MECHANISMS FOR THE DESIGN OF SECURITY AND QUALITY OF SERVICE TRADE-OFF SOLUTIONS

DISEÑO DE MECANISMOS PARA EL DESARROLLO DE SISTEMAS SEGUROS CON CALIDAD DE SERVICIO (QoS)

By
Ana Nieto Jiménez

SUBMITTED IN FULLFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR IN COMPUTER SCIENCE
AT
UNIVERSITY OF MALAGA
CAMPUS DE TEATINOS
BLV. LOUIS PASTEUR, 35. 29071 MALAGA
2015

Advisor:
Fco. Javier López Muñoz
Full Professor, University of Malaga

D. Fco. Javier López Muñoz, Catedrático de Universidad del área de Ingeniería Telemática del Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga,

CERTIFICA QUE:

Dña. Ana Nieto Jiménez, Ingeniero en Informática, ha realizado en el Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga, bajo mi dirección, el trabajo de investigación correspondiente a su Tesis Doctoral, titulada:

Mechanisms for the Design of Security and Quality of Service Trade-off Solutions

Diseño de Mecanismos para el Desarrollo de Sistemas Seguros con Calidad de Servicio (QoS)

Revisado el presente trabajo, estimo que puede ser presentado al tribunal que ha de juzgarlo, y autorizo la presentación de esta Tesis Doctoral en la Universidad de Málaga.

Málaga, a 18 de Mayo de 2015

Fdo: Fco. Javier López Muñoz

Catedrático de Universidad del
área de Ingeniería Telemática

Quid pro quo.
Life is a tradeoff.

Abstract

Security and Quality of Service (QoS) are terms in constant conflict. In this thesis, a detailed analysis of the characteristics and requirements of security and QoS in the candidate networks to be part of the Future Internet (FI), and in the Internet of Things (IoT) is provided, with the aim of evaluating the suitability of the current alternatives for assessing security and QoS tradeoffs. Based on the results of such analysis, it is determined which general models for assessing security and QoS tradeoff in heterogeneous networks should be provided, not only for allowing the analysis at service level, but also allowing the composition of *things*. Moreover, these new mechanisms should allow the integration of new information dynamically, when it is allowable. With this goal, in this thesis, a *Context-based Parametric Relationship Model* (CPRM) is proposed to define the heterogeneous system based on a set of parameters and the relationships between them. This model is implemented in the *Security and QoS Tradeoff Tool* (SQT), which defines a handler for elements in the CPRM. These elements, or contexts, may be integrated and extracted from and to the different CPRM-based systems. SQT uses a base set of parameters that are taken from the initial analysis of security and QoS tradeoff literature. Finally, to enhance the visualisation of the results provided by SQT, initially given as graphs, we define and provide the implementation of a *Recommendation System for SQT* (SQT-RS), which is integrated inside SQT. The analysis of the model and the tool is performed in two use cases within the FI: authentication mechanisms in *Wireless Sensor Networks* (WSN) and recommendations in the composition of mechanisms and tools in 5G Green relay scenarios under eavesdropping and jamming attacks.

Resumen

Seguridad y Calidad de Servicio (QoS) son aspectos ampliamente confrontados. En esta tesis se realiza un análisis detallado de las características y requisitos de seguridad y QoS en las redes candidatas a formar parte de la Internet del Futuro (IF) y de la Internet de los Objetos (IdO), así como de los mecanismos actuales para el análisis de la compensación entre mecanismos de seguridad y QoS. De este estudio se desprende la necesidad de definir nuevos modelos para la evaluación del impacto entre mecanismos de seguridad y QoS, dado que la mayor parte de los estudios centra sus esfuerzos en entornos específicos y características determinadas que no pueden ser fácilmente mapeadas a otros entornos, o cambiar dinámicamente. Por ello definimos un modelo para la composición de esquemas de definición paramétrica basado en el contexto, definido por sus siglas en inglés, Context-based Parametric Relationship Model (CPRM). Este modelo es implementado en una herramienta para la evaluación de mecanismos de Seguridad y QoS (SQT), y su rendimiento evaluado en base a la información integrada en los contextos y la dependencia paramétrica. Finalmente, para mejorar la visualización de los resultados y agilizar la comprensión del modelo definimos un sistema de recomendaciones para la herramienta SQT (SQT-RS). El análisis del modelo y de la herramienta se realiza empleando dos casos base dentro de escenarios del FI: mecanismos de autenticación en redes de sensores (WSN) y recomendaciones para la composición de mecanismos en escenarios de 5G Green sometidos a eavesdropping y jamming.

Acknowledgements

Life is full of decisions which lead to different paths. In my case, it was my advisor, Fco. Javier Lopez Muñoz, who encouraged me to make one of the most influential decisions in my life to date. To him I owe the realisation of this thesis, and the very positive impact on my work and personal life. From a professional point of view, I greatly appreciate the confidence placed in me, giving me the freedom to grow as a researcher. For myself, and I know that for many, Javier is a great example to follow; his profesional achievements are unquestionable, but it is also impossible not to feel a deep respect for all his effort and dedication to our group. Moreover, thanks to Jose Maria Troya and his confidence in our group, NICS Lab. Having that confidence is one more cog in the wheel that moves us to continue.

Thanks also to Charalabos Skianis and his research group for their warm welcome during my predoctoral stay on the island of Samos (Greece). In particular, much gratitude to Nikos, Dimitris and Dimitris's family. It was certainly a very profitable stay for the thesis and I personally have very good memories of those months.

I would also thank to Lisa Huckfield for her great work on the revision of this thesis, usually against the clock. Any possible grammatical errors are my own in making last minute changes. Rodrigo, thank you very much for your help over the last few months. To me it means a lot to receive advice from someone like you, with *stripes* but with his feet on the ground and who works so passionately.

To my colleages in the NICS Lab over the years (by seniority): Monte, Onieva, Isaac, Gerardo, Miguel, Rodrigo (again), Carmen, Pablo, Cristina, Pepe, Ángel, Andrés, Dani, Fran, Jesus (and Maria Jose, by adoption), Noelia, Ruben, Edu, Jose, Saul, David, Lorena. Thanks to all of you for creating such a good environment for learning. It means a lot to me to be in a group where there are always new hot-topics to learn about. Specially thanks to all of you that have contributed to making the SPRINT project possible, which is the reason why I had the opportunity to realise this thesis.

Without diminishing the appreciation that I have for all of you, my *fellows in the trench*, Fran and David deserve special mention for being so close to me over the years. Thank you both because without your company and support this process of becoming a doctor wouldn't have been the same. Also, thanks to Noelia for being there for me, to Ruben for his valuable comments about my work during the travels, and to Jesus for his skill in managing any problem.

To write a thesis involves much personal time, so I would also like to thank to friends for their patience for my missing numerous social gatherings because "a girl has to publish". Thank you to my oldest friends Ana Mari, Elena and Carolina for being there for more than twenty years. My polemic Carolina, even when you're away you're incredibly close, thanks for being you.

Thanks to my aunt Toñi for being an example of strength, and for the merchandising "Nieto" (single branch), that accompanies me on all my travels. Also thanks to my uncle Biel - as he wants to be called now - for always teaching me something new, he is my favorite uncle, and I'm sure he'll be an enviable godfather. Thanks to both of you for our virtual meetings during my research stay in Greece, I really appreciate you.

To my dear and eagerly awaited Julia, thank you very much for being with me (literally) these last few months. You light up my life and give me the inner peace I was searching for. The outer peace is already taken care of by your daddy, whom I love more with each passing day. I'm sure that you're gonna love him too. There is not better father because there is no better partner in life than him. Gerardo thanks for your patience and your support all these years, and in the years ahead.

Last but not least, thanks to all researchers that consider this work useful and want to use it for their own research. There is no greater achievement for a researcher that to know their contribution will serve to advance further in the field to which they have devoted so much time.

Agradecimientos

La vida está repleta de decisiones que llevan a diferentes caminos. En mi caso, fue mi director de tesis, Fco. Javier López Muñoz, quien motivó una de las decisiones que más ha influido en mi vida hasta la fecha. A él he de agradecerle la realización de esta tesis, y el impacto tan positivo que ha tenido en mi vida laboral y personal. Centrados en el ámbito profesional, le agradezco enormemente la confianza depositada en mi y haberme dado la libertad necesaria para crecer como investigadora. Para mi, y me consta que para muchos, él es todo un ejemplo; sus logros a nivel profesional son incuestionables, pero, además, es imposible no sentir un profundo respeto y afecto por todo su esfuerzo y dedicación a nuestro grupo. También quisiera agradecer a Jose María Troya su confianza en NICS Lab. Contar con esa confianza es una pieza más en el engranaje que nos impulsa a continuar.

Gracias a Charalabos Skianis y a su grupo de investigación por su buena acogida durante mi estancia predoctoral en la Isla de Samos (Grecia). En especial, gracias a Nikos, a Dimitris y a su familia. Sin duda fue una estancia muy provechosa para la tesis y personalmente me llevo muy buen recuerdo de esos meses.

También quisiera agradecer a Lisa Huckfield su estupenda labor de revisión, en varias ocasiones contra reloj. Cualquier posible error gramatical se debe a mi intervención al realizar cambios de última hora. De igual forma quiero agradecer a Rodrigo su inestimable ayuda estos últimos meses. Para mí significa mucho recibir consejo de alguien como él, con galones pero con los pies en la tierra, y que trabaja de forma tan apasionada.

A mis compañeros de NICS Lab durante todos estos años (por antigüedad): Monte, Onieva, Isaac, Gerardo, Miguel, Rodrigo (otra vez), Carmen, Pablo, Cristina, Pepe, Ángel, Andrés, Dani, Fran, Jesús (y María José, por adopción), Noelia, Ruben, Edu, Jose, Saul, David, Lorena. Gracias a todos por crear un clima tan bueno para el aprendizaje. Aporta mucho a nivel profesional estar en un grupo en el que siempre hay algo nuevo y puntero que aprender. En especial, gracias a todos aquellos que hicísteis el proyecto SPRINT posible, y con ello la beca que me dio la oportunidad de realizar esta tesis.

Sin menoscabo del aprecio que tengo por todos vosotros, mis compañeros de trinchera, Fran y David se merecen una mención especial por ser tan cercanos a mi estos últimos años. Ambos son estupendos y, sin su compañía y apoyo, nada sería igual. También quisiera agradecer a Noelia el podar contar siempre con ella, a Rubén por sus comentarios sobre mi trabajo en los viajes, y a Jesús por su destreza al gestionar cualquier tipo de problema.

La realización de una tesis implica mucho tiempo personal, por lo que también agradezco a mis amigos por su aguante a los innumerables escaqueos de reuniones sociales porque "una chica tiene que publicar". A mis amigas de toda la vida más cercanas, Ana Mari, Elena y Carolina, gracias por seguir ahí tras más de veinte años de amistad. Mi polémica Carolina, aún cuando estás lejos estás increíblemente cerca, gracias por ser tú.

Gracias a mi tita Toñi por ser un ejemplo de fuerza, y por el *merchandising* "Nieto" - rama única - que me acompaña en todos mis viajes. También gracias a mi tito Biel – como le gusta que le llamemos ahora – por enseñarme siempre algo nuevo, inevitablemente es mi tito favorito, y será un padrino envidiable. Gracias a ambos por las charlas virtuales durante mi estancia en Grecia, siempre me aportáis mucho, os adoro con locura.

A mi ya esperada Julia, gracias por estar conmigo, literalmente, estos últimos meses. Sin duda me has dado la paz interior que necesitaba. De la paz exterior ya se ha ocupado tu papi, al que quiero más cada día que pasa, y al que tú vas a adorar. No puede haber un padre mejor, porque no hay mejor compañero en la vida que él. Gracias Gerardo por tu paciencia y apoyo todos estos años, y los venideros.

Por último, pero no menos importante, gracias a todos aquellos investigadores que consideren útil este trabajo y que quieran usarlo para sus propias investigaciones. No hay mayor logro para un investigador que el que su aportación sirva para avanzar más en el ámbito al que ha dedicado tanto tiempo.

Table of Contents

List of Figures

List of Tables

Acronyms

Acronym	Term
AAA	Authentication, Authorization, Accounting.
AC	Asymmetric Cryptography.
AF	Amplify and Forward.
AH	Authentication Header.
AP	Access Point.
AR	Action Rule.
BD	Brute Dependence.
BDFR	Battery-Dependent Fixed Relay.
BDMR	Battery-Dependent Mobile Relay.
BER	Bit Error Rate.
BFS	Basic Formulation Set.
BTS	Base Transceiver Station.
CAS	Certificate-based Authentication Scheme.
CBC	Cipher Block Chaining.
CBC-MAC	CBC Message Authentication Code.
CCM	Counter with CBC-MAC.
CDP	Call Dropping Probability.
CEA	Channel Estimation Accuracy.
CFS	Complex Formulation Set.
CML	ChaMaLeon.
CPRM	Context-based PRM.
$CPRM_i$	Instance of CPRM/CPRMi.
CPS	Cyber Physical System.
CSI	Channel State Infomation.
DAS	Direct Storage based Authentication Scheme.
DB	Dependencies Brutes.
DbO	Domain-based Optimization.
DiffServ	Differentiated Services.
DoS	Denial of Service.
DRQoS	DoS-resistant QoS protocol.
DS	Data Services.

Continued on next page

Continued from previous page

Acronym	Term
DSCP	DiffServ Code Point.
e-MANET	Emergency MANET.
ECC	Elliptic Curve Cryptography.
ECDSA	Elliptic Curve Digital Signature Scheme.
ECMQV	Elliptic Curve version of Menezes-Qu-Vanstone.
ECN	Explicit Congestion Notification.
ESP	Encapsulating Security Payload.
FD	Full Duplex.
FI	Future Internet.
GBR	Guaranteed Bit Rate.
GC	General Context.
GCD	Generic Communication Diagram.
GUI	Graphical User Interface.
HD	Half Duplex.
HMIP	Hierarchical MIP.
HLR	High-Level Requirements.
HW	Hardware.
ICV	Integrity Check Value.
IDS	Intrusion Detection System.
IdO	Internet de los Objetos.
IMEI	International Mobile Equipment Identity.
IMSI	International Mobile Subscriber Identity.
IN	Individual priority to the Node / Intermediary Node.
IntServ	Integrated Services.
IoT	Internet of Things.
IP	Internet Protocol.
IPsec	Internet Security Protocol.
J2ME	Java 2 Micro Edition.
LoWPAN	Low power Wireless Personal Area Network.
LTE	Long Term Evolution.
MAC	Medium Access Control / Message Authentication Code.
MANET	Mobile Ad-Hoc NETworks.
MD	Matrix of Dependencies.
MIP	Mobile IP.
MIH	Media Independent Handover.
MOS	Mean Opinion Score.
MS	Membership.
M2M	Machine-to-Machine.
NFC	Near Field Communication.
NFV	Network Functions Virtualization.
NGBR	Non-GBR.
NL	Number of Layers.

Continued on next page

Continued from previous page

Acronym	Term
NND	Native Network Diagram.
NO	Number of Operations.
NP	Number of Parameters.
NProp	Number of Properties.
NT	Number of Types.
NTD	Nothing To Do.
PC	Particular Context.
PM	Parametric Map.
PRM	Parametric Relationship Model.
PDR	Packet Delivery Rate.
PKC	Public Key Criptography.
Pr	Private.
PSFR	Power Supplied Fixed Relay.
PSR	Packet Sent Ratio.
PT	Parametric Table.
Pu	Public.
QCI	QoS Class Indicator.
QoS	Quality of Service.
QR	Quick Response.
R	Rule.
RR	Requirement Required.
RTT	Round Trip Delay Time.
SA	Security Associations.
SC-ECMQV	Self-Certificated ECMQV.
SDN	Software Defined Networks.
SeaSoS	Seamless Mobility with Security and QoS Support in 4G Networks.
SeQoMo	Secure, QoS-enabled mobility.
SIM	Subscriber Identity Module.
SINR	Signal to Interference plus Noise Ratio.
SIR	Signal to Interference Ratio.
SLA	Service Level Agreement.
SNR	Signal to Noise Ratio.
SOK	Sakai-Ohgishi-Kasahara.
SOP	Secrecy Outage Probability.
SP	Service Provider.
SQT	Security and QoS Tradeoffs.
SQT-RS	Recommendation System for SQT.
SW	Software.
TCP	Transmission Control Protocol.
TPM	Trusted Platform Module.
ToS	Theft-of-Service.
UE	User Equipment.

Continued on next page

Continued from previous page

Acronym	Term
UMTS	Universal Mobile Telecommunications System.
VPN	Virtual Private Networks.
WISA	Wireless Interface to Sensor Actuators.
WiMAX	Worldwide Interoperability for Microwave Access.
WLAN	Wireless Local Area Network.
WMAN	Wireless Metropolitan Area Network.
WMSN	Wireless Multimedia Sensor Network.
WPAN	Wireless Personal Area Network.
WSN	Wireless Sensor Networks.
W-SKE	Wireless Shared Key Exchange.
3G	Third Generation.
4G	Fourth Generation.
5G	Fifth Generation.
6LoWPAN	IPv6 over LoWPAN.

CHAPTER 1

Introduction

In this chapter, we provide an overview of the Future Internet (FI) and related network technologies that are candidate to be part of it. In particular, the integration among Wireless Sensor Networks (WSN), Mobile Ad-Hoc Networks (MANETs) and cellular networks is analysed in order to identify the main challenges to be addressed in future developments within the scope of Security and QoS tradeoff. We explain why these will be relevant networks in future deployments, and the motivation and justification behind the work presented in this thesis. Finally, the goals of the PhD work, direct contribution and the structure of the thesis are detailed at the end of this chapter.

Moreover, security services and mechanisms complicate the work of the QoS mechanisms, because, in general, security solutions consume network and local resources to analyse the network and devices to identify a wide range of threats [14, 15]. Therefore, in addition to the complexity of the QoS solutions themselves, the system administrators have to deal with the deployment of security solutions to protect the networks and their users' rights. In dynamic and heterogeneous environments – more so when the environment is open to the Internet – predicting when the security solutions will be required is very difficult, because is not always possible to know the exact moment when security services will request additional resources [12]. For example, when a new threat is detected by an intrusion detection system, various reactive measures may be needed to mitigate or prevent the security risk. Besides, security mechanisms are better when more is known about the network and the devices and this can be very intrusive from a QoS performance point of view. Certain decisions may be very inconvenient to a productive system, for example, isolating part of a network affects the availability of resources and indeed some attacks can force this behaviour to produce denial of service. The cooperation of security and QoS mechanisms can help to avoid some of these problems however is not always possible (or desirable).

In a traditional sense, the configuration of the services in a system depend on the resources available, so in many cases either security or QoS mechanisms were ignored because of their incompatibilities and to reduce the complexity. Part of the problem is that security and QoS mechanisms are usually designed without considering the requirements of each other, and their interdependencies [14, 16]. Traditional approaches focus on specific purposes, and having to choose between security and QoS solutions can be understandable – to a certain degree - in closed and isolated environments. However, the convergence of heterogeneous networks is a reality that cannot be ignored. Networks in the Future Internet paradigm are fully cooperative, open to a rich number of user and system profiles, and very dynamic. Hence, both type of mechanisms, security and QoS have to coexist, because both are required in the different networks in different ways.

Finally, the interpretation of security and QoS can be different depending on the environment considered, and therefore the context. For example, in energy-constrained environments traditional QoS mechanisms cannot be directly applied, and the QoS requirements are focused on saving as much as energy as possible [17, 18]. This also complicates the deployment of security mechanisms. When resource-constrained networks want to be part of the Future Internet perspective, these restrictions must be considered, adjusting or developing new mechanisms able for protecting these networks – always ensuring that they do not become a back door to the rest of the networks – while ensuring the QoS requirements of the devices of the same.

1.1.1. The Balance of Security and QoS

In general terms, security and QoS can be placed on a balance, as is Figure 1.1 shows. If we do not give enought importance (or relevance) to either of them, then the balance could crush user and system requirements which are very important for the survival of any system. Besides, note that, the security and QoS tradeoff, can be seen as a composition of features to balance the use of resources.

As different environments understand the security and QoS parameters in a particular

way, the relevance of a component can be very subjective, because three main reasons: (i) the cost of implementing a mechanism, service or requirement in a system may be higher than in another system, (ii) based on the context, perhaps the system does not require a specific requirement to be implemented, and (iii) the relevance of a component can change over the lifetime (e.g., authorization in a particular application once the permissions have been granted).

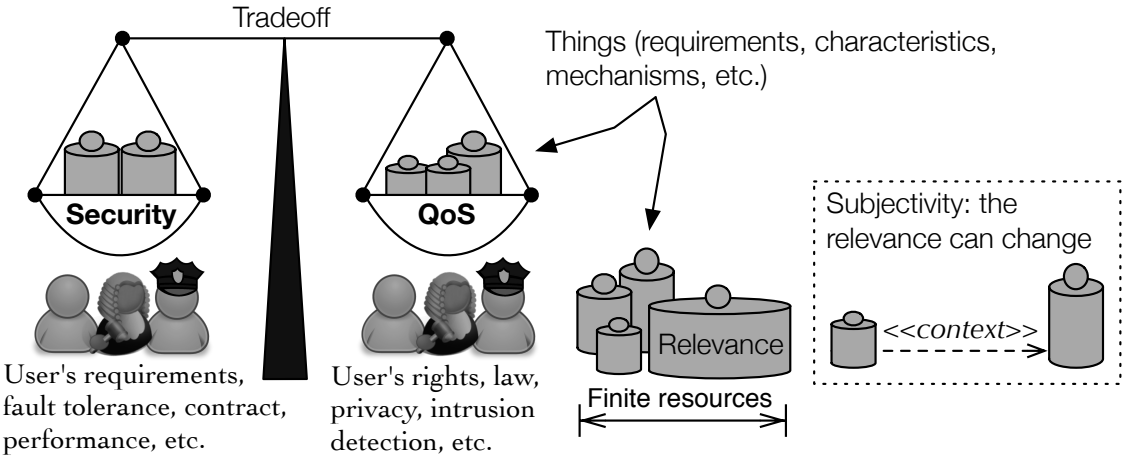


Figure 1.1: Security and QoS Tradeoff.

The problem of achieving a balance between security and QoS has been identified by the research community [14, 16, 19, 20, 21, 22]. At present, by analysing the current contributions it is possible to identify a set of trends in the analysis of security and QoS tradeoffs. Most of these contributions focus on service composition/selection, as in [23], where model checking techniques are used to verify the composition of security services, or in [24], where a tool for evaluating the composition of security services based on *Multi-Objective Optimisation* (MOS) is provided. In [16] trust and QoS tradeoffs are analysed for web services composition. In this case, the selection of services concentrates on general services (not necessarily security services), and trust is a property of the services that must be considered during the agregation of services. In [25] the composition of security services is proposed for *Software Defined Networks* (SDN), where high-level approaches are feasible. SDN separates the data plane from the control plane, simplifying the orchestration of resources, and reducing the deployment costs; causes that make it a trendic topic. Security is seen in some approaches like [26], as a requirement to protect the network against *Denial of Service* attacks for ensuring the QoS. In [27] an approach to provide security in SDN considering QoS guarantees is presented. However, high level analysis is required to identify tradeoffs between the components based on their characteristics. Moreover, the security and QoS tradeoffs are also a problem in resourced-constrained environments that are connected to the Internet [16, 28]. In these environments defining the parameters, operations and the rest of the components and properties within a context, is key in identifying their relevance in the final composition of elements in the environment [29]. Alternatively, in [30] a model based on three static contexts (computing, physical and user) is defined based on a utility function in order to take into consideration the user's preferences to capture fine-grane tradeoffs between security and QoS.

In this Chapter and in Chapter 2, additional related work is analysed, and the result is a wide ranging set of solutions that cannot be directly related with each other at the same layer. We conclude that most of the current approaches (i) focus on specific objectives (security and QoS tradeoffs using specific parameters or at specific layers, typically at the service layer), (ii) define generic models but do not consider partial-knowleadge of the environment (however, it is not always possible to predict the final mechanisms that will implement the properties), or (iii) do not consider the subjective perception of the user (what the user wants is not always best, but it is what they want). Moreover, (iv) the collaboration between different networks is not properly addressed (however, mobile nodes can traverse different networks and therefore different requirements must to be considered based on the context).

