS (previously developed within UC Labs) was therefore used. The original physical testbed was controlled by the Beacon Controller and implemented GRAS in order to switch between the following routing algorithms: Modified Dijkstra's (proprietary to UC Labs), Floyd-Warshall and Ford-Fulkerson. This testbed was virtualized in an OpenStack private cloud and the Pyretic SDN Controller was investigated. Experimental results in the virtualized testbed showed that multiple high-level rules may be composed in parallel, thus allowing the implementation of single/multiple routing mechanisms. Moreover, an application-aware routing mechanism which minimizes the end-to-end delay in wireless sensor networks was also introduced. The path computation process is governed by a novel Objective Function based on delay between source and sink nodes (OF-DELAY) in order to minimize the latency, a critical feature for sending alarms in WSNs. In the end, a model for estimating the power consumption of the Raspberry Pi model 2B is provided, based on the CPU utilization and upload and download transfer rates. This entire contribution proved to be a key factor in demonstrating the scalability of the global mechanism proposed by this PhD thesis, in conjunction with the requests of the CHIST-ERA "DIONASYS" project.

**Contributions can be found in:** Chapter 6
**Publications:** [Com15], [Iva15b], [Iva16a], [Lup16], [Luc16a]

## 2.2 Final Remarks

The main goal of this thesis was to design and implement the active measurement tools needed for routing in cloud-based networks. However, during the doctoral stage, the contributions were adjusted according to the requirements of the CHIST-ERA "DIONASYS" project, to which we adhered in 2015. This lead to the birth of the last contribution related to systems of systems composability and interoperability based on the holonic model. Moreover, the idea of SDN-based routing was enhanced from the original goal by leveraging overlay networks and the higher level of abstraction they allow for distributed operations. The achievements of this PhD are already partially deployed within CHIST-ERA "DIONASYS". However, several components are still waiting to be integrated with software modules designed by our partners in the aforementioned project.

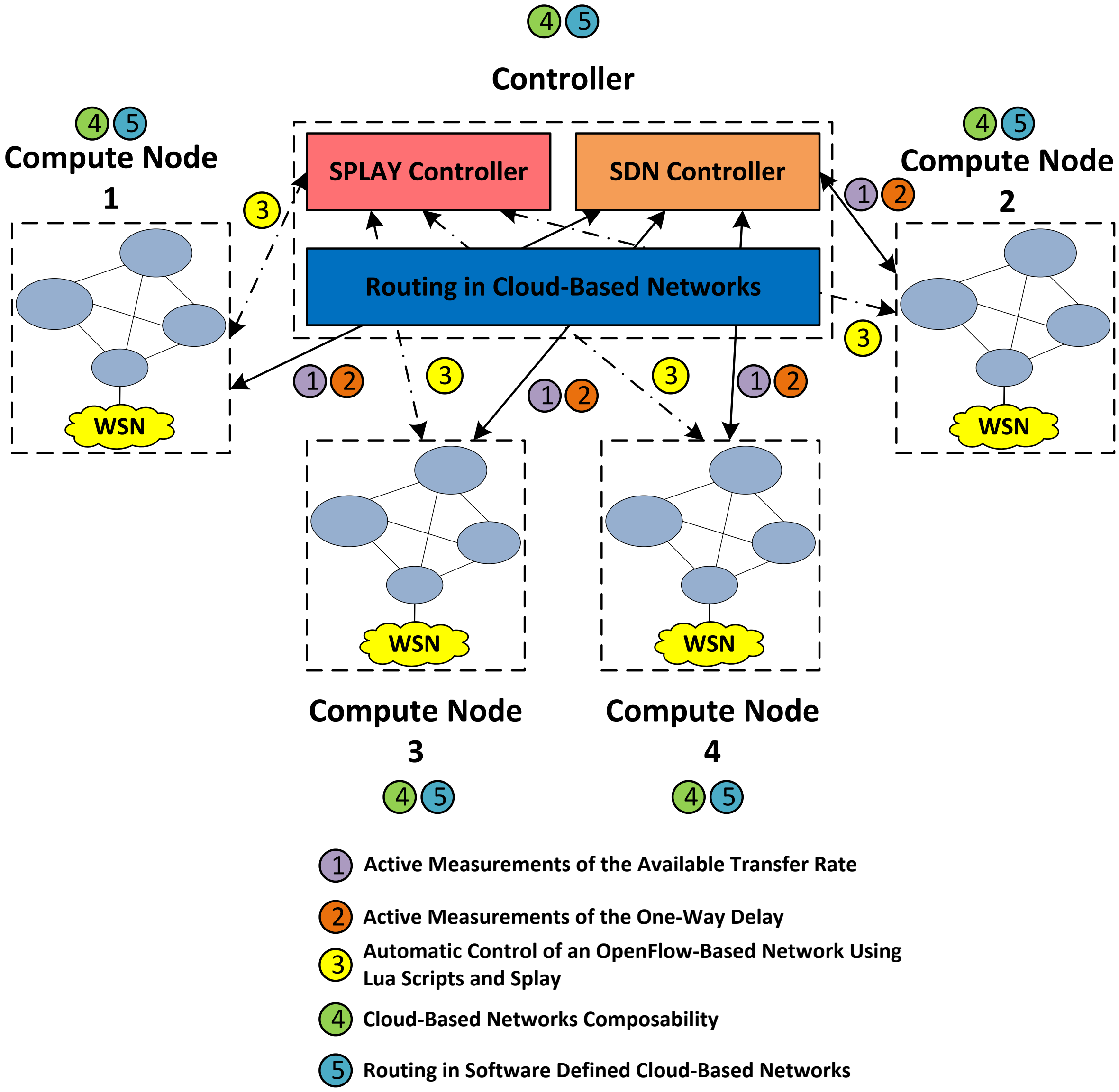


Figure 1. Contributions to the thesis

## 2.3 Awards

Advisor for the papers awarded the First Prize in the "Students Scientific Communication Session in Electronics and Telecommunications", organized by the Faculty of Electronics, Telecommunications and Information Technology, Technical University of Cluj-Napoca, Romania, May 2015 and May 2016.

## 2.4 Personal Publications

### 2.4.1 ISI Journals

| | |
|---|---|
| [Hos14] | A.C.Hosu, Z.I.Kiss, **I.A.Ivanciu**, Zs.A.Polgar, A.Consoli, M.Egido, "Ubiquitous Connectivity Platform for Intelligent Public Transportation Systems", *10th ITS European Congress*, June 16-19, 2014, Helsinki, Finland, WOS:000359811000003, ISSN: 1751-956X |

### 2.4.2 Indexed Database Journals

| | |
|---|---|
| [Iva14c] | **I.A. Ivanciu**, A.B. Rus, and V. Dobrota, "A Tunnel-Based Solution for Seamless Vertical Handover and Load Balancing", *ACTA TECHNICA NAPOCENSIS, Electronics and Telecommunications,* ISSN 1221-6542, Vol.55, No.3, 2014, pp.20-26 |
| [Luc16a] | E. Luchian, **I.A. Ivanciu**, A.B. Rus, G. Lazar, and V. Dobrota, "Migration of an SDN-Based Testbed into a Private Cloud: An OpenStack Approach", *ACTA TECHNICA NAPOCENSIS, Electronics and Telecommunications*, ISSN 1221-6542, Vol.57, No.1, 2016, pp. 1-10 |

### 2.4.3 ISI Proceedings Conferences

| | |
|---|---|
| [Com15] | A. Comsa, **I.A. Ivanciu**, E. Luchian, V. Dobrota, and K. Steenhaut, "End-to-End Delay Minimization in an Application-Aware Routing for Wireless Sensor Networks", *14th RoEduNet Conference: Networking in Education and Research NER'2015*, Agency ARNIEC/RoEduNet and University of Craiova, Craiova, Romania, September 24-26, 2015, pp.80-84, Print ISBN: 978-1-4673-8179-6, DOI: 10.1109/RoEduNet.2015.7311832 |
| [Iva14a] | **I.A. Ivanciu**, A.C. Hosu, Z.A. Polgar, and V. Dobrota, "Capacity and Available Transfer Rate Evaluation for Wireless Links", *10th International Conference on Communications COMM 2014,* Military Technical Academy, "Politehnica" University of Bucharest, "Electronica 2000" Foundation and IEEE Romania Section, Bucharest, Romania, May 29-31, 2014, pp.1-4, DOI:10.1109/ICComm.2014.6866693 |
| [Iva14b] | **I.A. Ivanciu**, A.B. Rus, V. Dobrota, and J. Domingo-Pascual, "Active Measurement of the Available Transfer Rate Used in an Algorithm for Generalized Assignment Problem", *Proceedings of the 11th International Symposium on Electronics and Telecommunications ISETC 2014,* Timisoara, Romania, November 13-14, 2014, Print ISBN: 978-1-4799-7265-4, pp.147-150 |
| [Pad16] | M. Padurariu, B. Rosca, **I.A. Ivanciu**, E. Luchian, A.B. Rus, and V. Dobrota, "Automatic Control of an OpenFlow-Based Network Using Lua Scripts and SPLAY", *11th International Conference on Communications COMM 2016,* Military Technical Academy, "Politehnica" University of Bucharest, "Electronica 2000" Foundation and IEEE Romania Section, Bucharest, Romania, June 9-11, 2016, pp. 299-302, DOI: 10.1109/ICComm.2016.7528286 |