Besides, in future environments, it is envisioned that the user will be included as part of the network itself and will be able to interact with the things or objects around him. As future networks can be used for a wide range of purposes, subjective values have to be considered. In fact, subjectivity is essential to identify the components or things that are key to a user or any other actuator in the network. Therefore, as users' nature is dynamic and heterogeneous, and takes different objectives into account over time, subjective values have to be included to allow the dynamic changes, according to users' perception.

What characteristics and fetures are considered static or dynamic depends on the context. Therefore, a high-level analysis of the security and QoS tradeoff is only possible if we focus on a set of representative environments. This way, we can be able to identify the main topics and challenges to be addressed. We consider that the diversity of scenarios in future networks can be addressed selecting as representative environments those with the following characteristics: (i) ad-hoc environments *where the user is present*, (ii) resource-constrained environments *with elements/sensors capable for collecting information*, (iii) environments whith a *communication infrastructure with more resources*, and (iv) the chosen environments have proved *their interoperability with each other*, or are taking several steps towards the networks convergence.

These requirements are needed to provide full coverage to the current and new scenarios considering: user's requirements, the different capabilities of the networks, sensing of events, dynamic, distributed and collaborative approaches and allways-on trends between others.

Future networks must to be analysed to identify the characteristics, main parameters and relationships among them, then we will be able to provide different security and QoS mechanisms as alternatives to be deployed in productive environments, and at different layers. However, the analysis cannot be static or isolated, and in our own solutions we should open the door for the integration of rich information from external sources.

1.2. Networks Convergence

An important part of the research in information technology focuses on convergence and network integration, with the aim of benefitting from features provided by the different types of networks. As a consequence, there is a growing convergence in order to achieve the *all-IP* and *always-on* paradigms. While the first one provides the common infrastructure for network communication, the second one focuses on the need for permanent connection to the Internet. Moreover, the definition of new concepts such as *Future Internet* (FI) or the

Internet of Things (IoT) encourages such steps towards the convergence of networks.

The concept of FI is concerned with the future interconnection of heterogeneous networks. For instance, within FI, the IoT considers that the interaction with any object of the real world is an essential requirement where the user will be necessarily and inevitably involved. The ideal scenario is one interconnected world where *things* can connect to each other and users are able to interact with those *things* using the technology deployed for this purpose. Indeed, one of the main challenges is how to deploy the interoperability mechanisms in that scenario without compromising the security and the *Quality of Service* (QoS) in resource-constrained technologies. In that sense, *Wireless Sensor Networks* (WSN), *Mobile ad hoc networks* (MANETs), and cellular networks are expected to become key networks within the IoT and the FI due to the advantages that they provide to users. Previous studies such as [31] endorse these networks as particularly significant from a security point of view.

Intuitively, the inclusion of *things* in FI is only possible if these are able to provide some kind of information. The simplest way is the use of *Radio-Frequency IDentification* (RFID) [32]. RFID tags contain information about the item in which they are placed, but they are limited by their resources and computational capabilities. These limitations are solved in part by the use of sensors, which are devices with computational capabilities. Although the size of the sensors may vary according to their purpose, those which are dependent on small batteries are very common. Indeed, the main advantage of these ones is, in many cases, their self-organised capabilities, that enable these devices to collaborate, creating WSNs. However, as sensors are typically resource-constrained devices and, moreover, depend on their batteries, the protocols that these devices use to collaborate are specifically designed to optimise the power consumption, extending as much as possible the lifetime of the sensor, and, therefore, the lifetime of the WSN. This feature allows interoperability with less powerful *things* (e.g. RFID) and also with less restricted devices (e.g. laptops).

Despite the fact that some sensors enable firmware over the air updates, making software changes more flexible, this is not always possible. Moreover, modifications in industrial sensors can take years to be approved. For example, sensors used in planes or nuclear plants are under strict technical control. Testing these systems can take from several months to years, depending on the components and their functionality. Therefore, WSNs are limited by their main purposes and resources.

Another type of network that is taking a major role precisely due to users' demands is cellular networks, that become, in many cases, intermediary networks between various technologies. Cellular networks use resource-constrained devices (smartphones) and an infrastructure composed of powerful devices (e.g. long-range base transceiver stations).

User dependence on mobile phones and smartphones has greatly increased, and are getting closer to offering the same functionality required by a Web/application user, hence evolving from specific-purpose platforms to general-purpose platforms. The way in which users interact with each other (social networks) is transforming personal devices into MANETs in various scenarios, where ad hoc communication is required. Nonetheless, behind the mobile phones, the infrastructure provided by Service Providers is expected to be less resource-constrained than an ad-hoc infrastructure formed by battery dependent mobile nodes. However, security mechanisms are currently applied within a closed and private environment.

Consequently, WSNs, MANETs and cellular networks are closely interrelated. Although WSNs and MANETs can be part of the IoT, cellular networks are more related to the FI and

the role of the user in it. In any case, we understand that the relationship between these three types of networks is very interesting from the point of view of convergence within the IoT and FI. For example, sensors can help with early-detection of changes in the environmental conditions where they are deployed (e.g. physical monitoring). However, energy restrictions means they are less able to transport this data directly through a powerful network without the use of an intermediary. The optimal situation would be for the sensor to be able to communicate data to a device within a MANET, for example, so that it could delegate the transmission of urgent data to a powerful device without draining its own battery. This is not always possible, and is highly dependent on the scenario.

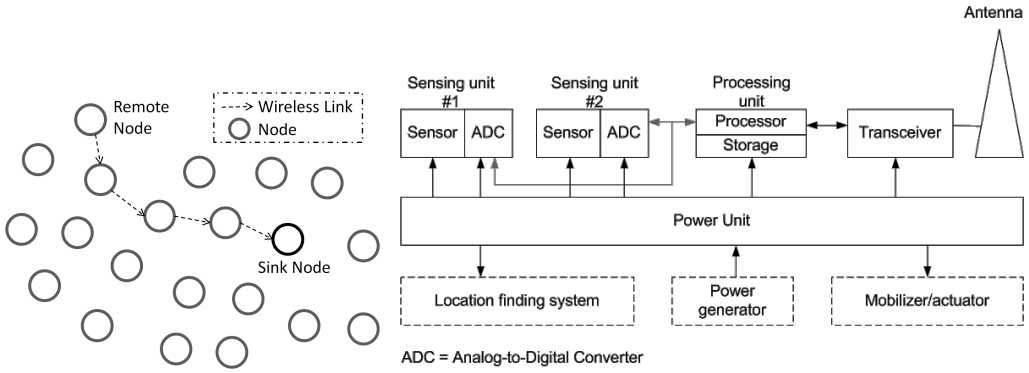
Finally, the networks selected meet the above requirements (i-iv): (i) Ad-hoc environments where the user is present: MANETs, and WSNs in some scenarios. (ii) Resource-constrained environments with elements/sensors capable for collecting information: WSNs. (iii) Environments whith a communication infrastructure with more resources: cellular networks. (iv) The environments chosen have proved their interoperability each other or steps towards their interoperability, that is, steps towards the networks convergence: MANET, WSNs and cellular networks.

In the following paragraphs, an overview of the resource-constrained networks addressed here is presented. Then, the particular case of cellular networks is discussed. Finally, given the dynamic nature of MANETs and cellular networks, the mobility management technologies over *Internet Protocol* (IP), *Mobile IP* version 6 (MIPv6), and media independent handover (MIH) are analysed. The first allows roaming between different networks (e.g. *Wireless Local Area Network* (WLAN), *Worldwide Interoperability for Microwave Access* (WiMAX), and *Universal Mobile Telecommunications System* (UMTS)) without loss of connection [33], while the second allows the handover between different network technologies at low level. Thus, from the point of view of network technologies, the goal for this structure is to cover those networks that will play an important role in the Future Internet.

1.2.1. Wireless Sensor Networks

Wireless Sensor Networks (Figure 1.2) are composed of sensors, resource-constrained autonomous devices with limited functional capabilities. WSNs are designed to solve specific problems, and are normally used to monitor physical or environmental conditions within an area (e.g. temperature, humidity, radiation, light, acceleration/seismic and magnetic measures among others). Typically, a WSN is composed of a large numbers of sensor nodes, where each node has the capability of collecting and analysing data before routing it to a principal node called Sink, that collects all the data from the network [34]. Although it is possible to send the data directly to any of the sensors within the range of the source node, the communication between nodes takes place hop-by-hop in order to save energy [35].

The typical location for the sink node is the centre of the network, because the greater the distance between the source and the Sink is, the higher is the number of intermediate sensors used to route the information, and more energy is consumed in the process. We have to note that if all the data is routed to the sink node, then those nodes nearest to the sink experience a high amount of traffic in contrast with the nodes that are farther from it [36]. For this reason, these nodes could use up their energy long before the rest of the nodes in the network. Moreover, it is essential that the sink node keeps its connectivity with the



(a) Wireless Sensor Network (b) Architecture of a Sensor [34]

Figure 1.2: Wireless sensor network.

whole network at all times because, otherwise, the loss of data could render the network useless. Some WSNs define the set of regions of the network that are essential for a good performance, known as areas of interest, or critical regions. In these cases, the sink has to maintain the connectivity as long as possible with those areas in order to ensure the correct operation of the network.

1.2.1.1. Challenges in FI

1.2.1.1.1. Deploying Security Mechanisms. Studying the impact that security mechanisms have on QoS in the scope of WSNs becomes a challenging task [37]. Clearly, deploying security features in sensors that are directly connected to Internet can be a daunting task [15], and traditional security mechanisms are not always suitable for use in WSNs [38, 39]. Internet opens the door to a large number of possible threats, and because sensors are resource-constrained devices, they are unable to implement complex security mechanisms. This could severely limit the lifetime of sensors and other devices with similar characteristics, and inevitably affect QoS when security mechanisms are used [40]. For example, location privacy in sensor networks may require packet injection, increasing transmission and therefore increasing energy consumption [38].

In this sense, routing tasks is one of the key issues because, in most cases, they consume more energy than the rest of operations in the sensor or any other wireless device [41]. When security mechanisms are used, the cost usually lies in the communications associated to the implementation of the steps of secure protocols. Besides that, adding cryptographic functionality to sensors or other small devices increases complexity and requires more memory and processing time. This is also a problem for some security mechanisms based on distributed information systems. For instance, establishing a reliable trust system requires the exchange of data between various nodes of the network, what severely affects energy consumption [42] and increases overhead in the network. Therefore, the problem of applying security mechanisms is not only resource consumption but also the uncertainty of the security solutions that need to be integrated into the sensors. For example, this is a problem in some industrial sensors, with strong restrictions and real-time procedures.

Finally, in many cases, routing protocols must consider not only the nodes closest to the destination or the safest nodes, but also their energy levels [41, 43], and multiple high-level restrictions [43]. The latter is very difficult when sensors are optimised for a unique purpose because the possibility to modify the firmware is minimal.

1.2.1.1.2. Security as a Key Factor for Performance. Paradoxically, the lack of security mechanisms can have negative consequences for QoS in WSNs. Thus, in [18], the effect of not providing confidentiality, integrity, authenticity and availability services in a sensor network is analyzed. The study encompasses various WSN technologies, namely *Wireless Interface to Sensor Actuators* (WISA), WirelessHart, ISA 100.11a, ZigBee and 802.15.4 *Medium Access Control* (MAC). The results show that, for instance, the lack of integrity in communication increases the packet loss and decreases the throughput.

Moreover, without authentication mechanisms, a malicious node can impersonate other nodes in the network and affect availability. In addition, that study shows that the standards are still vulnerable to jamming, collision and flooding attacks, which affect QoS of the system. Moreover, it shows that QoS and security support for heterogeneous network segments remain unexplored fields.

While the approach in [18] shows that security can prevent QoS degradation (Security for QoS), [44] states that QoS is a requirement for security in a sensor network (QoS for Security). In that approach the security levels are classified based on the confidentiality of information, data integrity and availability of resources. The QoS is discussed in terms of availability, reliability and serviceability, also taking the energy performance into account. Indeed availability can be considered as a security requirement [45], and is a key factor for good intrusion detection. For example, availability of the devices within an *Intrusion Detection System* (IDS) or the databases that store the evidence of attacks are critical factors that have to be considered.

1.2.1.1.3. Data redundancy and Hierarchy. Another aspect to consider is data redundancy. In WSNs, several sensors can cover the same area, and therefore they can measure the same event. This redundancy allows the sink node to assess whether the event is valid or an anomaly. For example, in a forest in which sensors are deployed for fire detection, if all the sensors in an area except one detect the presence of a fire, it probably means that the sensor that did not detect the event is not working properly. Likewise, if only one sensor warns about a fire, most probably the fire does not exist. The same may be applied to intrusion detection.

The relationship between data redundancy, reliability, energy consumption, data fusion and network delays is studied in [17]. In fact, the more data redundancy, the more reliable the information is, although the sensors use more energy in delivering data.

In order to alleviate energy consumption due to data redundancy, data aggregation is performed. For example, if there is a hierarchical structure, the cluster head could decide whether there is indeed a fire and lead the response to the sink node or the next cluster head in the hierarchy. However, this process of aggregation may cause delays in the network due to the decision process, and the cluster heads must devote part of their resources to that end.

Regarding security, aggregation process is very appealing to an attacker because it does not have to misrepresent or impersonate any nodes but just to discover the cluster heads and replace them. Therefore, the clusters not only become potential bottlenecks, but they become key points for distortion of the measurements of a WSN environment.

Nevertheless, hierarchical structures improve the local management of segments in the network because clusters have faster access to the information collected than the rest of devices. Hence, given a context, it is possible to optimise the decision process. This is one of the reasons why centralised architectures are still considered useful despite advances in distributed systems.

1.2.1.1.4. Deploying QoS Mechanisms. We cannot forget that guarantying QoS without considering the security requirements would not be trivial in these systems because, in order to offer QoS guarantees, we need a certain degree of predictability, difficult to provide for the vast majority of resource-constrained networks or dynamic networks due to, for example, changes in network topology [46, 47].

Predictability is related to resource reservation, which is a common technique in QoS mechanisms. The protocols for resource reservation guarantee that a path is available for transmission within a period of time. In order to do this, these protocols require the sending of requests to reserve resources through various paths in the network, thereby consuming the resources available for data transmission. Additionally, the use of such mechanisms leaves the network exposed to QoS signaling attacks, where an attacker reserves unused resources. The result is that, on the one hand, the legitimate nodes can not reserve resources for their own use, and on the other hand, the intermediary nodes waste their energy in the QoS signaling process.

If the network topology is known, the attack could be targeted at specific nodes (e.g. cluster nodes) to damage the network connectivity. In the case of the WSN, it is also necessary to take into account the environmental conditions that can affect some devices in the network. For example, a storm could wipe out several sensors and then isolate the network, or the part of it that could be critical for data collection or their transmissions [48]. So, for intrusion detection it is necessary to consider these conditions, and it is not always possible to distinguish without any doubt, and in real time, whether the network is under attack or whether it is experiencing failures due to other external factors.

1.2.1.2. Summary

In conclusion, implementing security or QoS mechanisms in sensor networks is not trivial, even less when we intend to implement both types of mechanisms simultaneously. The nodes are resource-constrained and, thus, from a performance point of view, the mechanisms are very costly to implement. Actually, QoS mechanisms for sensor networks are simplified, generally focusing on extending the network's lifetime. Most of the efforts to adapt QoS traditional techniques to sensor environments are intended for *Wireless Multimedia Sensor Networks* (WMSN) [49]. However, it is important to point out that if security and QoS mechanisms could collaborate with each other and be properly integrated into WSNs, the advantages obtained would be huge. There is a key point here, which is the additional

difficulty of distinguishing a real attack from a change in the network due to environmental conditions, or changes in the network topology, for example, due to mobility.

1.2.2. Mobile Ad Hoc Networks.

Like WSNs, MANETs (Figure 1.3) are also composed of self-configuring devices connected wirelessly in multi-hop communications. However, MANETs are dynamic networks that can be composed of heterogeneous devices. Those devices are close to the user (e.g. tablet) and can store private data (e.g. photos, contact addresses, etc.), and for this reason they cannot be easily replaced, like it is the case of WSNs. Furthermore, the communication architecture for a sensor in general is cross-layered in order to optimise the available resources, while in a MANET, communication architectures may be used (for example, the TCP/IP architecture to allow the interoperability with traditional networks).

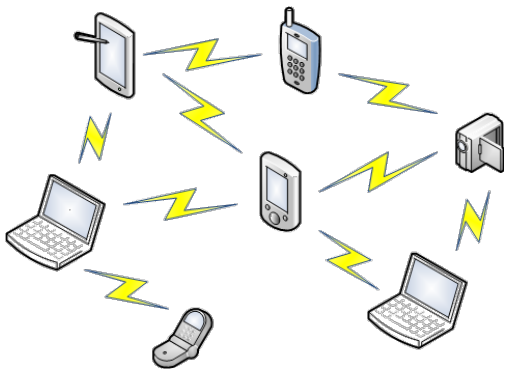


Figure 1.3: Mobile ad hoc network.

Indeed, in MANET scenarios, there are new security and QoS considerations to be taken into account. While the attacks on WSN can be geared towards falsifying the measurements taken from the environment, the attacks on MANETs, where the user can be a potential door to the infrastructure, may be intended to trick the user or obtain personal data. Heterogeneity of devices also makes even more difficult to establish QoS guarantees and to deploy security mechanisms. Most QoS models proposed for MANETs are influenced by the Integrated and Differentiated Service protocols (IntServ and DiffServ) [50].

1.2.2.1. Challenges in FI

1.2.2.1.1. Deploying QoS and Security Mechanisms. In [51], authors analyse the security threats in resource reservation (QoS signaling) in MANET, using the INSIGNIA and SWAN protocols, respectively, based on IntServ and DiffServ. In this case, while INSIGNIA ensures sufficient resources along the communication path, SWAN makes an estimation of available resources along the path. The paper concludes that, regardless of the protocol, an important problem is that reservation requests are accessible by any device with access to the transmission channel, which is of free access. In more detail, it means that several devices could identify these and other control messages and distort them or sabotage the resource reservation for their own benefit.

Moreover, device mobility makes it difficult to verify the legitimacy of a QoS request, and the limited resources make difficult the deployment of QoS monitoring techniques. Along the same lines, the paper [52] lists several security and QoS problems in MANET, but focuses on intrusion detection mechanisms to detect and prevent QoS signaling attacks. In [53] the authors focus on defining the DoS-resistant QoS (DRQoS) protocol, a QoS signaling protocol for MANETs that is resistant to some variants of flooding and over-reservation attacks. In order to do this, each node needs to store an entry in a state table for each stream of communication that attempts to transmit. This means that a node has an entry (i, j) for each neighbour node i and j that communicates through it. Managing these tables can be somewhat complex and costly given the dynamic nature of MANETs.

1.2.2.1.2. Self-organisation and Dynamic Nature. A particularly interesting feature of MANETs is their capability for self-organisation and the added advantage of being designed for highly dynamic scenarios. These factors have led to their study as networks to be deployed in critical situations. For example, in [54], the author defines a framework for secure real-time communications in MANET used for emergency rescue scenarios (e-MANET), adding authentication of the sender, integrity and confidentiality (using IPsec), and providing intrusion detection. Specifically, Chamaleon (CML) is proposed as an adaptive routing protocol for MANET.

The use of IPsec in MANETs is studied in [50]. The paper shows that, at MAC level, the frames would be protected using the IEEE 802.11i protocol, which adds protection hop-by-hop. That is, encryption is performed hop-by-hop, so it may require the intermediate nodes to have pre-shared keys or to be enabled to use certificates. The proposed solution is for a military scenario, where pre-shared key assumption is a feasible option, using a symmetric key, and only considers one QoS domain, so the problem is simplified. The IPsec AH header is modified to include the values *Data Services* (DS) and *Explicit Congestion Notification* (ECN) as well as an optional field that can be used by attackers, verifying the integrity of these data by using an *Integrity Check Value* (ICV).

1.2.2.2. Summary

Based on the information above, there are current approaches for adapting traditional QoS mechanisms (DiffServ, IntServ) to MANET, in order to perform the resource reservation and its maintenance (QoS signaling). Furthermore, QoS signaling protection is fundamental in avoiding DoS or similar attacks that have a negative impact on the resource availability in MANETs. Nevertheless, the dynamism in MANETs makes intrusion detection difficult, which should take into account the input and output of nodes in the network, as well as their mobility within it.

1.2.3. Cellular networks

Cellular networks are composed of cells (Figure 1.4), which are the physical space of coverage of a *Base Transceiver Station* (BTS). The BTS provides wireless coverage to all the mobile devices in its cell. The mobile devices can change cells while they are on the move and still be connected to the network through the BTS of the new cell.

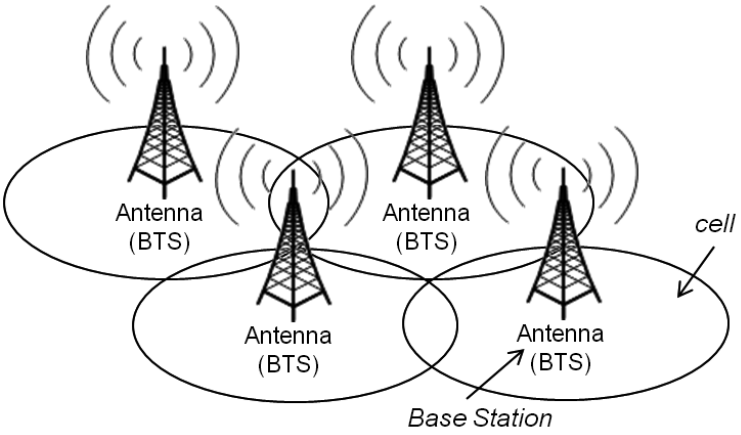


Figure 1.4: Cellular network.

In contrast to WSNs, cellular networks are dependent on service providers and need a network infrastructure to allow *Authentication, Authorisation and Accounting Services* (AAA Services).The accounting service, in particular, contributes to controlling the use of the network for subsequent payment. Moreover, while WSNs and MANETs are multi-hop networks, Figure 1.4 shows the single hop device-BTS, which means that the nodes send data directly to a specific access point.

1.2.3.1. Generations of Cellular Networks

It is worth mentioning here the *Global System for Mobile communications* (GSM), UMTS and *Long Term Evolution* (LTE), as each of them represent the beginning of a new generation of cellular networks. Moreover, the 5th generation of networks (5G) has emerged with new definitions for improving the network utilisation, and therefore the user's experience.

1.2.3.1.1. Second Generation of Cellular Networks. The first notions of security appeared with the GSM specification. Here, the security mechanisms can be divided into two types: the first one used for communication security, and the second one to protect the terminal against unauthorized use. As for the first type, the *Subscriber Identity Module* (SIM) card, and the *International Mobile Subscriber Identity* (IMSI) are essential mechanisms. As for the second, and in order to protect the unauthorised use of the terminal, the *International Mobile Equipment Identity* (IMEI), a unique identifier per terminal, has been defined. The IMEI can be used to remotely disable the terminal in case of theft or loss. The IMEI is written into the hardware of the mobile phone, and can only be disabled by a service operator.

1.2.3.1.2. Third Generation of Cellular Networks. As part of the third generation of cellular networks (3G), the UMTS specification introduces packet switching and, therefore, security and QoS improvements, such as, for instance, the use of public key cryptography or increased bandwidth. Over the years, mobile terminals have increased their complexity, not only as a measure of protection, but also expecting to increase user satisfaction, for

example promoting the use of new services and improving the connectivity of terminals to other networks (e.g. Internet) or devices (e.g. Bluetooth).

1.2.3.1.3. Fourth Generation of Cellular Networks. The fourth generation of mobile terminals (4G) is based on *all-IP* based technologies. UMTS Long Term Evolution (LTE) is the predominant technology for 4G. LTE is a 4G technology optimized for packet switching, with a simplified architecture that enables faster data transfer at low cost and energy [55]. One of the main focuses of 4G technologies is to offer higher bandwidth and support for different access technologies (e.g. IEEE 802.11) and multimedia applications [8]. It is under this umbrella where the multiple purpose devices and multiple input-output interfaces appear as a feasible alternative to multipurpose devices.

1.2.3.1.4. Fifth Generation of Cellular Networks. The deployment of the fourth generation (4G) of mobile networks is a reality, where the advantages of multiple devices have to be considered in the deployment of new solutions for the new market demanded by users' requirements. Moreover, new improvements and architectures have to be developed to increase the coverage of networks and the use of the spectrum. The current problem arises from the fact that there is a growing number of devices coexisting in the same environment, most of them with high requirements of network traffic [56].

These devices are wireless, what means that they use the same medium to communicate with each other, and this is a problem for several reasons. Firstly, wireless private networks coexist with other networks and are physically available to malicious activities from external intruders. Moreover, users in a network usually have a role which identifies their priority and privileges. In these cases, there is the risk that a user will take advantage of his/her role to overcome other users for an undefined period of time, or to impersonate a user to gain access to protected resources or capabilities. Furthermore, the wireless nature of the communication medium allows a dynamic number of users into the network. This overload problem is unacceptable in comercial environments where the networks are deployed.

As a consequence, the fifth generation (5G) of mobile networks has to consider these and other open issues and challenges. To do this, there are initiatives that propose the use of the military spectrum and other reserved frequencies when they are not in use, in order to increase the bandwidth utilization, thus encouraging new improvements. Moreover, the term *Green* takes care of the operational cost of deploying the future networks and the impact on the human resources. Therefore, in a 5G Green environment the requirements to provide 5G environments considering the Green perspective are carefully considered.

Relay networks are key in this new vision because the use of different types of devices, as routers, increases the network coverage and also reduces the power consumption of the device by using ad-hoc communication and behavioural-based techniques. For example, in order to predict the QoS in sections of the network, the QoS perceived by the user helps to adjust the data rate in those sections where the QoS is better, before reaching a section with poor QoS [57]. However, it entails several issues with regard to the security and QoS tradeoff. In particular, to ensure the privacy and confidentiality of the user during the whole process of handling the information acquired by the infrastructure is crucial for avoiding future security flaws.

1.2.3.2. Challenges in FI

1.2.3.2.1. Business Considerations. The majority of the studies based on 4G architectures highlight the approach All-IP on which they are designed as well as their security problems and the need for QoS guarantees. In [58], the importance of dealing with attacks that affect the performance and availability of cellular networks is studied; in particular, *Theft-of-Service* (ToS), DoS and IP spoofing attacks. Indeed, these attacks can damage the service provider reputation and this may incur the loss of customers.

To prevent these and other threats, the security mechanisms must be strengthened, but without forgetting that the indiscriminate use of resources could in itself become a threat to the whole system. Therefore, in [59], the combined use of the *Elliptic Curve Cryptography* (ECC) and symmetric key to address the vulnerabilities of a 3G-WLAN hybrid system is proposed.

The problem with cryptographic techniques is that they are difficult to implement in nodes and, moreover, they increase the cost of mobile devices, especially when secure elements are used for generating and storing the keys, providing anti-tampering capabilities. Note that the cost per unit affects the whole production chain. Hence, implementing these solutions in mobile platforms will be profitable only when these are used en masse, and if finally they are demanded and accepted by the user.

1.2.3.2.2. Alternatives to Cryptography in 5G networks. Several authors consider that an alternative to cryptographic algorithms is the selection of the nodes/relays based on low-level security metrics such as secrecy rate, secrecy capacity or secrecy outage probability. The aim is to select, depending on the network, the best range to transmit the information, based on the maximum values of the chosen metric and additional characteristics. These mechanisms are known as physical security mechanisms. The focus is to avoid eavesdropping, and the restriction is that the mechanisms are based on the knowledge of the global state information of the network, and in many cases these assume the knowledge of the *Signal to Noise Ratio* (SNR) and capacity of the eavesdropper, which is not always possible.

These physical techniques may consider the use of cooperative jamming between the nodes to avoid eavesdroppers. The cooperative jamming consists of the generation of gaussian noise, which can be decoded by the receptor that knows how the noise was generated. This means that the parameters for generating the gaussian noise acts as a key. In many cases, it is not possible to apply such techniques because the coordination between the nodes requires too much energy.

However, detecting diverse physical attacks is feasible by analysing certain network parameters at the interface layer. For example, in [60] the authors analyse different types of jamming and the suitability of typical network parameters, such as the *Packet Delivery Rate* (PDR) or the *Packet Sent Ratio* (PSR) to identify these attacks. As in previous cases, where the behaviour of the network is dynamic, it is very difficult to distinguish between a jamming attack and a poor QoS of the network, given certain environmental conditions.

Finally, security at the physical layer, in theory, is less intrusive than cryptographic techniques, because the protection is adjusted to the restrictions in the node considering critical parameters which define the core of the node; for instance, the communication range, and the energy.

1.2.3.2.3. Security and QoS Tradeoff in Green 5G Relay networks Although 5G relay networks bring with them several benefits they also pose challenges to be Green, as analysed in [5, 61, 62, 63]. For example, in [5], diverse methodologies for relay selection in 5G networks are described, and analysed from the point of view of performance and cost. Moreover, in [64] four challenges are identified that need to be addressed in 5G Green environments: (i) the increasing volume of data, (ii) the growing number of devices, (iii) the diversity of applications and (iv) the need for energy-aware solutions. All these factors affect the security and privacy. For example, the management of the information has to be done considering privacy laws; the growing number of devices increases both the chance to misbehaviour, and the computational cost of controlling these environments; moreover, the heterogeneity of the devices complicates the deployment of security solutions without affecting the energy. In [65] the sources from where the data in 5G environments is generated are classified at different layers, from where it is possible to obtain the parameters to be analysed. For example, the user's information handled by the cells can help to identify the user mobility behaviour. In turn, it is possible to configure applications to improve the user's experience, as in [57], where an approach for balancing the data rate based on the user's context (e.g. location, time of the day) is provided, introducing tradeoffs to power consumption. How these data are obtained - user's consent - and how the information is handled must be the focus in these cases of the security mechanisms. Moreover, to maintain the trustworthiness of this information is not trivial. If the measurements about the user's behaviour are affected, then the adjustements given by the balancing system will be useless.

In addition, delay-bound QoS constraints in 5G mobile networks are analysed in [66], and in [67] the challenges for handling high-volumes of data efficiently, considering energy restrictions and QoS support for real-time applications are discussed. Again, these analysis do not consider that data must be protected with security mechanisms to ensure that the system remains secure. Therefore, when security mechanisms are deployed – in a real environment – said analysis must be repeated to consider the effect that said mechanims have on the QoS parameters. The latest trends focus on defining *Software Defined Networks* (SDN) based architectures for 5G networks [68], where the main criteria should be improve the performance of the solutions. However, security must be considered in these scenarios too. Unfortunatelly, the analysis of security and QoS parameters combinated is not trivial in these systems. Not only heterogeneous parameters must be considered, but also these have to be analysed considering high level descriptions provided by the abstract solutions deployed on SDN-based environments.

1.2.3.2.4. Internet Protocol-based Mobility in Cellular Networks. IP-based mobility is also a hot topic in this area. For example, in [8] the architecture SeaSoS is proposed for 4G networks. SeaSoS integrates QoS signaling, AAA Services and MIPv6 for 4G network infrastructure. SeaSoS also envisages the possibility that the end user or network operator can change the network attributes dynamically (e.g. using HMIPv6 instead of MIPv6) in order to facilitate the interaction between heterogeneous networks. The paper shows a comparative table with the security, QoS and mobility mechanisms used in other studies based on 4G architectures. From this comparison, it is noteworthy that most of the mobility protocols used in the solutions are based on MIP, with the exception of W-SKE protocol

[69], which focuses on efficient key management (creating, distributing, etc.).

Along the same lines, in [70] Tiny SESAME is proposed. Tiny SESAME is a security mechanism based on the SESAME architecture for distributed systems that extends Kerberos with additional security mechanisms. Tiny SESAME is a security mechanism based on dynamically reconfigurable components at runtime, so it is possible to add on-demand components and remove them if not needed at any given time. As a restriction, the mobile client should be able to run Java code, which is too aggressive for resource-constrained devices, so an open challenge in such work is to migrate the actual scheme to *Java 2 Micro Edition* (J2ME), a more lightweight language for software development.

1.2.3.2.5. Behaviour-based Context. In [71] the need to provide QoS techniques adaptable to user needs and the importance of developing secure and efficient IP-based services is highlighted. To this end, authors propose the use of cognitive techniques to provide intuitive responses to the changing environment, offering the possibility of selecting a set of parameters, appropriate for the context of the device. For example, while the user is on the move, the system could obtain information about their neighbours' devices and report the optimal performance and security settings based on the availability of computer resources. For performance metrics, authors take into account the distance, the number of hops to destination, the *Bit Error Rate* (BER), the PDR, the signal strength, the energy, the time response, the prioritisation of messages and the *Call Dropping Probability* (CDP). The latter parameter is specific to 4G networks. These techniques, in conjunction with techniques such as keystroke-based authentication [72], help to provide a better service to the user.

1.2.3.3. Summary

The coexistence of QoS and security mechanisms in future cellular network architectures is therefore acceptable, as are schemes that provide the user with mobility without loss of connection. The ideal scenario is one which allows these technologies to responsibly coexist with each other in order to seek maximum performance and network efficiency. In addition, these mechanisms will continue to be refined to make them resistant to new threats due to the IP-based architecture of the new generations of mobile telephony, without forgetting that the end user plays an important role in the adoption of these new technologies. Moreover, it is fundamental to identify those contexts where the use of physical security mechanisms is enough to prevent threats in the environment or whether high-level security mechanisms are required instead.

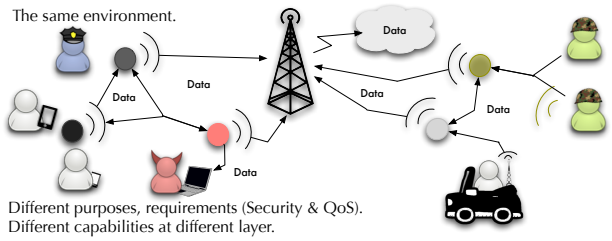


Figure 1.5: Green 5G Relay Environment.

The particular case of 5G relay networks is very interesting because it combines the ad-hoc behaviour that was discussed in Section 1.2.2 and the powerful infrastructure of a cellular network (Figure 1.5). Related to the security and QoS tradeoff in said environments, we have seen that 5G Green approaches handle a wide range of parameters at different layers. So, the security and QoS tradeoff must be analysed considering (i) how to combine information from independent sources, and (ii) how to work with partial information - integrating new information when it is known. For example, when it is known that certain security properties (e.g., trust) must be provided but how these will be provided (the specific mechanisms) is not known at an early stage. How to answer the questions (i-ii) is discussed in Chapters 2 and 3.

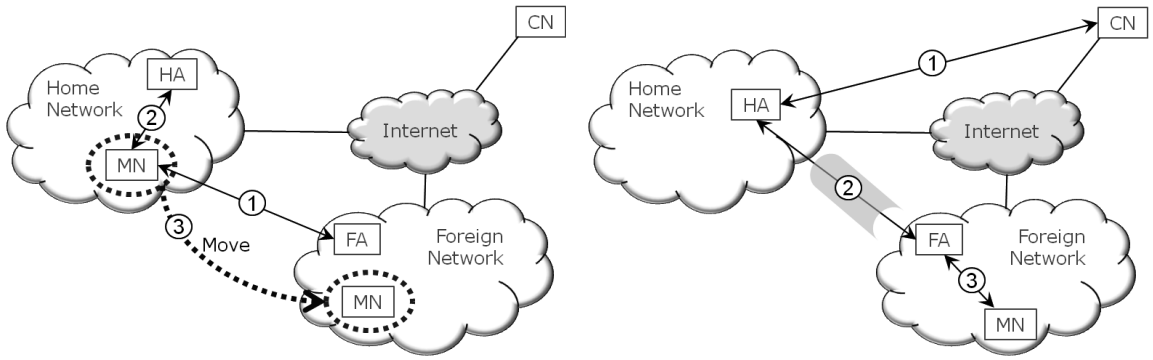
1.2.4. Collaboration: Approaches and Concerns

1.2.4.1. Internet Protocol

There are several initiatives for adapting the Internet Protocol to power-dependent devices [73]. In the following, two crucial aspects to ensure the survivability of IP in future networks are discussed. On the one hand, the use of personal devices makes the mobility capabilities crucial in future architectures. Moreover, the mobility itself entails severe security risks to be addressed in future deployments. The secure protocol for IP networks should be adapted for personal devices in order to be useful in these new paradigms.

1.2.4.1.1. Mobility Management Protocols. Mobile IP (MIP) is a standard communication protocol designed to enable mobile users to move from one network to another while maintaining a fixed IP address [74]. MIP permits macromobility, namely, the change over from one network (home) to another (foreign), transparent to other users (Figure 1.6). This is possible because the IP datagrams are forwarded from the home network to the foreign network. However, the use of intermediary entities for traffic redirection increases the overhead in both networks, home and foreign. On the other hand, during the migration process (known as handoff or handover) some packages addressed to the mobile node can be lost. The performance problem is compounded if we also consider the security of the network because, in that case, it may be necessary to establish AAA Services, increasing the response time when the user's intervention is required or for accessing to the information handled by remote databases.

In general, such controls are established before the network change is in effect, and can involve multiple participants. Thus, the handover process is critical to network performance. If this process drags on, then the connection may be interrupted. Moreover, the user may perceive a poor connection, even detect the change of network, affecting to transparency. Furthermore, reserving network resources before data transmission ensure some QoS guarantees for certain users, but can be the focus of several problems when it is not properly handled [8]. One of the problems is that the resource reservation process (QoS signaling) implies a waste of network resources in itself, particularly with the bandwidth required for network communication and the energy consumption when wireless communications takes place in the nodes.



(a) Networks before mobility

(b) Networks after mobility

Figure 1.6: Mobile internet protocol.

The consumption of resources when performing QoS signaling is compensated because posterior communications have the availability of such resources guaranteed for data transmission. However, in mobility scenarios applying QoS signaling techniques may not be justified when successive changes in the network may involve the new calculation of new routes for the delivery of data. Thus, the network resources can be consumed in maintaining the resource reservation indefinitely.

In addition, the *Media Independent Handover* (MIH) technology (IEEE 802.21 standard), allows the handover among different network technologies, denoted as vertical handover. The MIH protocol provides low-level mechanisms required to improve the performance of MIP. One of the motivations of MIH is to enable a common information service that provides a global network map with data about the available networks within a location (e.g. a cellular network may indicate the presence of a suitable WiFi station). This information service would be managed by the operators, and users could have access to the information via their mobile devices. MIH defines its own messages to the MAC layer. For example, a client with MIH could change the connection from WiMAX to 3G WWAN, without loss of connection [75].

1.2.4.1.1.1. Current Approaches. As we have seen, several studies consider MIP to be an ideal mobility management protocol for network convergence, although as mentioned, MIH is more generic and defines frames designed to manage mobility at the MAC level.

Regarding the MIP, current efforts primarily focus on reducing the handover time and solving those problems caused by the use of more than one domain (e.g. change of domain). For example, the secure QoS-enabled mobility (SeQoMo) architecture has been developed to provide security and QoS support in MIPv6 [76]. The idea behind this architecture is to mitigate the high latency and overhead during the handover, while the network infrastructure is protected by security mechanisms such as authentication or authorisation in conjunction with QoS processes. Indeed, the security in MIP is provided by additional protocols like, for example, the aforementioned IPsec or AAA services. There are several approaches that consider MIP as a part of the infrastructure for interoperability in which security and QoS are key requirements.

In this sense, [77] highlights the importance of the reliability of the source from which the information is obtained in MIH networks, the need for a secure channel between the user and the end point, and the handover optimisation, especially when handover is performed through different administrative domains. In [78], a tutorial on security in MIH networks, indicates that due to the large number of different AAA domains, a pre-authentication solution in these domains is required. In [77], such a proposal is adopted, as well as the pre-configuration of the terminals.

MIH is also used in [79], where the integration of WLANs IEEE 802.11 and WMANs IEEE 802.16 (WiMAX) is analysed. The paper focuses on managing the handover process, where they state that the handover decisions should be based on several factors, among them, the QoS and security support. Moreover, in [80], MIH is used in the handover process between WiFi and WiMAX. The authors recommend that the nodes running critical operations (e.g., security decisions that effect the handover process) have to form part of the core of the network to decouple *Access Points* (APs) and BTSs. This decoupling is natural and understandable even for heterogeneous networks. For example, in order to include security properties such as the authentication of the terminal, the system has to be able to support efficient and secure data management (e.g., the IMEI and IMSI in cellular networks), but if the communication architecture is distributed with different administrative domains, such tasks can be too complex for the BTSs and APs.

In fact, in order to properly identify users or their terminals, the system has to store unique identifiers and, in case of loss, theft or terminal extinction, the system should disable the utilisation of these data to avoid fraudulent use by unauthorised parties. Moreover, in distributed systems, the management is more complicated as well as expensive. Taking into account that the user can be directly harmed by system failures or personal data leaks, such problems can degenerate into monetary losses for the service providers, owners or coordination managers of the infrastructure and physical media. Therefore, assigning data management to the most powerful and robust services is not a bad approach, but always bearing in mind that this information could eventually pass through the BTSs or the APs, and that intruders have different ways of obtaining information, such as traffic analysis, or create deliberate damage by performing attacks that affect the performance or availability of services, such as DoS or ToS attacks.

1.2.4.1.2. Internet Protocol Security IPsec integrates security features such as source authentication, data integrity and confidentiality, and avoids packet replay attacks by using the sliding window mechanism, although it affects the performance. However, it does not implement non-repudiation, protection against DoS attacks or traffic analysis.

IPsec can be added to the versions 4 and 6 of IP using additional headers to provide aforementioned services [81]. In particular, it uses the *Authentication Header* (AH) and the *Encapsulating Security Payload* (ESP) protocols. The first one is an authentication protocol, while the second combines encryption and authentication. The ESP and AH protocols enable IPsec to create secure tunnels for communication, the *Virtual Private Networks* (VPN). Specifically, IPsec defines two ways of packaging: transport and tunnel.

On the one hand, in the transport mode, the respective headers are located after the headers that have to be read by the routers (IPv6 head and optional headers) and before the

payload, in case of cyphered data that should not be read until destination. On the other hand, in the tunnel mode, the entire original IP datagram is encrypted and placed as data. Then, a new IPv6 header is created with the basic data to carry the packet to its destination. This mode allows the full authentication of the IP datagram, while the transport mode only authenticates the payload.

From a performance point of view, the transport mode requires less time to complete the data packaging and the final datagram is smaller. In many cases, authentication of the whole datagram does not mean a significant advantage. For example, if the probability of attacks is low, then the transport mode can be a better option than the tunnel mode. Moreover, no matters the packaging mode, at the beginning of the communication, IPsec requires extra volume of traffic for negotiating the *Security Associations* (SA) with the rest of the nodes and, in some cases, the communication is avoided because of the different criteria in the nodes. The most restrictive mode in IPsec avoids communication with nodes without IPsec enabled. Hence, end-to-end communication is not possible if an intermediary is unable to provide IPsec capabilities.

Intuitively, in resource-constrained devices, the use of IPsec is highly limited due to the resources in the node. Not only are computational and energy resources limited, but also the functionality that they can provide. Some approaches along these lines for adapting IP networks to the Internet of Things considering security restrictions are [15, 82, 83].

Finally, some problems with using IPsec combined with QoS mechanisms are discussed in [50]. For example, QoS options are listed in the header of IP datagrams without being encrypted, and therefore are exposed to being interpreted by an attacker. Moreover, in case that the QoS options have been encrypted, the intermediate nodes cannot use them without preprocessing for decoding. For example, the Differentiated Services (DiffServ) and DiffServ Code Point (DSCP) fields are present in the IPv4 and IPv6 headers to indicate the behaviour of the intermediary routers (hop-by-hop options).

Other options related to QoS, such as the bandwidth reservation and flow differentiation (not classes, but using a unique flow identifier), are specified by optional IPv6 headers. If the whole package is encrypted, then these options are useless unless the routers integrate the required functionality to decrypt and encrypt the package (e.g., using a preshared key or authorisation certificates), but in any case it introduces communication delays, even more when the optional headers are used.

Another drawback is that some protocols require flow identification (e.g., IntServ) and therefore keep the source and destination IP addresses, port numbers and the protocol identifier visible. This implies that these data could be captured by any sniffers or traffic analysers in the network. There is also the disadvantage of datagrams received out of order, which IPsec tries to compensate for by using different SAs for different classes of traffic.

1.2.4.2. IPv6 over Low power Wireless Personal Area Networks

Low power Wireless Personal Area Networks (LoWPANs) are low cost communication networks that enable wireless capabilities in resource-constrained devices [84]. Specifically, the aim is to reduce the power consumption and satisfy throughput requirements in those devices that connect the physical environment with high-layer applications (real-world applications). Therefore, the typical devices in LoWPANs are wireless sensors and actuators,

and the systems that specifically concentrate on managing physical information are known as *Cyber Physical Systems* (CPSs) [85]. These systems are growing in popularity because of the cost and size of the devices used for this purpose and their implication in early response systems to mitigate the effect of failures, threats or attacks in the networks [86, 87].

LoWPANs are characterised by their use of small packet sizes and low bandwidth, as well as star and mesh topologies ad-hoc communications to avoid the pre-defined location of devices. Security is considered at link-layer (AES-CCM), with the cost of increasing the overhead, reducing the number of octets available per packet from 102 to 93 (AES-CCM-32) or 81 in the worst case (AES-CCM-128).

However, LoWPAN is limited because the cooperation with powerful devices is unavailable, unless these powerful services implement LoWPAN, what is very difficult considering that this may imply changes in deployed and stable solutions. Hence, the adaptation of LoWPAN to IP (6LoWPAN) enables this cooperation. Furthermore, such adaptation is based on IPv6 to take advantage of the addressing space and the new advantages in terms of security and QoS.

Alternatives to 6LoWPAN are ZigBee and *Machine-to-Machine* (M2M) [88]. ZigBee specifies a vertical protocol stack solution similar to Bluetooth, that works on the IEEE802.15.4 protocol. Nevertheless, ZigBee has been developed taking into account industrial considerations and, therefore, to be useful in controlled networks without scalability problems and without connection to the Internet. Unlike ZigBee, the approach for M2M is more generic. It considers the remote control of devices over the Internet, in general, using cellular modems integrated inside embedded devices and Internet-based back-end systems. Therefore, in M2M there are different devices with different purposes, and some of them connect to the Internet using the back-end system. So, as IP is native in 6LoWPAN, the collaboration with M2M devices is possible.

Common problems to be solved in previous low-power integrations with IP are security and QoS. Besides the adaptation required for IP to be embedded in resource-constrained protocols, security has to be enhanced so as to consider threats related to the Internet and coexistence with powerful devices [18, 89, 90].

1.2.4.3. Software Defined Networks and Virtualisation

Software Defined Networks (SDNs) separate the data plane from the control plane, simplifying the orchestration of resources, and reducing the deployment costs. This is different from *Network Functions Virtualisation* (NFV), that distinguishes logical services from physical resources and is directed to handle the resources moving them to different locations when required [4]. But despite their differences, they had one important requirement in common: in these contexts diverse technologies and methodologies have been proposed and analysed, increasing the need for high-level composition of services, where the security and QoS tradeoff must be considered.

The benefits of these technologies are numerous; for example, the virtualisation of hardware components mitigate the hardware limitations [3, 91, 92] to the cost of memory, virtual storage, requirements for network performance (bandwidth, throughput, etc.) and time processing depending on the machine where the services that emulate the hardware are deployed. The cost in resources and maintenance is fully justified, even more when one considers that

these solutions are devised to be combined with Cloud Computing technologies and virtualisation. Security problems regarding privacy and data protection to satisfy the different data protection laws are the main issues to be solved in this area.