### 2.4.4 Indexed Database Proceedings Conferences

| | |
|---|---|
| [Iva16b] | **I.A. Ivanciu**, E. Luchian, E. Riviere, and V. Dobrota, "OpenStack-based Clouds as Holons: A Functional Perspective*", 22nd IEEE International Symposium on Local and Metropolitan Area Networks LANMAN 2016*, Rome, Italy, June 13-15, 2016 |
| [Luc16b] | E. Luchian, C. Filip, A.B. Rus, **I.A. Ivanciu**, and V. Dobrota, "Automation of the Infrastructure and Services for an OpenStack Deployment using Chef Tool ", *15th RoEduNet Conference: Networking in Education and Research*, University Politehnica Bucharest, September 7-9, 2016, pp. 1-5 |

| | |
|---|---|
| [Lup16] | F.L. Lupaescu, **I.A. Ivanciu**, E. Luchian, and V. Dobrota, "A Firewall Application for Performance Evaluation of the Pyretic Controller in Software-Defined Networks", *15th RoEduNet Conference: Networking in Education and Research*, University Politehnica Bucharest, September 7-9, 2016, pp. 17-21 |


### 2.4.5 Technical Project Reports


| | |
|---|---|
| [Dio15] | A.B. Rus, **I.A. Ivanciu**, and V. Dobrota, "Automatic Control of an OpenFlow-Based Network Using Lua and SPLAY", *„CHIST-ERA DIONASYS "Declarative and Interoperable Overlay Networks, Applications to Systems of Systems"*, Version 1.0, 12 July 2015, 17 pages. |
| [Dio16a] | **I.A. Ivanciu**, E. Luchian, and V. Dobrota, "Implementing Security in the CHIST-ERA "DIONASYS" Testbed", *„CHIST-ERA DIONASYS "Declarative and Interoperable Overlay Networks, Applications to Systems of Systems",* Version 1.0, 8 May 2016, 8 pages. |
| [Dio16b] | **I.A. Ivanciu**, A. Taut, E. Luchian, and V. Dobrota, "Active Measurements of the One-Way Delay in Cloud-Based Networks", *„CHIST-ERA DIONASYS "Declarative and Interoperable Overlay Networks, Applications to Systems of Systems",* Version 1.0, 29 September 2016, 18 pages. |


### 2.4.6 PhD Scientific Research Reports


| | |
|---|---|
| [Iva15a] | **I.A. Ivanciu**, "Active Measurements of the Available Transfer Rate", Ph.D. Scientific Research Report 1 (unpublished), Technical University of Cluj-Napoca, Romania, January 2015 |
| [Iva15b] | **I.A. Ivanciu**, "Active Measurements of the One-Way Delay. Energy Consumption Estimation", Ph.D. Scientific Research Report 2 (unpublished), Technical University of Cluj-Napoca, Romania, September 2015 |
| [Iva16a] | **I.A. Ivanciu**, "Routing in Cloud-Based Networks", Ph.D. Scientific Research Report 3 (unpublished), Technical University of Cluj-Napoca, Romania, February 2016 |


## 2.5 List of Projects

| | |
|---|---|
| 1 | C. Martis (coordinator), V. Dobrota, **I.A. Ivanciu**, E. Luchian (included in list of members), ID P_40_333 " URBIVEL - Advanced Technologies for Intelligent Urban Electric Vehicles", 2016-2017 |
| 2 | V. Dobrota (coordinator for TUCN), A.B. Rus, **I.A. Ivanciu**, G. Lazar, E. Luchian (members) et al., „CHIST-ERA DIONASYS "Declarative and Interoperable Overlay Networks, Applications to Systems of Systems", 2015-2017 |
| 3 | C. Munteanu (coordinator for TUCN), **I.A. Ivanciu** (PhD scholarship) Interuniversity Partnership for Engineering Excellence "PARTING", 2014-2016 |

| | |
|---|---|
| 4 | Z. Polgar (coordinator for TUCN), V. Dobrota, M. Varga, A.B. Rus, G. Lazar, **I.A. Ivanciu**, Z. Kiss, A. Hosu (members), FP7-SME-2012-1/315161 "UCONNECT – Implementation of Ubiquitous Connectivity for Public Transport", 2012-2014 |