Furthermore, software defined networking can help to provide 5G capabilities [68, 93]. For example, in [93], an analysis on open source platforms, applications and tools for SDN and 5G is provided. This is only an additional proof of the role that these networks are taking for the future.

1.2.4.4. Collaboration based on Natural Behaviour

It is essential to identify those collaborations that are natural so that the network is deployed to do what it usually performs while providing novel advances to the rest of the networks. For example, diverse studies consider the use of cellular networks to provide secure access to the Internet. Indeed, some analyses claim that the future of cellular networks is strictly dependent on their integration with IP networks [8], considering the use of MIP for 4G mobility management, or all-IP networks in general [94]. However, this collaboration based on natural behaviour entails various challenges to be addressed because, although the collaboration provides different advantages to the participants, the process to achieve the collaboration introduces modifications that need to be carefully considered. For example, in [95], authors elaborate on the problems that may arise in the integration between cellular networks and WLANs, choosing MIP as the mobility protocol. One of the biggest challenges to be addressed in order for MIP to be used by various technologies and domains is reducing the handover time in order to improve the response time for users.

Obviously, the great advantage of 4G-MIP integration is to be able to use global information handled by cellular networks. This is very useful for security mechanisms; for example, to authenticate the user or his/her terminal, or to perform accounting and billing tasks. Moreover, the integration of cellular networks with MANET is also natural. This alliance provides both security and flexibility advantages. On the one hand, the global information handled by cellular networks can increase the security in dynamic networks, where retaining information about the activity of the mobile users is very difficult. On the other hand, MANETs lack organisational structure and are highly dynamic, so they are currently much more flexible than cellular networks. For example, the approach followed in [96] enables the MANET devices to connect to a cellular network, taking a step towards the cooperation between heterogeneous networks and convergence.

One additional example of collaboration is in [54], where a three level communication architecture is proposed. At the lowest level, there are *emergency MANETs* (eMANET), in the intermediate level semi-mobile nodes, and at the highest level, a gateway to access an IP Cloud. The proposed scheme targets emergency rescue situations. This gives us another perspective of the network convergence and the importance of self-configuring and self-organised networks such as MANET. Furthermore, the inclusion of communication networks to take measurements of the affected environment, such as WSN, can help to prevent the rescue services taking unnecessary risks; for example, warning of high levels of gas, or if there is a risk of nuclear leaks, in the case of a nuclear power plant.

The integration of cellular networks and WSN is proposed in [97], where the 4G paradigm is presented as a combination of heterogeneous networks where the sensors are included. In

this sense, the sensors could contribute to industries (e.g. nuclear plants) or the home, and would use cellular terminals as gateways for access to IP networks.

1.2.4.5. Always-on as a Need

Network integration brings about benefits beyond collaboration between networks for exchanging information of interest or the use of services such as providing access to the Internet. In fact, giving the user the possibility to be always connected to the Internet is very interesting because it favours business opportunities for service providers and small and medium enterprises that are migrating their traditional business plans so as to take advantage of the latest technologies developed for small cities.

Along the same lines, in [98], an architecture to integrate heterogeneous wireless systems used to provide ubiquitous high-speed services to mobile users is proposed. Among the technologies covered are WLANs, UMTS and satellite networks, and IP is used as the protocol for the interconnection. The security is implemented through specific algorithms for authentication and billing, and MIP is used to facilitate the roaming between different wireless systems. Both security and mobility are managed by a third party, and each operator needs to establish a *Service Level Agreement* (SLA) with it. The idea behind this approach is that the user device connects to the network available with the capabilities to provide the best service for data transmission. For example, assuming that the user's device supports various forms of connection (WiFi, 3G, satellite), if the user is in a WiFi-enabled shopping center, then the device can use the WiFi access point of the commercial centre for accessing the Internet. However, if that access is not available, then the device could try to use 3G to connect, and finally the device could even use a satellite link, although this last option consumes far more resources than the previous two.

Another approach that seeks to exploit the expected host of alternatives for connectivity is the one proposed in [99]. More specifically, the study addresses the integration of cellular networks (e.g., 1G, 2G, 2.5G, 3G, IEEE 802.20), WLANs (ej.IEEE 802.11a/b/g, HiperLAN/2), WPANs (ej.Bluetooth, 802.15.1/3/4) and WMANs (ej.802.16). It also takes into account the existence of MANET, that can act as routers by using the WLAN/WPAN interfaces, and the AAA services to provide security. The elements responsible for managing the load balancing and handover are the BTS and AP. The drawback is that the protocol stack must be modified to include an interface for each piece of technology involved in the mobile network (e.g., cellular networks and 802.11 require different MAC, link and physical levels).

1.2.4.6. Performance, QoS and Security

Performance is interpreted as QoS in several papers. However, performance and QoS are not the same thing, although they are closely related. For example, the heterogeneity of devices sharing the same environment can affect network performance by increasing the risk of collisions. When this occurs, the QoS can be damaged, however, it is also possible that QoS management will not be required to solve the problem. Multi-hop communications can help to reduce the risk of collisions while saving energy because the transmission range is less than when directly connected to the BTS or the AP, and therefore requires less power

consumption. Moreover, as the transmission signal does not affect the whole network at the same time, the risk of collision is concentrated in specific areas at any given time.

Indeed, traditional QoS mechanisms cannot be directly applied to resource-constrained networks. For this reason, multi-hop communication and energy-efficient techniques should be applied for tiny devices. Following these lines, several studies concentrate their efforts on estimating the optimal number of hops in a communication [99].

A clear example of the negative effect of QoS in performance is on the use of traditional QoS signaling mechanisms. These cannot be deployed in certain networks precisely due to the additional traffic required. However, it is obvious that improving the performance increases the probability that the system offers a better QoS.

The most widespread mechanisms for providing QoS guarantees in IP networks are *DiffServ* and *IntServ*. A very important aspect is the adaptability of QoS mechanisms to environmental changes. For example, [100] defines QoS policies that are automatically included in the configuration of network devices, with the possibility of adapting them as network conditions change. The QoS is provided by using the DiffServ provisioning technology, which incorporates mechanisms for classifying, managing network traffic and providing QoS guarantees over IP networks. The tools of this architecture are used to provide QoS in GESEQ [101], a generic model of security and QoS which uses IPsec to enable secure communications with the deployment of VPNs. The QoS mechanisms used in GESEQ helped to improve the performance, quantified according to the latency, jitter (delay variation) and packet loss parameters.

Note that it is a common practice to evaluate the QoS based on network performance, and this is due to the fact that QoS can be greatly compromised if the performance is poor. So, the aim of QoS mechanisms is to orchestrate the network resources to be able to use them without affecting each other. Security mechanisms can also affect performance, because they add network traffic that may cause overhead. So, the security system can interpret the poor connectivity as an attempt against the safety of the network and take action.

In conclusion, deploying QoS and security mechanisms can negatively affect the performance, and therefore also themselves, but when they are properly deployed, taking into consideration the requirements given a specific scenario, the benefits can be laudable. If security mechanisms are able to collaborate with QoS mechanisms and these can be adapted to enhance the performance, then this may help the convergence, reduce the false positives due to changes in the network's behaviour and enable the QoS mechanisms to be scalable and used in resource-constrained networks. Moreover, if the QoS mechanism considers the overhead caused by the security mechanisms, it can help to predict certain traffic demands, improving the orchestration of services.

1.2.4.7. Summary

We have found that the current approaches are based on tree network architectures, where the mobile nodes are the leaves and the root is an element acting as the gateway for internetwork communication. This is natural, due to the benefits of centralised information. However, an alternative approach is to move towards more distributed and dynamic architectures in order to allow any device to connect to the Internet by itself. Nevertheless, we must tackle several difficulties, and maybe the most important is the coexistence and coop-

eration of services that belong to different domains. In other words, currently, it is feasible that different domains use different mechanisms, protocols and policies, to provide QoS and Security, but the problem is that they are not necessarily interoperable.

Moreover, in a heterogeneous environment, constrained-resource nodes can coexist with more powerful nodes that could launch an attack that a constrained-resource node is unable to avoid because of its limited capabilities. Another problem is the cost of deploying distributed trust schemes or other security mechanisms that require access to users' data or the mass storage of information in order to be effective. Furthermore, improving the handover efficiency (both horizontal and vertical) is crucial for the integration of heterogeneous networks, as well as enhancing the security mechanisms to protect the infrastructure and its users without negatively impacting on the performance of the handover procedure. However, despite their relevance, both aspects are open challenges.

It must be noted that the security and QoS tradeoff should be done considering dynamic high-level definitions as is required by the last trends (e.g., SDN and NFVs). Future networks trends to be more conceptual and abstract, and therefore, not always is possible to predict with great accuracy the final devices or components that will form the network.

Finally, we have to remember that due to the cooperation between service providers, it is fundamental to take precautions to avoid unfair competition. Furthermore, the traceability of information and possible data leaks are aspects to be carefully considered in environments where user data are handled.

1.3. Goals of this Thesis

Notwithstanding the clear dependency between Security and QoS in heterogeneous networks, the current analyses are still focusing on isolated tests on specific parameters. Therefore, it is very difficult to analyse the parameters taken from different analyses simultaneously, even more when these parameters belong to different abstract layers. For example, most of the solutions directed towards the composition of parameters are developed to service composition for web services.

Based on our previous analysis, focused on the main topics and challenges in the candidate networks to be part of the FI, we have seen that the specific purposes of the networks and the specific characteristics of the devices difficult the analysis of the security and QoS tradeoff in said heterogeneous environments. Although specific analysis are possible within each network or context, the ideal scenario is when this analysis can be enhanced with new information about new capabilities, dynamically. The current solutions are unable to provide this type of dynamic behaviour.

Furthermore, in the FI, the heterogeneity of devices enables the composition of capabilities inside the devices that provide useful functionality that must to be considered in the analysis. These capabilities may belong to the application layer (e.g. certificate), but may also be present at the physical layer (e.g. anti-tampering mechanisms), and at service layer (authentication service, multimedia capabilities).

In this thesis, we focus the discusion on how to assess the security and QoS tradeoff, considering the requirements in FI environments. In particular, we suggest a parametric analysis for enabling: 1) the definition of environments or systems in the different sub-

parts that may be integrated in different scenarios or contexts, and 2) identify the steps for integrating the parameters based on the context and the properties and functions that may be applied for this purpose. The focus of these points is to be able to provide the analysis of dependencies between security and QoS in FI environments, which is the main objective of this thesis.

1.3.1. Direct Contributions

The direct contributions of this thesis to security and QoS analysis are as follows:

O1. Provide an overview of the main topics and challenges to be addressed in the Future Internet and the Internet of Things related to Security and QoS tradeoffs.

O2. Provide a classification of relevant security and QoS parameters, dividing the behaviour into general context, where the main concerns are the same in the different networks, and specific contexts, where the individual requirements for the survival of the networks are addressed.

O3. Include the subjective perception of the parameters in the context in order to allow the identification of the main parameters based on that context. With this aim, we identify that the parametric-based approach is the most suitable to analyse the Security and QoS tradeoff in the FI.

O4. Provide a general model to describe scenarios in the FI based on their parameters and relationships, and provide a set of general parameters of security and QoS that can be adjusted to the needs of any user that makes use of the model.

O5. Provide the steps to develop a tool which implements the model, and finally we implement a prototype of the tool, for handling the set of security and QoS parameters provided.

O6. Define the integration/extraction of contexts as part of the prototype, and define the architecture of the model to be easily modified and enhanced by other developers.

O7. Provide a recomendation system for the tool which uses artificial intelligence techniques to make recommendations based on the dynamic information generated by the model, in turn based on a set of goals and requirements selected by the user.

O8. Provide a modular design of all the processes in order to allow the modification of any of the files, separately. Even the recomendation system engine may be modified to provide additional information not considered in this thesis.

O9. Validate the applicability to future heterogeneous environments, providing relevant use cases to the problem.

1.3.2. Outline of this Thesis

This thesis is divided into six chapters, a structure that considers: (i) the candidate scenarios to be part of the FI, (ii) the parameters of security and QoS to be evaluated, (iii) the definition of a model for assessing the Security and QoS tradeoff in the FI, (iv) the context and the dynamic integration of new information, (v) the implementation of a new tool for handling the elements defined in the model and, finally, (vi) the analysis based on two use cases in FI. Figure 1.7 shows a general overview of the structure of this thesis.

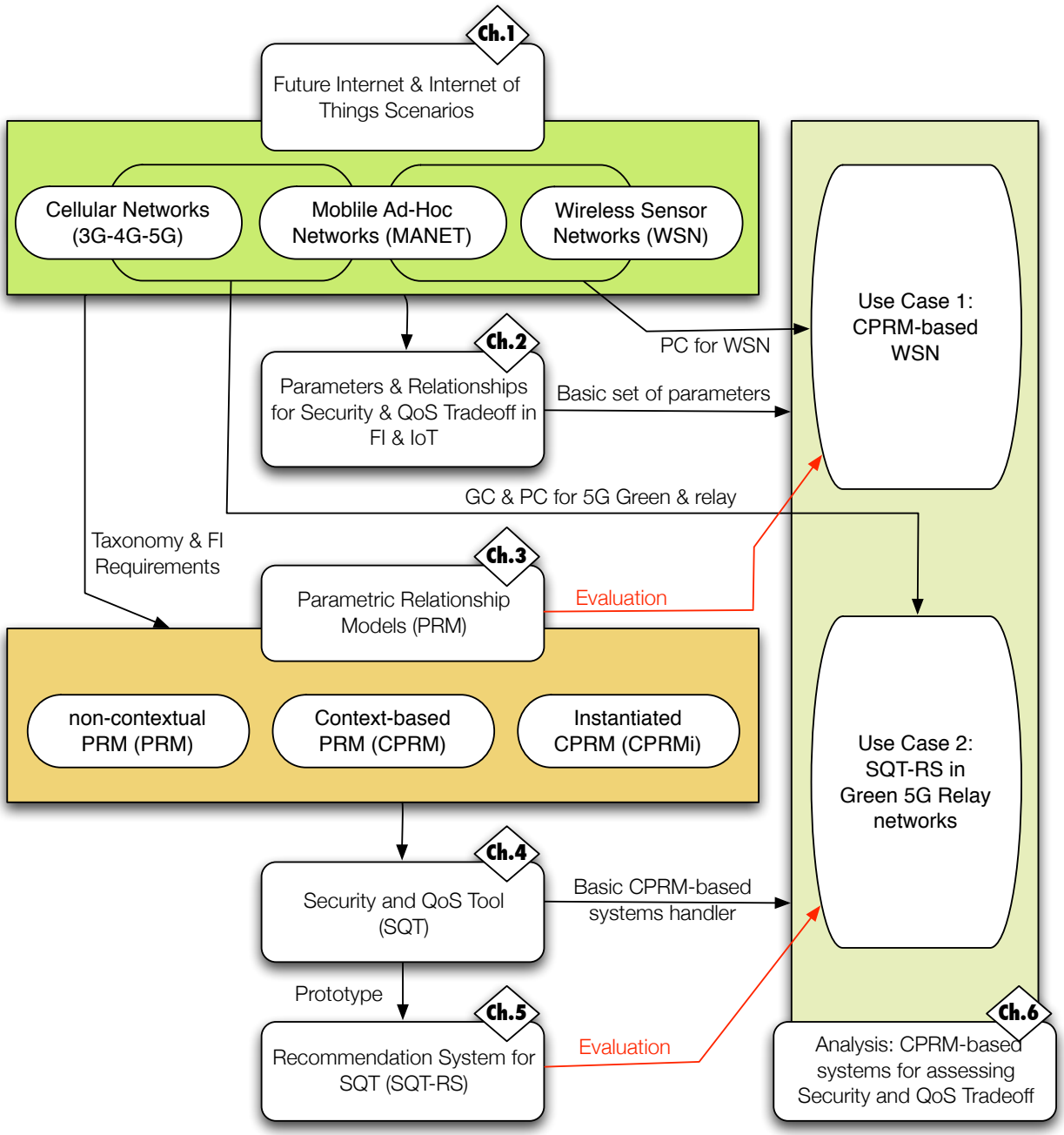


Figure 1.7: Chapters map.

The first two chapters are an exhaustive analysis of security and QoS mechanisms in future networks (**O1**, **O2**). As could be observed at the beginning of this chapter, we have chosen three tyeps of networks as the most representatives: WSNs, MANETs, and cellular networks. We study the interoperability and dependencies between security and QoS parameters **[IC1]**, identifying those that are common to the networks and those others that are particular and specific **[J1]**.

The results show: 1) the lack of mechanisms to properly evaluate the security and QoS tradeoff in heterogeneous networks of dynamic composition; 2) the complexity when parameters at different abstract layers are evaluated together; and, 3) the need to separate the contextual information into two types of contexts: general or common aspects to the networks in convergence, and specific to the technologies in the scenario considered. The fact is that current solutions do not consider these three points and, therefore, we have developed a model to solve 1, 2 and to integrate 3.

In Chapter 3, we follow a parametric-based approach to carry out an analysis of the security and QoS tradeoff, when integrating the different security and QoS mechanisms that may exist in the FI. To achieve this, we define the *Parametric Relationship Model* (PRM) for definition of parameters based on their type, layer and relationships (**O4**). The model also defines a set of operations that are used to derive information about the dependence and impact of the parameters on the system. As an example, we define a mobile platform environment using this model **[IC2]**, and derive information using the formulation defined in it **[J2]**. Moreover, the model is enhanced to integrate contextual information, using the *Context-based Parametric Relationship Model* (**O6**). The CPRM allows the dynamic integration of information in the parametric model. This dynamic information is given as contexts, that can be general or specific to the environment **[IC3]**. Moreover, the model also considers the subjective and non-subjective values for the parameters in the contexts (**O3**).

The final model is implemented in a prototype, using MATLAB, as well as scripts to describe the different components defined in CPRM (**O5**, **O8**). The prototype implements a handler of CPRM-based systems, that is called *Security and QoS Tradeoff Tool* (SQT) **[NC1]**, defined in Chapter 4. The dynamic integration of information has a cost in performance **[IC4]**, and, therfore, some alternatives for mitigating the effect in performance are provided.

However, SQT is limited because of the visual representation of the data, which is based on graphs. When the model integrates several parameters and relationships, distinguishing the differences among the different configurations becomes very difficult. To overcome this limitation, in Chapter 5 we propose a recommendation system for SQT (SQT-RS) **[IC5]**, that is defined based on the properties satisfied by CPRM (**O7**, **O8**). SQT-RS provides recommendations in a language that the user can easily understand, making it easier for them to select the set of goals and requirements using a tool that has been developed for that purpose.

Finally, in Chapter 6, two use cases have been chosen to validate the usability of our approach in FI scenarios (**O9**). Concretely, the first use case is based on WSNs, where different weights are applied to modify the relevance of the parameters. A particular context where authentication mechanisms are defined based on CPRM is integrated inside the model scheme **[J3]**. The second use case focuses on 5G Green relay networks, and the aim is to validate the usability of SQT-RS in FI. We have chosen 5G Green relay networks because of

their high diversity of parameters. In this second use case, the focus is on the recommendations provided by the tool for the different contexts, based on the type of relay/device and the objective criteria selected by the user.

The conclusions of this thesis are provided in Chapter 7, where final remarks and future challenges to be addressed in the scope of Security and QoS tradeoff are discussed.

1.4. Publications and Funding

The contributions in this thesis have been published in ranked international journals, as well as in international conferences in the area of security and QoS tradeoff.

JCR Journals:

J1. A. Nieto, and J. Lopez, *Analysis and Taxonomy of Security/QoS tradeoff solutions for the Future Internet*, In Security and Communication Networks (SCN) Journal, no. 1939-0122, Wiley-Blackwell, pp.2778-2803, 2014.

J2. A. Nieto, and J. Lopez, *A Model for the Analysis of QoS and Security Tradeoff in Mobile Platforms*, In Mobile Networks and Applications (MONET) Journal, vol. 19, issue 1, Springer US, pp. 64-78, 2014.

J3. A. Nieto and J. Lopez, *Contextualizing Heterogeneous Information in Unified Communications with Security Restrictions*, In Computer Communications Journal, Accepted for publication.

International Conferences:

IC1. A. Nieto, and J. Lopez, *Security and QoS tradeoffs: towards a FI perspective*, In Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on, IEEE, pp. 745-750, 2012.

IC2. A. Nieto, and J. Lopez, *Security and QoS relationships in Mobile Platforms*, In The 4th FTRA International Conference on Computer Science and its Applications (CSA 2012), Lecture Notes in Electrical Engineering 203, Springer Netherlands, pp. 13-21, 2012.

IC3. A. Nieto, and J. Lopez, *A Context-based Parametric Relationship Model (CPRM) to Measure the Security and QoS tradeoff in Configurable Environments*, In IEEE International Conference on Communications (ICC'14), IEEE Communications Society, pp. 755-760, 2014.

IC4. A. Nieto, *Evaluation of Dynamic Instantiation in CPRM-based Systems*, In 9th International Conference on Risk and Security of Internet and Systems (CRiSIS'14), vol. 8924, Springer, pp. 52-66, 2014.

IC5. A. Nieto, and J. Lopez, *Security and QoS Tradeoff Recommendation System (SQT-RS) for Dynamic Assessing CPRM-based Systems*, In 10th ACM International Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet'14), ACM, pp. 25-32, 2014.

National (Spanish) Conferences:

NC1. A. Nieto, and J. Lopez, *Herramienta para la Compensación de Parámetros de QoS y Seguridad*, In XIII Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2014), pp. 303-308, 2014.

Other Publications

A1. A. Nieto, and J. Lopez, *Traffic Classifier for Heterogeneous and Cooperative Routing through Wireless Sensor Networks*, In Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on, IEEE, pp. 607-612, 2012.

This thesis has been funded by the Ministry of Economy and Competitiveness under the "Programa Nacional de Formación de Personal Investigador" (FPI). Some parts of the work were also carried out during the pre-doctoral visit of the first author to the Department of Information and Communication Systems Engineering, University of the Aegean, Samos (Greece).

CHAPTER 2

Classifications for Security and QoS Tradeoff

In this chapter, we identify the parameters that best describe the behaviour of the networks in Future Internet and Internet of Things scenarios. The parameters are classified in different groups according to their role in the collaboration and their relevance in the networks. Classifications are done according to the particularities of each type of network, and also according to the common points and their interoperability. The set of parameters to consider is taken from a selection of approaches focusing on security and QoS tradeoffs, and the selection is based on the dependencies of the networks or the collaboration between these parameters. The chapter concludes with a set of assumptions or recommendations about future directions for the coexistence and collaboration of security and QoS mechanisms.

2.1. The need to classify

In Chapter 1, three representative networks in the FI are chosen, and an overview of the main topics and challenges to be addressed in these representative scenarios are provided. Based on said analysis, the specific behaviour of the networks foreshadows that a high number of parameters must be analysed for the security and QoS tradeoff. However, this raises new questions that will be considered in this chapter:

- Identify the parameters used in the candidate networks to be part of the FI, and those that are used for evaluating the integration of these networks (Figure 2.1). This must be done in order to understand, overall, the dimension of the problem.
- Whether the analysis of the security and QoS tradeoff can be done using a restricted set of parameters. If the set of parameters can be restricted, the analysis can be simplified.
- If there are parameters that not only are more analysed in the field of security and QoS tradeoff, but also have an special relevance in the FI networks. If a parameter is more important than the rest in an environment, then this information should be approapiately interpreted during the security and QoS tradeoff.
- What are the basic requirements for the integration and the interoperability. This must be analysed to identify the characteristics that an analysis of the security and QoS tradeoff must consider when multiple environments are considered together.

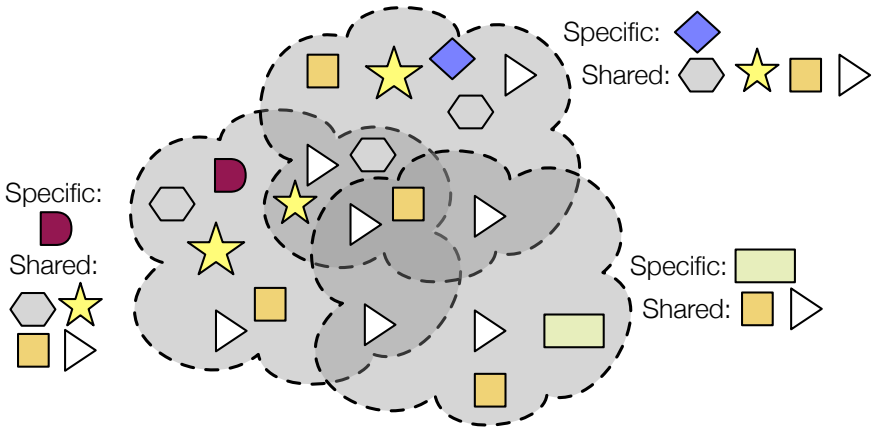


Figure 2.1: Identification of specific properties and shared/general properties.

Moreover, knowing how the candidate networks are interrelated and their similarities and differences helps us to catalogue the expected behavour of the devices in these networks. Besides, foreign devices must know the parameters must to know the parameters that are considered in the visited network (as minimum while are traversing the network), even when they aren't considered at the home environment. In this chapter, we not only provide a classification of parameters, but also provide a discussion about how security and QoS can be addressed when the integration of the heterogeneous networks will be made.

2.2. Security and QoS topics based on open challenges

In this section, we provide an analysis based on related works on Security and QoS tradeoffs. The aim is to provide an analysis based on current research tendencies and network requirements presented in the previous chapter. The analysis describes the classification of several research works from a Security and QoS tradeoff point of view, with the aim of identifying common points as well as differences based on the technology. Moreover, this classification is also essential for the identification of parametric relationships between security and QoS parameters, that will be part of further analysis. We have approached our study from three points of view. Firstly, the characteristics of each type of network are studied in order to find similarities between them (Table 2.1). Secondly, we have studied the requirements for network interconnection (Table 2.2). Thirdly, we also consider general studies related with the Security and QoS tradeoff (Table 2.3).

2.2.1. Classification based on features

Table 2.1 shows that, in the research under consideration, authentication and communication integrity are two properties included in many of the research works that address security issues. This is especially true in cellular networks, where there has been a considerable increment in the relevance of security services when compared to the other two types of networks studied. In this sense, AAA Services (AAA Serv.) have become of special interest, as can be seen in Table 2.2 .

Regarding QoS, delay is the parameter that has received more attention, followed by bandwidth and availability in the case of MANETs; in WSNs, energy consumption is the most relevant parameter because of its influence in network lifetime. In the particular case of throughput in WSNs, it is noteworthy that, although it is a parameter mentioned in several papers, it is not discussed as thoroughly as delay and energy consumption. We must also note that QoS signaling is analysed in several articles related to MANETs, where the analysis of DoS attacks is meaningful.

2.2.2. Classification based on convergence and interoperability

Table 2.2 analyses the research performed in relation with network integration. From this follows that most of the work considered includes AAA Services, performance and security problems due to handover, and QoS signaling (QoS Sig.). Therefore, these approaches reflect the importance of deploying resource reservation and security mechanisms, and ensuring that such schemes do not adversely affect to handover. Moreover, the deployment of AAA Services is necessary due to the cooperation among systems and the participation of users.

However, there are several open issues here. Probably, the most worrisome is cooperation among mobile operators, particularly because in these environments collaborative AAA Services must be deployed in order to allow user monitoring and guarantying a correct use of the network. Those services that allow QoS signaling must be deployed too. Moreover, traceability or misuse of users' data must be avoided. There is no doubt that meeting all these requirements is complex, especially if we consider several domains. The resource reservation is also complicated if there are several operators involved, and with respect to

Table 2.1: Classification based on features.

	Security								QoS									Purpose		
Paper	Authentication	Authorisation	Integrity	Trust	Encryption	Key	AAA Serv.	IPsec	Delay	Throughput	Jitter	Bandwidth	Packet Loss	Overhead	Energy	Availability	QoS Sig.	Attacks	P. Analysis	Type
[37]	-	-	-	-	x	x	-	-	-	-	-	-	-	x	x	-	-	-	-	WSN
[15]	x	x	x	-	x	-	-	-	-	-	-	-	-	-	x	x	-	x	-	
[40]	-	-	-	-	-	-	-	-	x	x	x	x	x	-	x	x	-	-	x	
[41]	x	-	x	x	x	-	-	-	-	x	-	-	x	x	x	-	-	x	-	
[18]	x	-	x	x	x	-	-	-	x	x	x	-	x	x	x	x	-	-	x	
[44]	x	-	x	-	-	x	-	-	x	x	-	x	-	x	x	x	-	x	x	
[51]	x	-	x	x	-	-	-	-	x	-	x	x	-	x	-	x	x	x	-	MANET
[52]	x	x	x	x	x	x	-	-	x	x	x	x	x	x	x	x	x	x	x	
[53]	x	-	x	-	-	-	-	-	x	-	-	-	-	-	-	x	x	x	-	
[54]	x	-	x	x	x	-	-	x	x	-	-	x	-	x	-	x	-	x	-	
[50]	x	-	x	-	-	-	-	x	x	x	x	x	-	-	-	-	-	x	-	
[8]	x	x	x	x	x	x	x	-	x	-	-	-	-	-	-	-	x	-	-	Cellular
[59]	x	-	x	-	x	-	x	-	x	x	-	-	x	x	-	-	-	x	-	
[71]	x	x	x	-	-	-	-	-	x	x	-	x	-	-	x	x	-	x	-	
[70]	x	x	x	x	x	x	-	-	x	-	-	x	-	-	-	-	-	-	-	
[58]	x	x	x	-	x	x	x	-	-	-	-	x	-	-	-	x	-	-	-	

AAA Serv., authentication, authorisation, and accounting services; QoS, quality of service; QoS Sig., QoS signaling; P., performance.

handover, there are still unresolved problems in simpler networks than those proposed in a heterogeneous paradigm, as it is the case of FI.

2.2.3. Classification based on general purposes

Table 2.3 shows that several research papers deal with the study of cryptographic mechanisms and the effect that the key length and the type of mechanism (e.g. symmetric or asymmetric) has on the communication delay, as well as on other parameters.

In particular, delay is one of the most frequently occurring parameters in the research works analysed in Table 2.3, followed by overhead, throughput and energy consumption. It should to be noted that, since most of the works analysed are based on real-time systems, the relevance of the delay parameter is rather natural. In fact, in a real-time system, the data received beyond a period of interest become of no relevance (and, usually, they are discarded). Therefore, delay is a parameter with a more negative impact in QoS than the low throughput, for example. However, throughput is interesting in the sense that, if the data arrives within the period of interest, (ideally) the maximum amount of data arrives.

In addition, energy consumption is a parameter that mainly concerns networks with few

Table 2.2: Classification based on convergence and interoperability.

	Technologies							Type					
Paper	*WSN*	*MANET*	*Cellular*	*MIP*	*MIH*	*WLAN*	*WiMAX*	*Integration*	*Attacks*	*QoS Sig.*	*AAA Serv.*	*Handover*	*Analysis*
[8]	-	-	x	x	-	-	-	x	-	x	x	-	-
[95]	-	-	x	x	-	x	-	x	-	-	x	x	-
[96]	-	x	x	-	-	-	-	x	x	x	x	-	-
[97]	x	x	x	-	-	x	-	x	-	-	-	x	x
[98]	-	-	x	x	-	x	-	x	x	x	x	x	-
[99]	-	x	x	-	-	x	-	x	-	-	x	x	-
[77]	-	-	-	-	x	-	-	-	-	x	x	x	x
[78]	-	-	x	-	x	x	x	-	-	x	x	x	x
[79]	-	-	x	x	x	x	x	x	-	x	-	x	x

AAA Serv., authentication, authorisation, and accounting services; QoS, quality of service; QoS Sig., QoS signaling; WSN, wireless sensor networks; MANET, mobile ad hoc networks; MIP, mobile internet protocol; MIH, media independent handover; WLAN, wireless local area network; WiMAX, worldwide interoperability for microwave access.

resources, and is particularly very important in sensor networks. Indeed, sensors are used in order to take periodic measurements from the environment in isolated locations, so energy consumption will determine the utility time or network lifetime.

Finally, most of the research studies here consider performance analysis (P.Analysis), and some of them have considered the difficulty of avoiding some attacks (particularly DoS attacks). There are also papers where security is explicitly identified as a parameter to protect QoS (SfQ) [20, 106], or the opposite case, that is, QoS can be seen as a security requirement (QfS) [44].

2.2.4. Considerations in the Classifications

We have seen that much of the work based on cellular networks considers the integration of such networks with another type of technology, in particular with MANETs, and proposes the use of MIP or MIH as mobility management protocols. However, there are not many works that consider the integration between WSNs and the rest of the networks, although there are several approaches that investigate the interdependencies between security and QoS (especially considering the energy factor). Regarding MANETs, we can find in the literature both types of approaches, those that consider Security and QoS tradeoff only in MANETs, and those others that consider the integration with other infrastructures, especially with cellular networks. Such a relationship is understandable because both networks can be supplemented to provide the user with a greater range of services.

Note that the information in the tables can be easily computerised and even used by web applications in order to determine the priority between parameters prior to starting the communication with a particular network. In such cases, Security and QoS parameters can be generated independently. Indeed, it is possible to merge both types of parameters,

Table 2.3: Classification based on general purposes.

	Security						QoS								Purpose	
Paper	Authentication	Authorisation	Integrity	Trust	Encryption	Key	Delay	Throughput	Jitter	Bandwidth	Packet Loss	Overhead	Energy	Availability	Attacks	P.Analysis
[20]	-	-	-	-	-	x	x	-	-	x	-	-	-	-	x	x
[102]	x	x	x	-	x	x	x	-	-	x	-	-	-	-	-	-
[103]	-	-	-	-	-	-	x	x	x	-	x	x	x	-	x	x
[104]	x	-	-	-	-	x	x	-	-	-	-	-	-	-	-	x
[105]	-	-	-	-	x	x	x	-	-	-	x	-	x	-	-	x
[106]	-	-	-	-	x	x	x	x	-	-	-	x	-	-	x	x
[107]	-	-	-	-	x	-	-	-	-	-	-	-	x	-	-	x
[7]	x	-	x	-	x	-	x	x	-	-	x	x	-	-	-	x
[108]	-	x	x	x	x	-	x	x	-	-	-	x	-	-	-	-
[109]	x	-	x	-	x	x	x	-	-	-	-	x	x	-	-	-

QoS, quality of service; P., performance.

although it is not always desirable. In most scenarios, the context should be applied in order to determine whether either Security or QoS parameters have to be prioritised. For example, under an attack, security parameters should probably be prioritised, although this can be more complex, as we shall show in what follows.

Finally, the most common trends in the study of connections between QoS and security are: (i) tests to evaluate the performance of new security solutions [110, 111], (ii) studies of the network QoS to help to detect the existence of threats [112, 113, 114], and (iii) security techniques to help to prevent the QoS degradation (SfQ) [20, 52, 106, 115]. We can also find studies where the QoS is considered as a prerequisite for the development of security applications (QfS) [44].

2.3. Observations for integration and interoperability

In previous sections, we have explored the current state of the art for different networks considering Security and QoS tradeoff and network interoperability. Moreover, we have presented a classification that shows the most important parameters in such approaches. In this section, we analyse the requirements that a system for network integration should satisfy based on previous results. Furthermore, there are some QoS and security requirements to be considered by network interoperability architectures in FI. In order for these schemes to be effective, it is extremely important to prevent possible attacks that can affect performance.

2.3.1. Network integration

Figure 2.2 shows common and specific characteristics for the three types of networks analysed.

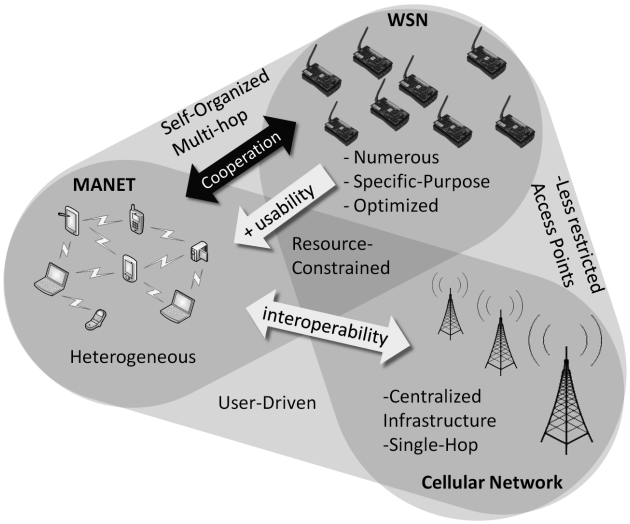


Figure 2.2: Network similarities and particularities.

The purpose of each type of network is certainly different. WSNs are formed by a large number of devices that can be replaced by others of the same type, and are specialised in obtaining measurements from the environment where they are deployed. Furthermore, sensor networks are usually specific-purpose oriented, so they can be optimised to achieve an objective efficiently (e.g. to observe events for a specific period of time with the minimum energy consumption).

On the other hand, MANETs are heterogeneous networks and, therefore, add more usability and flexibility than WSNs (although both are considered self-organising networks). However, the price to pay in MANETs because of having different types of devices is to loose the ability to optimise resources as efficiently as can be done, for instance, in WSNs. More precisely, in MANETs, we have no prior information about the types of devices that can be interconnected (e.g. terminal devices, laptops, etc.), and, as a consequence, the hardware and the communication protocols for such devices have to be more general.

Cellular networks add advantages in terms of interoperability due to their centralised infrastructure, that, as we have seen, can act as a gateway for Internet access, as well as to provide the infrastructure to deploy AAA services. Both types, MANETs and cellular networks, are user-focused, so their interoperability could be quite attractive from a commercial point of view. Sensor networks are more specific to a particular scope, but they are optimised.

Interoperability between WSN and cellular networks could provide sensors with a way to connect to the Internet, but the BTS can be far too aggressive in terms of resource consumption (e.g. consumption in data transmission) for direct use in sensor networks. Like cellular networks, WSNs have an access point with, presumably, more resources than normal devices within the network. However, even connecting these special elements to

each other could have serious consequences for QoS in WSNs, because if these devices with more resources use all their battery, they could leave critical areas of the WSN without connectivity and, therefore, useless.

Considering all this, the interrelation between WSNs and cellular networks is not quite clear at present, although the interrelation between cellular networks and MANETs has been defined better, probably motivated by users' participation in such networks. Maybe the interaction between MANETs and WSNs is more feasible, although MANETs devices should be adapted to communicate with sensor devices.

2.3.2. Basic requirements and observations for interoperability

We present here some of the possible components needed to provide QoS and security in FI. An important component for these schemes to be effective is to prevent the possible attacks that affect the performance, as well as to provide alternatives to improve the control mechanisms of the network. In addition, trust and privacy schemes are basic to ensure the adoption and survival of such architectures.

Figure 2.3 shows the main components for an interoperability scenario considering Security and QoS tradeoff within a generic node. The composition of the node will be explained in the following paragraphs. However, there is one additional consideration that we have added: the separation between static and dynamic parameters. The static parameters are those that can only be manually modified, while the dynamic parameters are dependent on the previous ones and other dynamic parameters[1]. Moreover, the context is crucial when setting the priority among parameters, and also to provide the values of some static parameters. For example, trust can be considered as a dynamic security parameter, and can be local to the node. Trust can be dynamically modified based on other security parameters which depend on the context. Furthermore, trust and security parameters should be stored in anti-tampering mechanisms in the device. On the contrary, additional security mechanisms should be deployed to periodically test the correctness of the data.

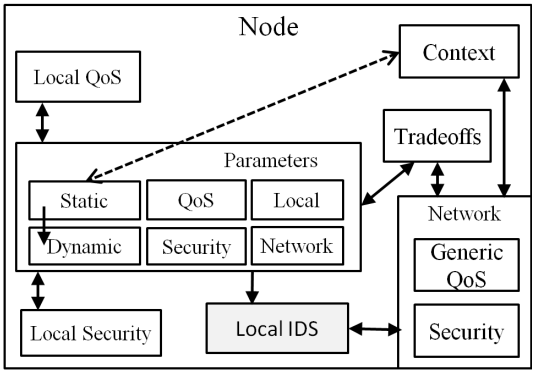


Figure 2.3: Security and quality of service (QoS) tradeoff components in a node

Note that depending on the node resources some characteristics cannot be implemented (e.g. local IDS), although this may also depend on the implementation of the solution.

[1]This can be proved using a dependency relationship diagram.

2.3.2.1. Quality of Service

In our approach, we consider different ways of understanding QoS. For example, in a WSN, QoS must be considered taking into account the lifetime of the network, and how to extend it to enable the WSN to keep working for as long as possible. Therefore, in the case of a WSN, it is possible to see the network as a single service, and if we immerse ourselves in the task we can probably determine what parameters need to be considered in order to extend the lifetime as much as possible. Likewise, other types of networks can also have their own requirements and need to keep their usefulness and continue provisioning services. We consider these QoS requirements as inherent to the network (or special QoS characteristics of the network).

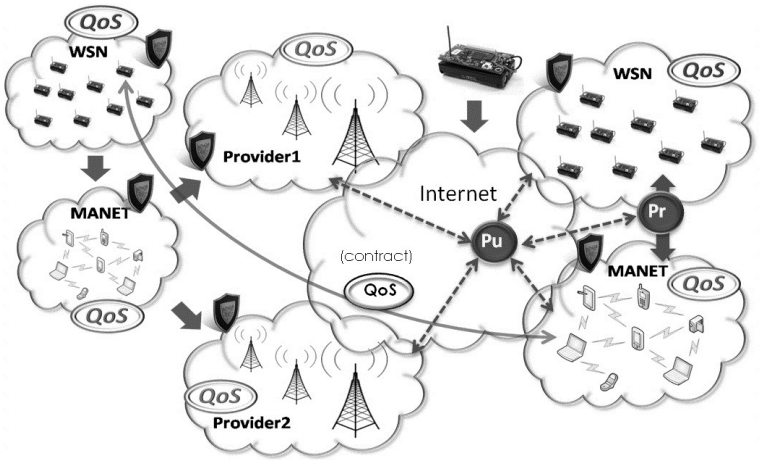


Figure 2.4: Cooperative security and quality of service (QoS) environment.

A key point of traditional QoS mechanisms for data transmission is network congestion management. Several studies have concluded that the effectiveness of such mechanisms is high in moderately congested networks, but are useless in scenarios with low congestion, and unworkable when congestion in the system is high. Therefore, after a certain threshold (which depends on the system's characteristics), a QoS mechanism may become a burden to the system instead of alleviating it. This type of QoS, more general and dedicated to data transmission, helps to ensure the efficient management of network resources, becoming more useful as the number of participants in the network increases, but also more complex to implement since it usually requires either the reservation of resources or the establishment of priority schemes. We consider these QoS requirements as general for network convergence and interoperability (or general QoS characteristics of the communication system).

We conclude that each network has its own QoS features that should be prioritised for their subsistence, and further, more general QoS characteristics for the communication. In fact, it is possible that the QoS for the communication matches the specific QoS for an environment, otherwise it is necessary to find a consensus and determine the requirements that are of higher priority based on the context to adequately orchestrate the behaviour of the system. Therefore, the policies in the node should depend on the context at a particular time, and may change dynamically as environmental conditions vary.

Figure 2.4 shows this idea. Each network has its own needs, but shares common concerns

in the transmission medium used for interoperability. Currently, this is possible using border gateways in each network. However, the difference with the new approaches is that, for total interoperability among networks, in which an element of any network can connect to a different network, the nodes need to be able to adapt to changing QoS requirements whilst respecting the QoS requirements of the network visited. The main objective should be that the node can enjoy the services that other networks provide (e.g. Internet connection, access to environmental information, etc.) but always without interfering negatively in the QoS of the system visited. The big challenge is how to do this while preventing nodes with fewer resources from being seriously damaged during interoperability. Furthermore, the adaptation of some devices could require hardware modifications, and this could be an unappealing option for manufacturers if the return on investment does not compensate.

2.3.2.2. Security

As we have seen, AAA Services play an important role in cellular networks, but may be extensible to other networks with the aim of seeking a unified security architecture. As we have already shown in Chapter 1, cellular networks can provide security to other architectures by using these services. Indeed, while QoS within each network can have its own characteristics that must be preserved, security usually has common needs, at least in the three types of networks studied. Therefore, it could be assumed that future security mechanisms will tend to be distributed and collaborative. These two features can be difficult to implement if there are different business domains involved. Service providers are cautious about sharing information with each other for several reasons. For example, there is the risk of confidential information leaks from one company to another, that could affect the sale of commercial products.

However, maybe the most damaging aspect is that the exchange of information affects users' data privacy. If this happens, it might incur individual or collective lawsuits, coupled with the possible compensation expense. This could damage the reputation of the service provider.

Figure 2.4 shows a possible security scheme for security cooperation. To avoid the unnecessary redundancy, the security mechanisms must be developed taking into account the open scheme that represents the FI, where the networks become open architectures that promote the cooperation between services. Thus, these mechanisms should be able to adapt to the environment where they are deployed, as well as to provide additional tools to allow the cooperation between different networks without affecting QoS. In addition to these local control mechanisms, it is necessary to deploy *private* (Pr) and *public* (Pu) security cooperation architectures to provide the security and trust mechanisms necessary for the exchange of sensitive information. The aim is to allow authentication of individuals while, at the same time, avoid traceability of information that could be analysed by unauthorised entities. Pr is responsible for the data exchange between service providers (SP) and other entities subject to data protection laws or other requirements. Thus, Pu uses the information provided by users to define models of trust and security mechanisms in order to enable secure cooperation among networks. The final objective is to allow both architectures to coexist and benefit each other, also increasing the collaboration between multiple paradigms.

The aim is to provide the service providers with a common infrastructure for sharing

information with other networks for security purposes and that, in turn, public networks can provide useful data to the system through the public architecture. The difficulty with this solution lies mainly in the fact that, in order to determine whether the information provided is reliable (especially in the case of Pu), it is necessary to deploy trust mechanisms on a large scale. However, currently, there are cooperation mechanisms in social networks and online forums that allow users to judge and penalise misbehaviour in the network. The improvement of these techniques and their integration into a common collaborative framework could provide great benefits for security in the FI.

2.3.2.3. Attacks that affect the performance

Providing QoS guarantees is essential to prevent those attacks affecting to performance and that can lead to DoS. Similarly, preventing DoS attacks is fundamental for maintaining QoS guarantees. Thus, if both QoS and security mechanisms can collaborate, it is not only possible to prevent the corruption of QoS mechanisms, but also to avoid some additional traffic. For example, the QoS mechanisms perform a study based primarily on parameters that indicate the network performance (e.g. throughput, delay, packet loss). This analysis is also of interest for the early detection of attacks, and to detect anomalous behaviour in networks that follow a predictable behaviour. Therefore, the IDS can work with the QoS mechanisms to obtain this information without generating additional traffic. This is shown in Figure 2.5. We cannot forget that, while in some environments the additional traffic is not a problem, in resource-constrained networks (e.g. WSNs) the repeated transmission of data can be damaging.

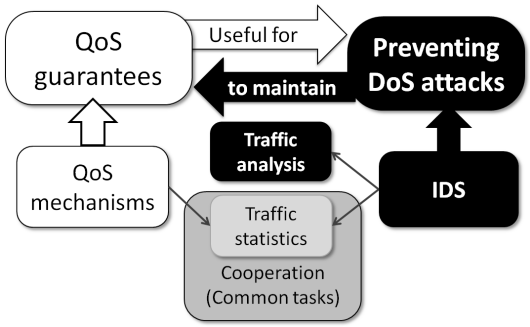


Figure 2.5: Avoiding additional traffic through cooperation.

The problem of attacks that affect the performance (e.g. signalling attacks) is that, in most cases, it is very difficult to accurately predict whether the network is under attack or, whether the network conditions are changing due to other causes, especially in dynamic networks. In the case of QoS mechanisms enabled to automatically react to changes in the network by varying the communication parameters to maintain the QoS guarantees, an attacker could force a change in the traffic conditions affecting the behaviour of the network. The IDS might not realise anything has happened because it is the QoS mechanisms and policies that manage the network traffic.

Another possibility is that QoS mechanisms do not act as traffic regulators and that, in case of sudden changes in the network, the IDS warns of this change (e.g. to a network

administrator). There are some mechanisms that can enable the IDS to isolate a part or the whole network in case of an attack. However, the isolation affects the availability and, for this reason, automatic and reactive responses are not a good option in environments that depend heavily on this parameter. In fact, detecting and preventing the attacks that affect the performance is a complex task that remains open to multiple interpretations. For example, in contrast to those systems where availability is a key parameter, in other environments the priority is to prevent data leaks; hence, in these systems the network isolation is an option which is more than acceptable.

Moreover, the attacks that affect performance are a big problem for network integration since the effects of such attacks can be propagated throughout the whole collaborative structure. Indeed, an attack may affect any of the parameters that influence to more other parameters in the environment. Likewise, the attack can spread through other networks that are connected to the infected network, producing an extremely undesirable chain reaction. However, an advantage of collaboration among networks is that, if a network providing a service has to be isolated, it is feasible to find another network to replace it in a short period of time. Nevertheless, a disadvantage is that abuse might be possible (e.g. an attacker isolates a network to force the use of another network) if the security architecture is not sufficiently robust and the QoS mechanisms of the networks are not able to prevent a total network collapse.

Finally, it is useful to consider the attacks that affect the performance from two viewpoints: local and network. Figure 2.3 shows a local IDS at the node. This solution increases the complexity of the node, but also provides local security, preventing the node from being misled or corrupted. The local IDS could identify when the changes in local parameters and requirements are related to each other, and raise an alert or react against the posibility of an attack.

However, it is not possible to assume that all the devices in the network have a local IDS. In such cases, the IDS can be implemented by another device in the network (network IDS). The network IDS examines the network traffic and determines whether there is a threat. Intuitively, if the network IDS can collaborate with a local IDS, it would be possible to reduce the data to be sent from the node to an external IDS (e.g. the node can send the result of computing its security state to the IDS).

Nevertheless, the way in which the local IDS is implemented in the node is fundamental to guarantee that it will work correctly. If the local IDS can be corrupted by the node, then the solution becomes unusable. There are anti-tampering solutions that enable storing certain data (e.g. keys, certificates) within a protected device in a node (e.g. TPM, NFC). But the problem of computing the security state while maintaining the trustworthiness of the application for it continues to be an open challenge.

2.3.2.4. Trustworthiness

Keeping trustworthiness is necessary to ensure that the restrictions imposed by the QoS application requirements are satisfied. It should be a primary objective, and therefore the control traffic dedicated to promoting or protecting this should be prioritised. Two possible implementations for introducing trust as a parameter by using the priority are based on *data stream* (DS) and *node* (IN):

- Priority based on DS. The priority of the traffic could change based on the trust level for the data stream. A data stream is a sequence of datagrams that follow the same path. Hence, the priority here would be based on the trust level of the nodes of the path (more trust implies more priority). The problem is the calculation of the priority and that, during the transmission, this priority could vary. This would then entail a recalculation of priority which would complicate the architecture.
- Priority based on the IN. The idea is to assign an individual priority to the node. Thus, the source node marks the datagram with its trust level, that is used as the datagram priority. The rest of the nodes in the path should transmit the information based on this priority level.

Considering the two alternatives, the IN would be easier to implement, since the data could be sent without prior calculation of a route, and therefore does not involve additional costs for maintenance. Since priority-based transmission using the trust level as metric is too strong for data transmission (trustworthy nodes could cause overhead), this possibility could be used when the control nodes attempt to transmit high priority information about the network state (e.g. IDS nodes).

2.3.2.5. Privacy

Future communication mechanisms for heterogeneous networks have to consider privacy as a primary requirement, taking into account the role that the user plays in networks, as we consider in our approach. So, new concepts such as *Privacy by Design* (PbD) have to be taken into account in order to look for consistency between the mechanisms developed to protect user privacy in different networks. Furthermore, future security mechanisms have to be able to avoid the traceability of users throughout the entire network. In fact, the traceability of users is directly related to privacy because it provides data to the attacker that can be analysed with the intention of discovering behaviour patterns of the user. Moreover, the user is usually exposed to these kinds of practices due to the use of services that require the acceptance of terms of service related to privacy, that are often not understood, but despite this, are signed by the user.

2.4. Discussion

New paradigms, and among them, FI and IoT, propose the interconnection of heterogeneous networks on a large scale. However, there are several issues regarding QoS and security mechanisms that need to be addressed first. In this chapter, we have presented an analysis of the current state of technology in network integration, focusing especially on the study of security and QoS issues. In order to achieve this we have focused our analysis on the the three representative networks selected in the Chapter 1. These are Cellular Networks, MANETs and WSNs.

Regarding the questions raised at the begining of this chapter, to identify the parameters used in the representative networks, we have shown different classifications to identify similarities among such technologies, and also to identify the requirements for network interconnection. Consequently, important dependencies between Security and QoS requirements

and parameters have been identified. We have also proposed high-level integration architectures for those networks in the FI scenario.

After our study, we also can deduce that the set of parameters cannot be restricted: technological improvements and other characteristics must be considered. If a static set of parameters is selected, the possibility of extending the analysis whenever necessary would be severely limited. The ideal case should be to provide a tool with a set of parameters than could be incremented when is required, but also be able for working with a reduced set of parameters without loss of generality when is required.

Furthermore, not all the parameters has the same relevance in all networks. For example, as we have seen, when there are mobile nodes, or isolated nodes, the parameter *Energy* is more relevant than, for example, *Data Rate* in many cases. Moreover, as a consequence of keeping the user satisfied, the step towards allowing the use of QoS mechanisms through the Internet seems even closer. In such cases, connectivity becomes a very important factor. Indeed, the subjective perception of the user is very important in the analysis of the security and QoS tradeoff. The solutions should be efficient and secure, but also useful to the user. If the security and QoS tradeoff is performed without considering the user's requirements, then the usability of the final architecture will not be adequate.

Based on our research, we conclude that there are important security and QoS problems that must be solved before full integration becomes a reality. Such problems must be solved prior to any integration because a fault in one system could spread through the network. Moreover, it seems that the security and QoS parameters are not directly related in a point-to-point relationship. Figure 2.5 not only shows an example of the symbiosis between security and QoS, but also that there are common aspects that can affect to both security and QoS. So, different type of parameters must to be considered (not only security and QoS parameters) to identify the indirect relationships.

Moreover, as is discussed in Chapter 1 (Section 1.2.3.3), when a high number of parameters must be analysed, it is very useful to work with partial information to reduce the complexity of the analysis, and combine the information from independent sources when is required. Along this chapter, diverse approaches have been analysed to provide different classifications, however, the results provided in said approaches are specific and therefore they are isolated; combining two or more results from these papers is not trivial at all, and depends on the context.

Furthermore, the development of new security and QoS mechanisms, designed to allow interoperability between different networks should be taken in parallel, without forgetting the current developments in different related technological areas. Abstract models can help to integrate diverse information of different technological areas without loss of generality. However, the models to provide this type of behaviour are not defined yet.

To reach effective convergence and interoperability, other protocols beyond traditional or established ones (e.g. IP) should be considered. In fact, although IP is a widely used protocol and is designed to be resistant against natural disasters, there is no truly effective QoS mechanism working over IP. Indeed, the analysis of the security and QoS shouldn't be restricted to specific protocols.

Finally, the assessment of the security and QoS tradeoff is fundamental in preventing poor behaviours in future deployments. However, as future networks are dynamic and heterogeneous, addressing these types of analyses require the systems to be defined based on

their characteristics. Indeed, the best idea is to define the system based on abstract, generic, characteristics and complete the definition based on the specific characteristics of the final environment. A parametric-based approach suits this definition very well.

2.5. Parametric Approach

Although extensive literature exists about security and QoS challenges in IoT and FI environments, the work is principally focused on solving specific problems or showing an overview of concepts within a particular area. This helps to simplify the problem so that it can be solved locally, without considering the whole environment. However, it gives us only a piece of the puzzle to be solved. Considering that there are not alternative solutions to provide an unified analysis of the security and QoS tradeoff, we implement a parametric-based approach because:

- A parametric-based approach enables the description of general/abstract concepts. So, to combine information from different sources is possible. In general, we consider that a parameter is *anything* that can be analysed. We have seen as in different sources, the word *parameter* is used indistinctly - independent on the type or the layer.
- A parameter typically is related with other parameters. Although a parameter can be defined without dependencies with other parameters, this is not usual. When a parameter is considered in an analysis of the security and QoS tradeoff, is because the parameter depends or affect to others.
- A parameter has a value. The parameters can be compared with other parameters because they have a value. The need to compare is a requirement – by nature – in a security and QoS tradeoff system.
- A parameter has a type and belongs to a layer. Although a parameter is anything that can be analysed, is considered of a specific type – depending on the context (e.g., performance, security) – and given at a specific layer – depending on the context (e.g., application, environment).

In Chapters 1 and 2 diverse types of parameters are shown, but these are not all. Depending on the context, different parameters can be required. We consider that, to perform the analysis of the security and QoS tradeoff, is unuseful if we focus the problem in only a set of parameters. What the state of the art suggest is that we need a general model that does not exist. Precisely, we address this problem in the following chapters, defining our own parametric-based model, considering all the requirements that we understand – based on the previous analysis – are needed to the analysis of the security and QoS tradeoff, and opening the door to extend this model to others areas, when possible.

It must be highlighted that, for assessing security and QoS tradeoff in future networks, it is necessary to handle the different parameters at abstract layer, in order to consider the alternatives that the different technologies may provide and support. So, we understand that a parametric-based approach gives us the possibility for combining heterogeneous parameters, including features and characteristics provided by the systems in convergence. Therefore, an additional requirement for analysing the security and QoS tradeoff is the classification of

the parameters in layers and types. This is because in some scenarios we need to assess the effect that a set of parameters has on the system or on a portion of the parameters of the system (e.g., the effect that a set of parameters of type *security* has on a set of parameters of type *performance*).

2.5.1. Classification based on Layers and Types

In order to analyse the parameters together (giving us more information about the system) while reducing the complexity and, therefore, increase the usefulness of this research, we consider that the parameters should be classified in different layers, called *Action Layers*. In particular, we decompose them into five main action layers, according to the location of the parameters in a given dynamic and heterogeneous scenario: High-Level Requirements, Local Properties, Communication, Measurements and Environment.

This decomposition of layers is proved to be useful in our use cases (Chapter 6), and for this reason, we provide this classification here. However, our approach allows the modification of the list of layers and types (Chapter 3) when is required. This is because we defend that the solutions should not provide fixed definitions in heterogeneous environments, as is the case of FI environments.

We follow a cross-layer perspective, where it is assumed that any parameter at any layer may be related to any other parameter at any other layer. Therefore, the difference is per type of parameter rather than per functionality. For example, if the *Response Time* is taken as one parameter at the Measurement layer, then it will be also related with the High-Level Requirements layer.

Intuitively, while more parameters are considered at each layer, less cross-layer dependability may occur because the cross-layer parameters set could be limited better. For example, if certain authentication mechanisms were introduced into the model, then relating the response time to the authentication requirement based on these mechanisms may be possible. Thus, the model could become richer and more specific. However, it would also be more complex. So, we have decided to maintain some direct dependencies, given as cross-layer dependencies, among different layers to simplify some relationships.

Furthermore, the parameters are classified according to their type, independently of the layer to which they belong. This allows, for example, the evaluation of the effect on a group of parameters of a type (e.g. performance parameters) when other groups of parameters (e.g. security parameters) are modified. Although both layers and types can be modified, we have considered the following types in our classification:

- Resource. Parameters which express a property of the component, typically at local properties layer, for example the processing capabilities or the memory, storage, etc.
- Security. Properties, mechanisms or other characteristics that are designed to provide security or quantify it. We use this type to evaluate the effect of all the security parameters on performance, regardless of the layer.
- QoE. Parameters related to the quality of experience. Within this type we have considered the users' experience and QoS classes in some analyses. This type is useful in those environments where the user is involved, and MOS are applicable. Sometimes, Performance or QoS types are used insetead.

- Performance/QoS. QoS parameters or parameters related with the performance of the system.
- Characteristic. Parameters which express specific properties that are very particular for some systems. For example, anti-tampering chips could be classified as security mechanisms, but are set up as characteristics because these are very specific. Not all systems can provide this type of solutions.
- Attack. For identification of specific attacks against the user, the infrastructure or part of it.
- Threat. For identifying those *things* that can be catalogued as a threat against the user, the infrastructure or a part of it. This type is more general and abstract than Attack. Depending on the system, some parameters can be considered as threats, without being attacks.
- Consequence. The parameters of this type identify something that occurs because of the effect of other parameters. This is used for parameters that identify scenarios that are undesirable in a system, or that do not occur if all is working well. For example, the overhead.

In the following, the different layers that we have been considered are explained. This is our own classification, but our approach is independent of the layers and types considered, and still works if these are modified.

2.5.1.1. High-Level Requirements

The first layer takes into account *High-Level Requirements* (HLR), that is, concepts normally understood by users or software developers. Security requirements and QoS traffic types are defined at this layer. Moreover, at this level, QoE requirements should be considered in order to evaluate the impact of requirements on users' experience/satisfaction. Moreover, users' experience is a very subjective parameter because it depends their personal experiences.

2.5.1.2. Local Properties

Local properties are directed to define the composition of the device. So, hardware improvements such as multiple antennas, NFC or any additional device can be considered as part of it. However, keeping a certain simplicity, those elements used to interact with the network are set in the communication layer. Thus, local properties here are closer to the independent-network properties, and closely related to the definition of the device itself. In this layer, the parameters define the device with and without connection capabilities.

For example, power consumption is extremely dependent on network transmission. However, it is also affected even when not connected to any network. The same is true for the allowable memory parameter.

The parameters in this layer depend on the implementation of the mechanisms and applications deployed and on the equipment built on the device by the manufacturer. Some examples are given in the following paragraphs.

2.5.1.3. Communication

The set of parameters that are considered as part of this layer are related to the communication of the node with the network. For example, the data rate parameter defines the number of bytes sent per unit of time (bit rate). Therefore, this parameter is closely related to data transmission and packet size, because the less data to be sent, the less time the system employs to sending the data. As a consequence, data transmission is also considered in this layer. It is supposed that security mechanisms have been applied in previous layers, thus, in this layer, encrypted data may be received to be sent, but there are no security mechanisms to manage the data. This layer is composed by:

- Resources and measurements of the local interfaces. For example: packet size or signal strength.
- Characteristics and elements that can be used at this layer. For example, the time that the interface is on (the required time that it has to be listening) or inactive (out of service or sleeping).
- Attacks related to the interface. These attacks can be considered in the environment layer in the case they are considered to be threats only present in the environment. The communication layer shows that the wireless capabilities provide the opportunity for these threats.
- Consequences due to poor quality in the communication. For example, the retransmission of data.

2.5.1.4. Measurements

This layer is used to take performance measurements related to the network. In this layer, typical parameters to measure the network performance are used. Specifically, this layer integrates most of the parameters that may be described based on their mathematical formulation.

2.5.1.5. Environment

Finally, in the environment layer, the parameters related to network conditions and characteristics of the environment are considered. For example, the noise in the environment, the number of devices or the probability of eavesdropping can be considered here.

In fact, the environment defines the scenario and the current context where the device is. So, it is precisely this layer that is expected to change the most throughout the lifetime of the device.

Similarly, the environment is affected in some way by the presence of a mobile device. Thus, the objective of defining this layer is twofold. On the one hand, it is useful from a context viewpoint. If the environment changes, the performance and therefore the value of the QoS parameters may vary in the whole device. On the other hand, from the point of view of service providers and network administrators, it is important to measure the impact that one device has on the parameters of the environment. This layer also considers consequences related to network conditions.

2.5.2. Knowledge: Extracting Parameters and Relationships

At this point, it is important to note that our approach is a knowleadge-based system. We use what we know at any given time – contextual information – to analyse the security and QoS tradeoff. However, how we know what we know is an important issue.

In our case, we consider two potential sources of information, all based in the state of the art or the observation:

(A). Formulation-based relationships. This is the most defendable and robust, we know how two parameters are related because we deduce this information of the formulations and this information is integrated into the model.
(B). Literature-based relationships. This is the most subjective, and can be based on the observation. We identify relationships in user-based language, and translate this interpretation to the model.

Figure 2.6 shows an example of both cases. Our model, presented in Chapter 3, define how to express the relationships using a language and the set of mathematical formulation required to interpret the relationships authomatically.

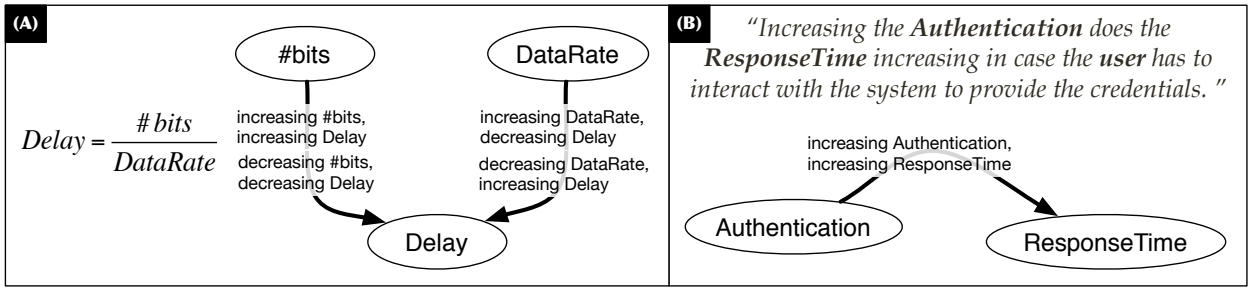


Figure 2.6: (A)–Formulation-based relationships, (B)–Literature-based relationships.

We consider that the set of parameters chosen will be dependent on the context. Some examples of set of parameters are provided in Chapter 3 (Section 3.1.3), and in the use cases discussed in Chapter 6 (Sections 6.2 and 6.3). Moreover, the complete list of parameters and their relationships considered in the use cases are detailed in Appendix A. The representation chosen is based on our own solutions to describe the parameters and relationships, before the security and QoS tradeoff analysis. Said solutions will be introduced in the next chapter, before the first example.

CHAPTER 3

Parametric Model and Context-based Behaviour

In the previous chapters, the parameters to be considered in Future Internet scenarios have been detailed. However, these have to be properly handled to extract information to be used for to assess Security and QoS, and the lack of abstraction in existing mechanisms, suggesting the development of new mechanisms to accomplish our objectives. So, in what follows, the definition of the Parametric Relationship Model (PRM) and how it can be extended to enable the integration of different contexts, dynamically, are discussed. This model is the Context-based Parametric Relationship Model (CPRM), which integrates the PRM and the different schemes to be integrated to define the heterogeneous environments based on parameters and relationships.

3.1. Parametric Relationship Model (PRM)

In the Chapter 2 we concluded that the analysis of the security and QoS tradeoff should be done considering heterogeneous parameters, and that this analysis may vary dynamically. As far as we know, the requirements that we have considered are not provided by the current tools, which are focused on specific problems that target specific layers (e.g., service layer).

In this chapter we present the *Parametric Relationship Model* (PRM) which defines the dependencies between parameters in heterogeneous environments. The PRM is defined in order to take into account the influence of one parameter on the rest of parameters considered. For this purpose, PRM-based systems consider the following elements:

- A set of parameters, which depends on the system under consideration.
- Relationships between the parameters, defined based on Table 3.1.

Moreover, this model is one of the basic inputs to get complex models as the *Context-based Parametric Relationship Model* (CPRM) (Section 3.2). Figure 3.1 shows the evolution of the models that are considered in a CPRM-based environment. The different components will be detailed in this chapter. The basic idea behind a CPRM is that models are progressively enriched as more information is known about the system. Moreover, the use of our model allow the user to work with a reduced number of parameters that have a predefined behaviour.

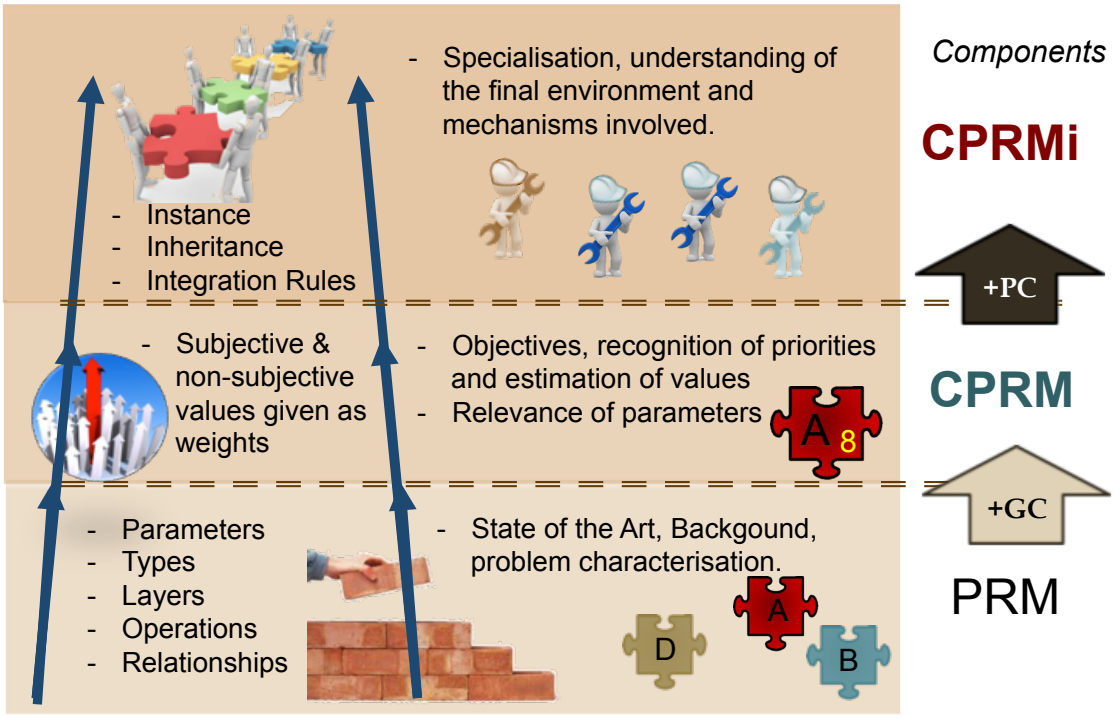


Figure 3.1: Evolution of models.

In this section we focus on the basic concepts to define the first of the models, the PRM.

3.1.1. Mathematical Definition

The possible relationships between the parameters a and b are defined in Table 3.1. It is composed by the *Basic Formulation Set* (BFS,3.1-3.4) and the *Complex Formulation Set* (CFS, 3.5-3.9).

Table 3.1: Parametric Relationship Model (PRM)

Basic Formulation Set	
$D^{+} :: aD^{+}b \Rightarrow (\Delta a \rightarrow \Delta b)$	(3.1)
$D^{-} :: aD^{-}b \Rightarrow (\Delta a \rightarrow \nabla b)$	(3.2)
$D^{\neg+} :: aD^{\neg+}b \Rightarrow (\nabla a \rightarrow \nabla b)$	(3.3)
$D^{\neg-} :: aD^{\neg-}b \Rightarrow (\nabla a \rightarrow \Delta b)$	(3.4)
Complex Formulation Set (based on (3.1)-(3.4))	
$D^{c} :: (\Delta a \rightarrow \Delta b) \wedge (\nabla a \rightarrow \nabla b) \equiv aD^{+}b \wedge aD^{\neg+}b$	(3.5)
$D^{t} :: aD^{c}b \wedge bD^{c}a$	(3.6)
$D^{\neg c} :: (\Delta a \rightarrow \nabla b) \wedge (\nabla a \rightarrow \Delta b) \equiv aD^{-}b \wedge aD^{\neg-}b$	(3.7)
$D^{i+} :: (\Delta a \rightarrow \Delta b) \wedge (\nabla a \rightarrow \Delta b) \equiv aD^{+}b \wedge aD^{\neg-}b$	(3.8)
$D^{i-} :: (\Delta a \rightarrow \nabla b) \wedge (\nabla a \rightarrow \nabla b) \equiv aD^{-}b \wedge aD^{\neg+}b$	(3.9)

The BFS (3.1-3.4) is defined in order to get a basic set of equations from which any relationship can be derived. Responsible for observing the behaviour of the system when the parameters increase or decreasse (or when the requirements are provided or not), the BFS is composed of the following relationships:

- Positive $(D^{+}, 1)$. The increment of the first parameter also causes an increment of the second parameter.
- Negative $(D^{-}, 2)$. The increment of the first parameter causes the decrement of the second parameter
- Inverse positive $(D^{\neg+}, 3)$.The decrement of the first parameter causes the decrement of the second parameter.
- Inverse negative $(D^{\neg-}, 4)$.The decrement of the first parameter causes the increment of the second parameter

CFS (3.5-3.9) is defined in order to simplify the dependency relationships diagrams based on the PRM. The equations in CFS are defined as follows:

- Complete $(D^{c}, 5)$. The action (increase/decrease) to be applied on the second parameter is the same as the first parameter.
- Total $(D^{t}, 6)$. Both parameters are related each other with a complete relationship.
- Inverse complete $(D^{\neg c}, 7)$. The action (increase/ decrease) to be applied to the second parameter is the opposite to that in the first parameter.
- Independent positive $(D^{i+}, 8)$. If the first parameter changes, then the second paramaeter will always increase, regardless the type of value change in the first parameter.
- Independent negative $(D^{i-}, 9)$. If the first parameter changes, then the second paramaeter will always decrease, regardless the type of value change in the first parameter.

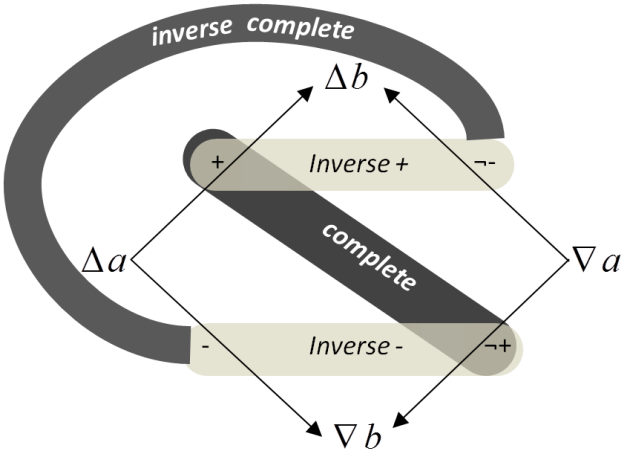


Figure 3.2: PRM relationships.

Figure 3.2 shows the relationship between the definitions in Table 3.1. Note that the *total* relationship occurs when the complete relationship between a and b is symmetric.

For example, Figure 3.3 shows the dependencies between performance parameters (ingot, e.g. availability), security properties (square, e.g. integrity), characteristics of the environment (oval, e.g. to be a real-time system) and some consequences (hexahedron, e.g. collisions).

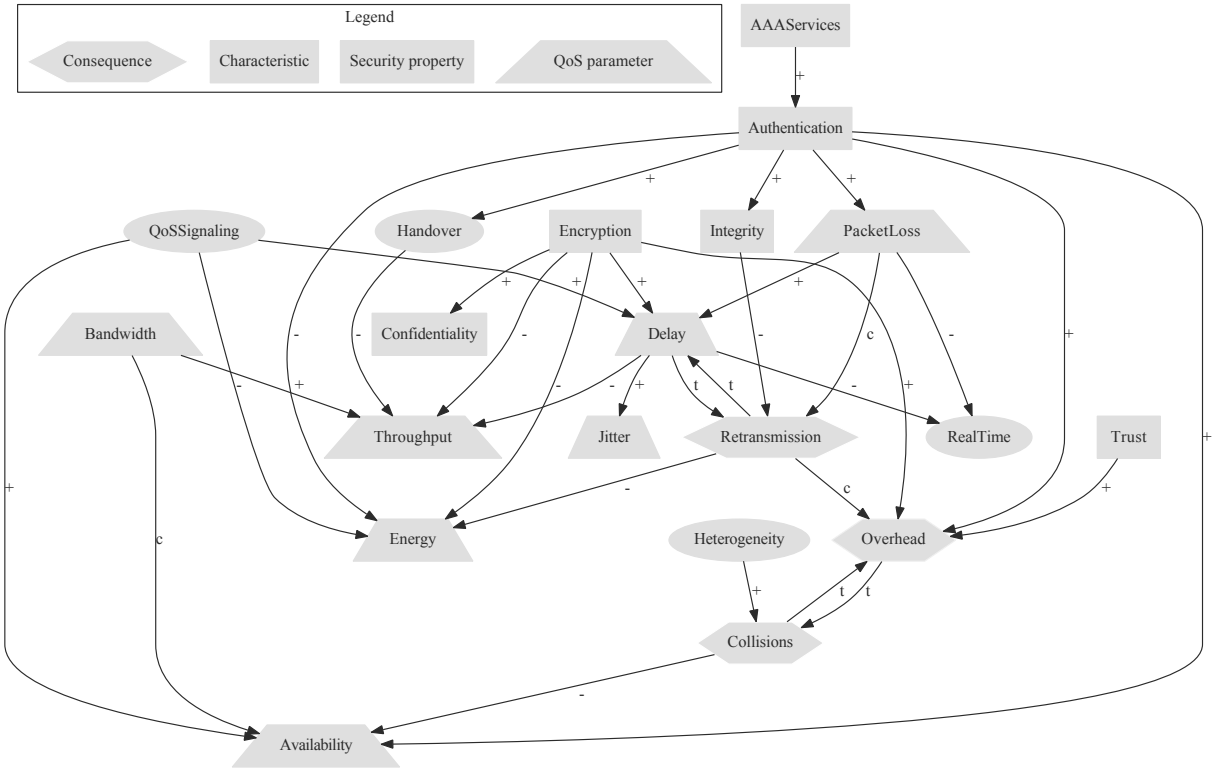


Figure 3.3: Parametric dependencies.

It is important to note that Figure 3.3 is a simplified map that does not cover all the

Table 3.2: Dependencies table.

$X \xrightarrow{D^k} Y$	Availability	Bandwidth	Delay	Energy	Handover time	Jitter	Packet Loss	Throughput	AAA S.	Authentication	Confidentiality	Encryption	Integrity	Trust	Heterogenity	Real-Time	QoS S.	Collisions	Overhead	Retransmission
Bandwith	c							+												
Delay						+		-								-				t
Packet Loss			+													-				c
AAA S.										+										
Authentication	+			-	+		+						+						+	
Encryption			+	-					-		+								+	
Integrity																				-
Trust																			+	
Handover								-												
Heterogeneity																		+		
QoS S.	+		+	-																
Collisions	-																		t	
Overhead																		t		
Retransmission			t	-															c	

possible parameters, properties and features that we can find in each different network. To cover all these possibilities the resultant schema would have to be even more complex. This gives us an idea of the difficulty of developing Security and QoS tradeoff mechanisms in heterogeneous systems, and maybe what is more important, the quite plausible risk of making a decision that affects several parameters and properties due to dependencies. This is particularly damaging in critical environments where different mechanisms that affect such parameters have to coexist.

3.1.1.0.1. Parametric Tables and Composition The information in Figure 3.3 can also be represented as a table, as Table 3.2 shows. Note that we consider that the heterogeneous nature of the environment depends on the context, such as the required *Bandwidth*, *Encryption* method used or the need for using QoS Signaling mechanisms, which depend on the application running in the node. Moreover, *Trust* can change over the node's lifetime. However, this sometimes depends on several factors related with the context where the node is (e.g. contact with malicious nodes).

Moreover, although we have considered a restricted set of parameters, it is possible to build complex dependency diagrams using the DOT language. For example, we used Graphviz[1] to build Figure 3.3.

DOT documents are easily implemented and modified. So, building the dependencies table from the DOT document is not complex and can be easily automated. For example, we used MATLAB to implement the model proposed using DOT documents. The result using the set of parameters shown in Figure 3.3 is shown in Figure 3.4. So, Figure 3.4 shows the information given in Table 3.2 once it is in the node to be used. Each square in the figure represents a $X \xrightarrow{D^k} Y$ relationship.

As a consequence, the proposed model makes it possible to measure the influence of a parameter on the rest of the parameters. For example, Figure 3.5(a) shows how the

[1]Graph Visualization Software, http://www.graphviz.org/.

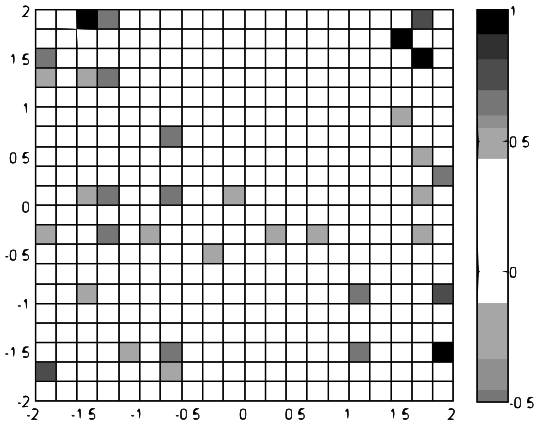
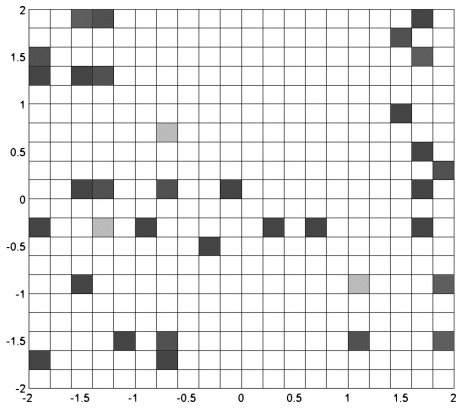
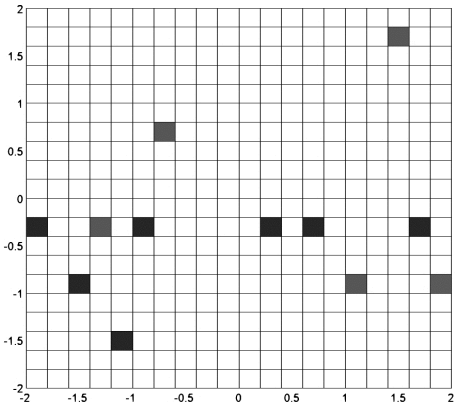


Figure 3.4: Parametric table.



(a) $PT \cup Authentication \xrightarrow{D^k+} Y$



(b) $Authentication \xrightarrow{D^k+} Y$

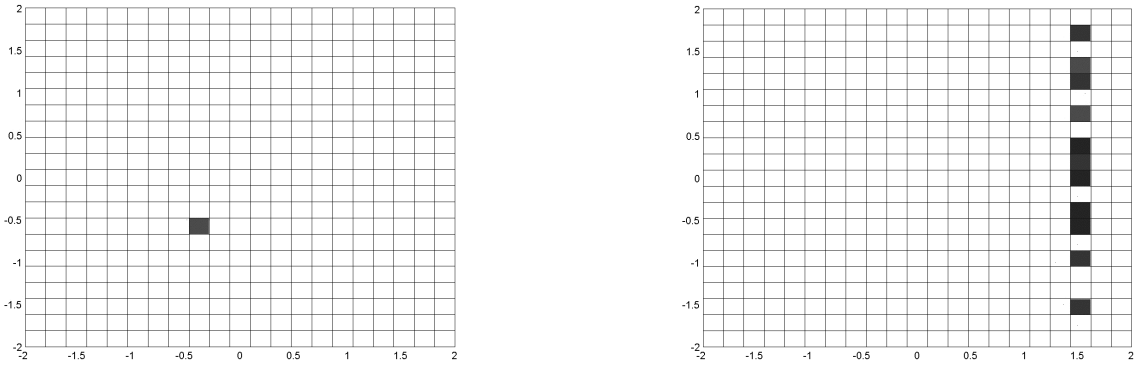
Figure 3.5: Influence of X, $X \xrightarrow{D^k+} Y$

parametric table in Figure 3.4 can change if just one parameter (Authentication in the example) is modified. Intuitively, when one or more parameters change their value, the global changes can be seen in the same table.

Moreover, Figure 3.5(b) shows only the parameters that change their values when the parameter *Authentication* changes considering the transitivity property: D^k applied once or more (+), or the same as $Authentication \xrightarrow{D^k+} Y$. In addition, the whole set of parameters affected by the modification of a set of parameters can be calculated by the union of multiple tables.

Figure 3.6 shows the opposite of the relationship shown in Figure 3.5(b). For example, the parameter *Authentication* only depends on the AAA signaling parameter (in our example), which is reflected in both Figures 3.3 and 3.6(a). Whereas, if we consider transitivity, the parameter *Overhead* depends on eleven parameters: Collisions, Heterogeneity, Retransmission, Trust, Delay, Packet Loss, Integrity, Authentication, Signaling, Encryption and QoS Signaling.

It is important to note that while in Figure 3.5(b) it is possible to see what parameters



(a) $X \xrightarrow{D^k+} Authentication$ (b) $X \xrightarrow{D^k+} Overhead$

Figure 3.6: Influence on Y, $X \xrightarrow{D^k+} Y$

are directly dependent on *Authentication*, in Figure 3.6(b) is not possible to see the chain of parameters on which *Overhead* depends. This is not a problem, if we consider that by combining both types of tables we can find out.

Finally, in this example, the parameters Availability, Energy, Jitter, Throughput, Confidentiality and Real-Time don't affect the others. This can vary depending on the type of problem and parameters being considered. For example, if we consider the number of nodes in a sensor network, then Energy is crucial and affects other parameters. In a sensor network, if the nodes die then the communication can be interrupted. Moreover, if we consider the configuration of computers to deliver Real-Time traffic, then the Real-Time characteristic can affect other parameters such as, for example, the Bandwidth, because the computer and the network have to be adapted to deliver this type of traffic.

So, general approaches can be built taking into account a common language to implement dependencies, but there are particularities which depend on the knowledge that we have of the system and the environment.

Furthermore, one additional problem to solve here could be the storage of the dependencies in the node. It seems to be fairly intuitive that those parameters which depend on the communication with external networks belonging to different domains have to be considered too, and it could be very tiresome to store various dependency diagrams in the node, one for each type of network. There are some solutions that can be included at this point:

- The node knows *two diagrams* minimum. The first one is the parametric relationship diagram related with its native network, while the second one is the parametric relationship diagram related to the communication with heterogeneous networks.
- The node only knows the parametric relationship diagram related with its *native network* (NND). In that case, there is an *intermediary node* (IN) which knows the communication diagram. The IN adjusts the local parameters at both ends of the communication.
- The node only knows the *generic communication diagram* (GCD). In that case, the node has the basic tools to communicate in a generic environment and respects a set of parameters. However, it is not possible to optimise its behaviour.

Table 3.3: Deployment of parametric relationship solutions.

Solution	Advantages	Disadvantages
NND+GCD	Authonomy	Storage space
NND	Domain-based optimisation	Dependency from IN
GCD	Heterogeneous communication	Not domain-based optimisation

Table 3.3 shows the advantages and disadvantages of each solution. In particular, *domain-based optimisation* (DbO) is very interesting from the point of view of resource-constrained networks. For example, if a powerful node wants to use a WSN, and it uses DbO, then it could adjust its parameters in order to become compatible with the foreign network.

Note that network policies (within the context) also determine the behaviour in these cases. For example, in some networks it may be fundamental to preserve the policy " The parameters in a foreign device have to be adapted to the network properties", while in others the policy "The parameters in a foreign device have to be preserved when it is resource-constrained in contrast with local devices".

In the first case, it prevents powerless devices' malicious behaviour within a powerful network (e.g. the resource-constrained node opens a communication with a powerful node to modify its behaviour to make it less productive). In the second case, it helps to incentivise the communication between resource-constrained devices and powerful devices.

Finally, the possibilities for implementation are extremely extensive and depend on several factors. In general, the deployment of the parametric relationship solutions depends on the context where the solution is deployed and the level of autonomy required at node level.

3.1.2. PRM-based Relationships between Parameters

For example, given Exp. (3.10)-(3.13), we extract the dependencies in Exp. (3.14)-(3.20) shown in Table 3.4. The difference between bit rate and data rate is basically the quantification, respectively, bits per second (bps) and bytes per second (kB/s). So, in the following we use Data Rate.

$$Delay = \#bits/DataRate \tag{3.10}$$
$$Jitter = |DelayTo - DelayT1| \tag{3.11}$$
$$Throughput(peruser) = DataRate/\#Users \tag{3.12}$$
$$TransmissionTime = PacketSize/BitRate \tag{3.13}$$

Delay, jitter and throughput are parameters considered at low level. These parameters are related with the performance in communication networks, but there are other parameters working at different levels which could be considered, depending on the system and the tradeoffs to be analysed.

Table 3.4: Using the defined model to derive relationships

Example using Delay (3.10)
$(\Delta DataRate \rightarrow \nabla Delay) \wedge (\nabla DataRate \rightarrow \Delta Delay) \equiv DataRateD^{\neg c}Delay$ (3.14)
$(\Delta \#bits \rightarrow \Delta Delay) \wedge (\nabla \#bits \rightarrow \nabla Delay) \equiv PacketSizeD^{c}Delay$ (3.15)
Example using Jitter (3.11)
$(\Delta Delay \rightarrow \Delta Jitter) \wedge (\nabla Delay \rightarrow \Delta Jitter) \equiv DelayD^{i+}Jitter$ (3.16)
Example using Throughput (3.12)
$(\Delta DataRate \rightarrow \Delta Throughput) \wedge (\nabla DataRate \rightarrow \nabla Throughput \equiv DataRateD^{c}Throughput$ (3.17)
$(\Delta \#Users \rightarrow \nabla Throughput) \wedge (\nabla \#Users \rightarrow \Delta Throughput) \equiv \#UsersD^{\neg c}Throughput$ (3.18)
Example using Transmission Time (3.13)
$(\Delta BitRate \rightarrow \nabla TransmissionTime) \wedge (\nabla BitRate \rightarrow \Delta TransmissionTime) \equiv BitRateD^{\neg c}TransmissionTime$ (3.19)
$(\Delta PacketSize \rightarrow \Delta TransmissionTime) \wedge (\nabla PacketSize \rightarrow \nabla TransmissionTime) \equiv PacketSizeD^{c}TransmissionTime$ (3.20)

3.1.3. Mobile System based on PRM

The decomposition in different levels not only allows us to properly define the relationships between parameters, the requirements and characteristics within the same level but also between different levels in a simplified way. The system becomes really complex when putting all the parameters together. Thus, in order to minimise the computation time, we decompose the problem into one layer-based problem, as Table 3.5 shows. Table 3.5 also shows the parameters that are considered at each level.

Throughout the analysis, seven types of parameters are considered:

- Traffic classes and Performance parameters. Traffic classes are defined based on the UMTS SLA traffic classification [9]. These parameters are directly related to performance parameters at low level, which are named QoS parameters. The traffic classification is part of the QoS engine, and, in particular SLA traffic classes are internally translated to QoS performance requirements. The SLA traffic classes are closer to the user's language than the QoS performance parameters, and can be mapped to the *Mean Opinion Score* (MOS) that is understood by the users, and therefore used to measure the user's opinion.
- The QoE considers the user's experience/opinion and the QoS. In this analysis, the QoS is considered as one QoE parameter, that is, a high-layer metric.
- The characteristics of one platform are logical improvements that can be present in a platform or not. They are mostly high-layer parameters.
- Security requirements and mechanisms focus on the two first layers (high-level requirements and local properties). This is because requirements are needed at high-level, understood by users or services. Moreover, they may also be present in the mobile device as part of the built-on security suite.
- Attacks may occur at different layers, but here they are considered only as one more indication of possible risks.
- Consequences due to the influence of the parameters in the system.

Table 3.5: Action layers and parameters considered

	HIGH-LEVEL REQUIREMENTS (HLR)
SLA traffic classes	Streaming, interactive, conversational, background
Performance	Reliability, availability, fault tolerance
Security	Authentication, authorisation, confidentiality, integrity, non-repudiation, trust, privacy, accounting
Attacks	Social engineering
QoE	QoS traffic classes, user's experience (UX)
	LOCAL PROPERTIES
Resources	Power consumption, allowable memory
Characteristics	Context-based behaviour, inheritance, concurrency, location, NFC, QR-code
Security	Space randomisation, anti-tampering, encryption, public key cryptography, symmetric cryptography, secure key exchange, secure key redistribution, key generation
Attacks	Break-in
	COMMUNICATION
QoS Parameters	Data rate, packet size, signal strength, data transmission, transmission time, transmission power
Characteristics	Time-sleeping, required time-on, multiple antennas, buffering
Attacks	Tracking, eavesdropping, injection
Consequence	Retransmmission
	MEASUREMENTS
QoS Parameters	RTT, throughput, delay, jitter, packet loss, response time, BER
	ENVIRONMENT
Performance	Handover time, allowable bandwidth, error probability
Characteristics	QoS signaling, mobility support
Attacks	DoS, Malicious devices
Consequence	Interference, congestion, overhead, fading, shadowing, noise

In the following we analyse each level showing the relative PRM diagram, emphasising the Security and QoS tradeoff. Finally we show the results of the analysis. The dependency diagrams are defined based on the PRM, implemented using the DOT language[2] and interpreted using MATLAB.

3.1.3.0.2. High-Level Requirements Parametric relationships focused on the HLR Layer are shown in Figure 3.7. In this first diagram, it is possible to observe the dependencies chain. In particular, security requirements are taken into account in this layer because it is close to the service requirements and user needs. Specifically, both security requirements and QoS traffic classes affect the user's experience. Note that, in this layer, low-level QoS parameters (e.g. delay) are not taken into account, so if the security requirements are provided, apparently the user's experience only increases.

Moreover, there are no characteristics. Instead, the HLR layer defines closer-service requirements that may affect the Environment characteristics. Of course, if the response time is taken as one Measurement, it is also related with the HLR layer.

[2]www.graphviz.org

The relationships with the HLR parameters in the Mobile Platform layer are shown in Figure 3.9. Note that Trust can be considered as a parameter that influences the user's experience, because if the system is not trustworthy and the user considers this to be the case, then the experience becomes poor. Trust can be addressed as the union of reliability and security properties.

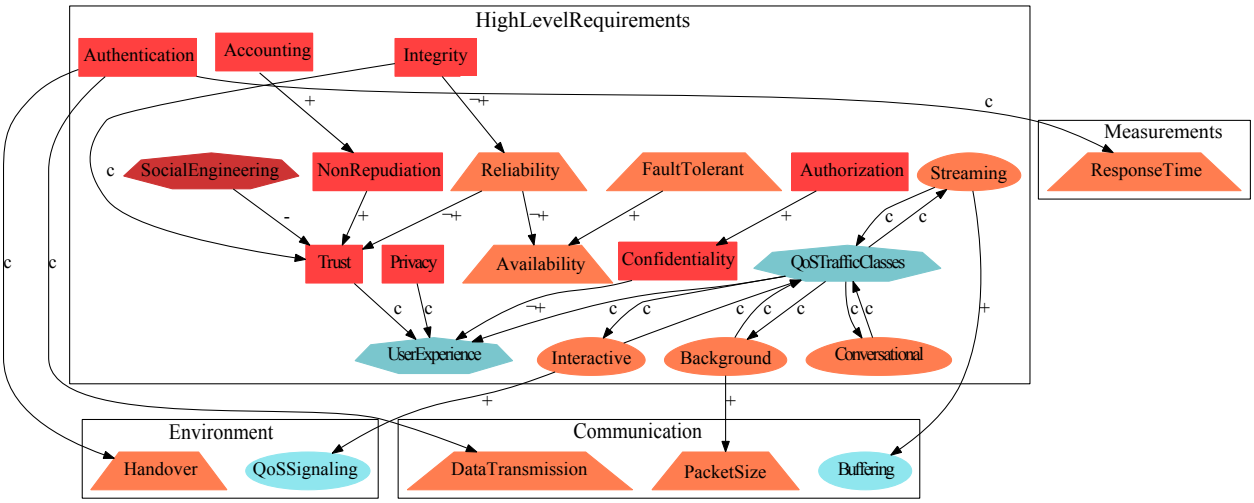


Figure 3.7: High-level requirements based on PRM

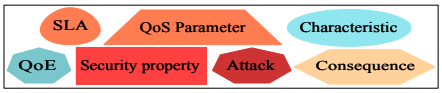


Figure 3.8: Legends for Figures from 3.7 to 3.12.

3.1.3.1. Mobile Platform (Local Properties)

Figure 3.9 also shows the parametric relationships related to this layer based on the PRM. In this layer the objective is to provide mechanisms that enforce high-layer requirements related with security and to ensure the QoS.

For this layer, security mechanisms, characteristics and local resources (e.g. memory) are considered. In this case, note that the Mobile Platform layer is related with the Communication layer in order to be used by a mechanism or to satisfy a security property (e.g. secure key exchange).

In this analysis, only power consumption and allowable memory have been considered as local parameters. Moreover, the allowable memory is defined in general terms, taking into account the memory of the user's applications and the memory at low level (e.g. buffer). However, it is possible to increase the number of parameters to be considered (e.g. time processing).

Note that security mechanisms are taken into account at this layer, and not in the Communication, Measurement or Environment layers. This is very important because nowadays

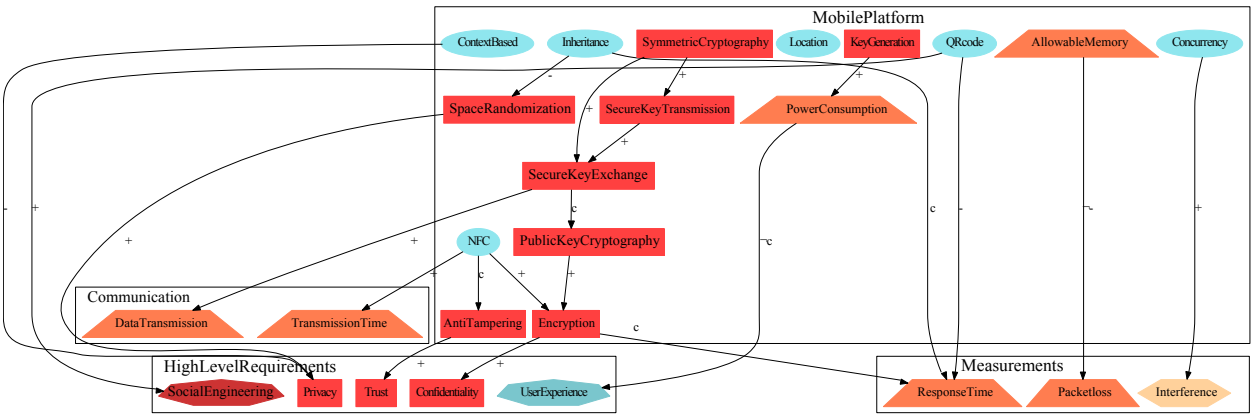


Figure 3.9: Local properties based on PRM

security mechanisms depend on the Mobile Platform. The only additional layer where security could be considered is the Environment layer, where Intrusion Detection Mechanisms could be taken into account, for example, in order to stop *Denial of Service* (DoS) attacks. However, these types of mechanisms are beyond the scope of the work here.

Furthermore, in the Communication layer, only the parameters related to the wireless interface are considered, and, in the Measurements layer, the QoS parameters used to evaluate the network performance are analysed. This is the reason why the following two layers do not consider security requirements or mechanisms.

3.1.3.2. Communication

Figure 3.10 shows the parametric relationships related to this layer based on the PRM. Specifically, in this layer, the local resources memory and power consumption are considered. In fact, the impact that wireless interfaces can have on the latter is well known. Intuitively, the measurements over the environment are influenced by the actions performed in the Communication layer.

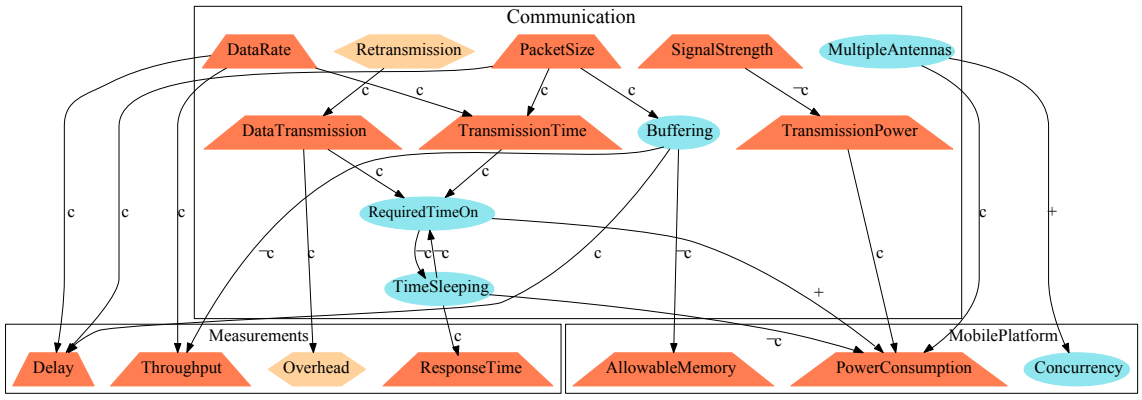


Figure 3.10: Communnication properties based on PRM

Furthermore, the parameters at the Communication layer are related to parameters at

the Measurements layers. Therefore, the decisions taken about the mechanisms in the Communication layer can affect the measurements of the system.

3.1.3.3. Measurements

Figure 3.11 illustrates the parametric relationships in the Measurements layer based on the PRM. Specifically, delay, jitter and packet loss are typical parameters for measuring the network's performance. In mobile platforms, it is also important to pay close attention to the response time and those parameters directly related to the type of service that a mobile platform is expected to offer satisfactorily.

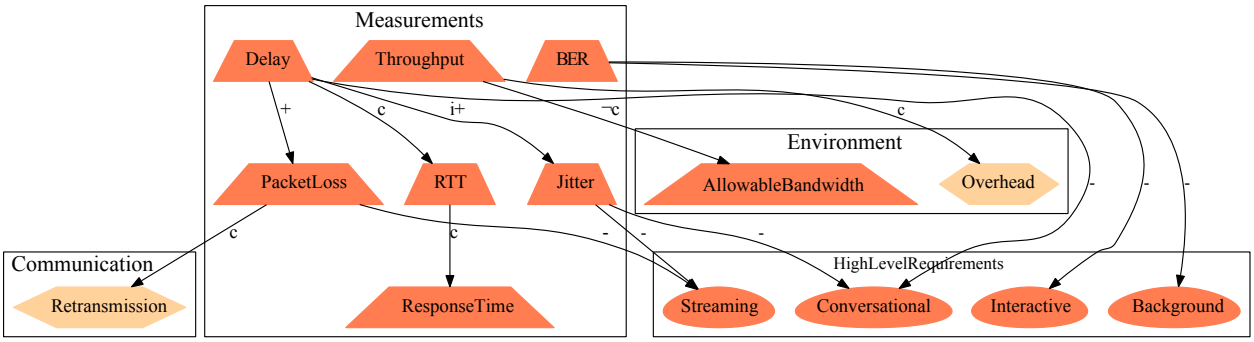


Figure 3.11: Measurements based on PRM

The relationships with the parameters in the HLR are very interesting because they provide feedback on the network utilisation. In fact, the type of traffic (SLA traffic classes) is highly dependent on the network performance and, in particular, on the QoS parameters considered in this layer. Moreover, Figure 3.11 shows that when a parameter affects any of the QoS parameters defined here, then the *SLA traffic classes* can also be affected, and so too the user's experience.

3.1.3.4. Environment

In the Environment layer, shown in Figure 3.12, the parameters of type Consequence are highlighted, because this layer represents the unpredictability of the context where the user is. Moreover, the Environment parameters are closely related to the Measurements parameters. It is intuitive because the objective of the Measurement parameters is to show the network conditions in order to prove that the QoS requirements defined in the high-layer are satisfied. In other words, by changing the parametric relationships (or their value) in the Environment layer according to one particular context, it is possible to get different measurements based on the context.

In addition, the presence of malicious devices or the influence of attacks such as DoS or the QoS signaling attack may be considered in this layer. In such cases, HLR as the availability or trust may be affected. Note that the QoS signaling is one characteristic which allows resource reservation along the path. However, these types of mechanisms can also be used by malicious devices in order to perform DoS attacks.

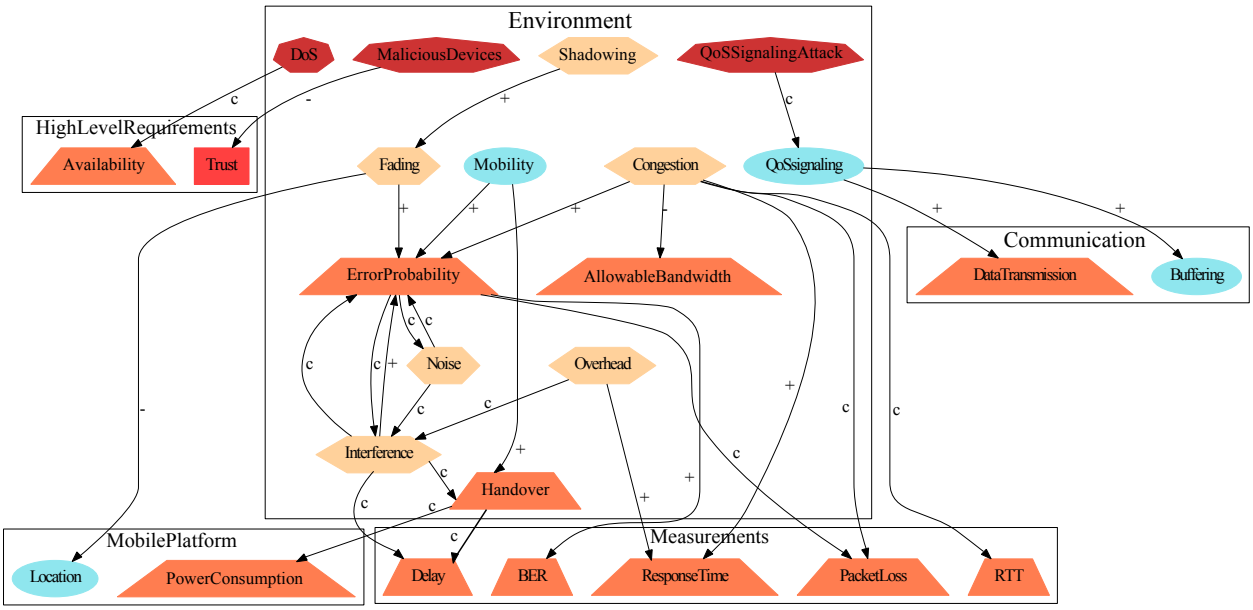


Figure 3.12: Environment based on PRM

3.1.4. Analysis based on Inter-Layer Results

PRM provides relevant data on a system once we have defined the parameters of the system accordingly. It is possible to extract the following information:

1. Influence on a parameter Y, $X \rightarrow Y$ (or on a set of parameters).
2. Dependence on a parameter X, $X \rightarrow Y$ (or on a set of parameters).
3. Impact on the system when a parameter (or a set of parameters) increases or decreases its value, or when a requirement is provided or not.

Figure 3.13 shows the acumulative influence (ι) and acumulative dependence (δ) based on the parameters considered in Table 3.5. ι and δ are calculated using Equations (3.21) and (3.22) respectively, for a generic parameter a which belongs to the set of parameters P defined in the PRM, as follows:

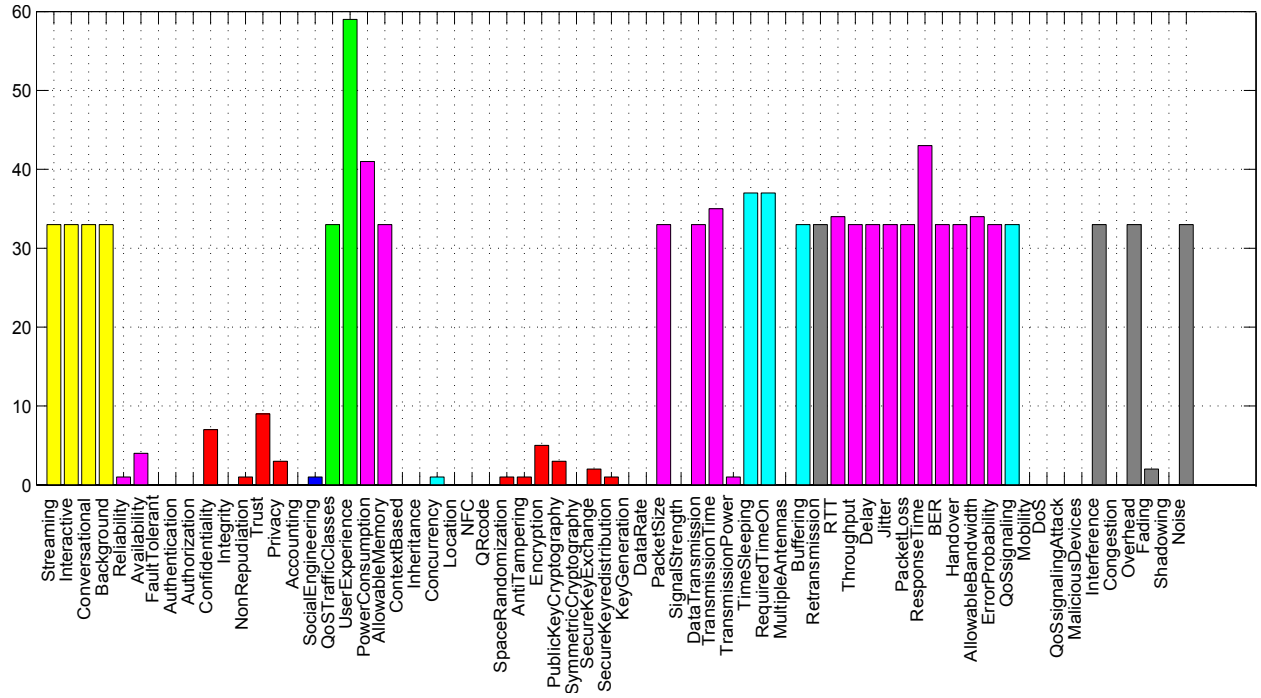
$$\iota(a) = |I_a| = |\{x | x \rightarrow a \vee xRa, x \neq a, x \in P\}| \quad (3.21)$$

$$\delta(a) = |D_a| = |\{y | a \rightarrow y \vee aRy, y \neq a, y \in P\}| \quad (3.22)$$

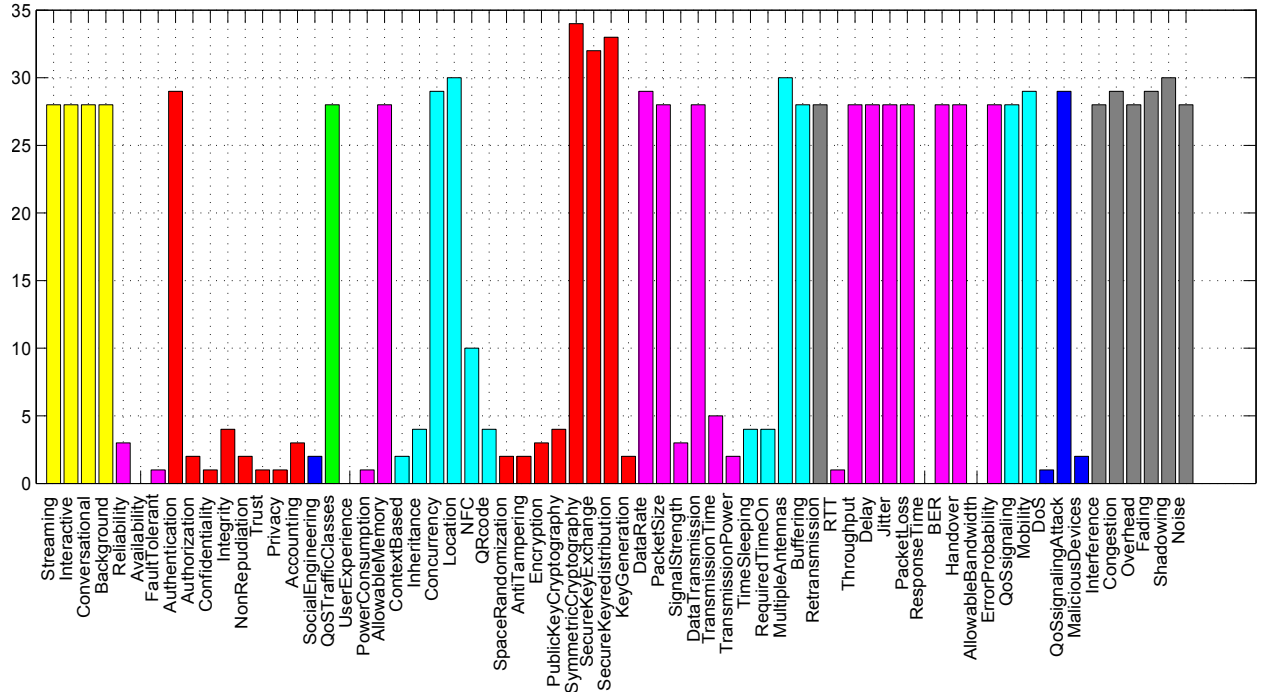
$$xRy \iff x \rightarrow y \vee \exists k | k \in D_x \wedge k \in I_y \quad (3.23)$$

3.1.4.1. Acumulative Influence

The acumulative influence on a parameter Y (Exp. (3.21)) reflects how many parameters X can affect the parameter directly ($X \rightarrow Y$) or indirectly through another parameter



(a) Acumulative Influence on Y, $X \rightarrow Y$



(b) Acumulative Dependence on X, $X \rightarrow Y$

Figure 3.13: Inter-layer results

($X \rightarrow ... \rightarrow Y$). In other words, in Figure 3.13 the parameters with values of up to zero are affected by the rest of the parameters and the parameters with values equal to zero are not affected in the diagrams considered. According to Figure 3.13(a), the QoS parameters are highly influenced by the rest of the parameters of the modelled system. In fact, they are influenced by nearly fifty percent of the parameters, characteristics and requirements considered [3].

It is especially important to highlight that the power consumption and the response time parameters are the most influenced parameters in this scheme, followed by the characteristics that in the Communication level describe the use of the antennas (e.g. required time on). This is logical, because the acumulative influence highlights those elements which in the end can be affected by the rest of elements.

For example, according to Figure 3.13(a) the user's experience may be influenced by the rest of the parameters. However, the user's experience value is not present in Figure 3.13(b), because in our scheme the user cannot take part in the system to modify the system's behaviour. However it is possible to intuit what can affect his/her perception from a business and usability point of view. So, in our scheme it is possible to influence the user's opinion but without them being able to do anything about it.

3.1.4.2. Acumulative Dependence

The acumulative dependence (or dependence degree) on a parameter X (Exp. (3.22)) reflects how many parameters Y are affected if the value of X changes (directly or indirectly through others parameters). The difference by type of parameter is not as remarkable as in the previous case. So, it is important to highlight one detail about how the measurements are taken. The values shown in Figures 3.13-3.14 are measured taking into account the formulation in Table 3.1, which does not consider weights on the links. This is because the weight should be set depending on the scenario being considered and will vary according to the current context where the user is. This issue is addressed in more detail in Section 3.2.

For example, according to Figure 3.13(b), the acumulative dependability value on trust is very low. However, this should not be misunderstood; in an insecure system of social environment this requirement could be very important. Moreover, security mechanisms such as the symmetric key cryptography or the requirement for a secure key exchange play important roles in this figure due to their impact on a long chain of performance measurements and requirements.

Regardless, note that even in a system where the links are marked with weights the number of relationships and acumulative dependability does not change. And, of course the system depends on the environmental conditions, security mechanisms, characteristics, and performance.

3.1.4.3. Impact of Security Requirements on the QoS Parameters

Finally, using the model it is also possible to get data about the positive or negative influence which a parameter or a set of parameters can have on the rest. The impact of

[3] In total around seventy six parameters are considered.

a parameter x on another parameter y, is measured according the Expressions (3.24) and (3.25).

The value of a parameter x (given by $v(x)$) is increased (Δ) or decreased (∇). When this happens, the system updates the values for the rest of parameters related with x. The updating process is described using Exp. (3.24)-(3.26), and depends on the operation performed on the antecedent on the relationship R defined betweeen the parameter x (antecedent) and the rest of parameters (consequents). The value of one relationship is given according with Ω, defined in Table 3.6.

$$\Delta x \implies \forall y | xRy, v(y) = v(y) + \Omega(R, \Delta x) \wedge u(y, \Omega(R, \Delta x)) \quad (3.24)$$

$$\nabla x \implies \forall y | xRy, v(y) = v(y) + \Omega(R, \nabla x) \wedge u(y, \Omega(R, \nabla x)) \quad (3.25)$$

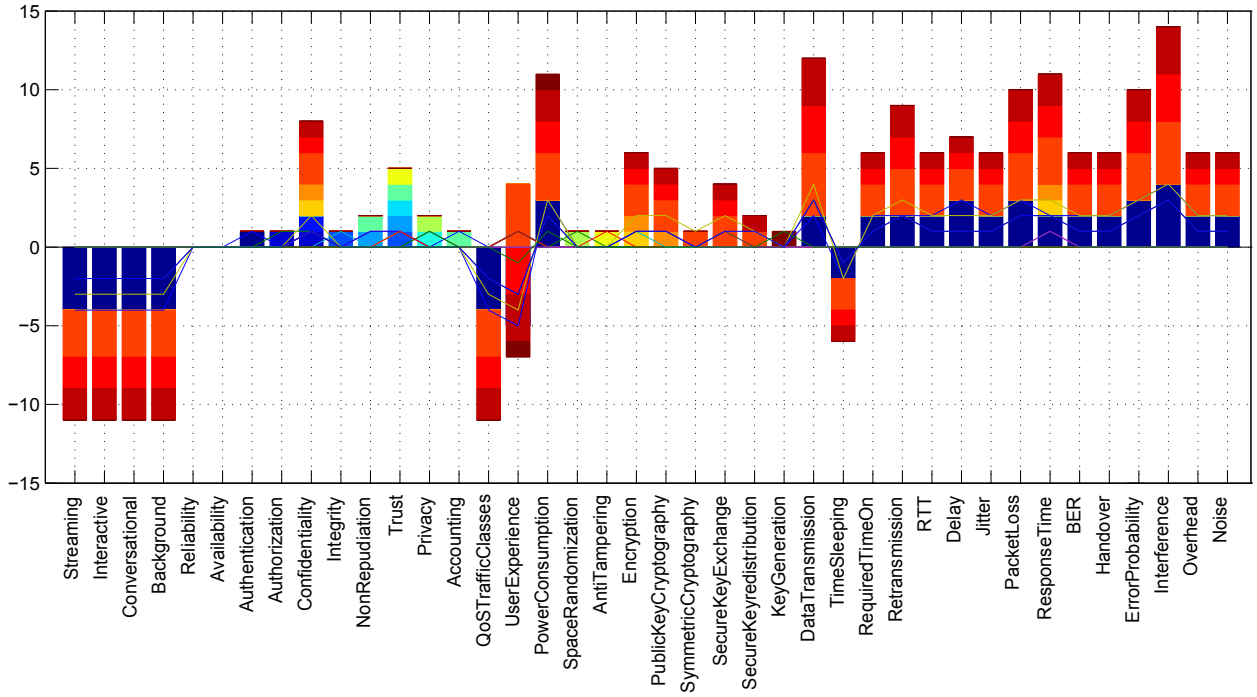
$$u(x, \omega) = \begin{cases} \Delta x & \text{if } \omega > 0; \\ \nabla x & \text{if } \omega < 0; \end{cases} \quad (3.26)$$

Table 3.6: $\Omega(R, x)$

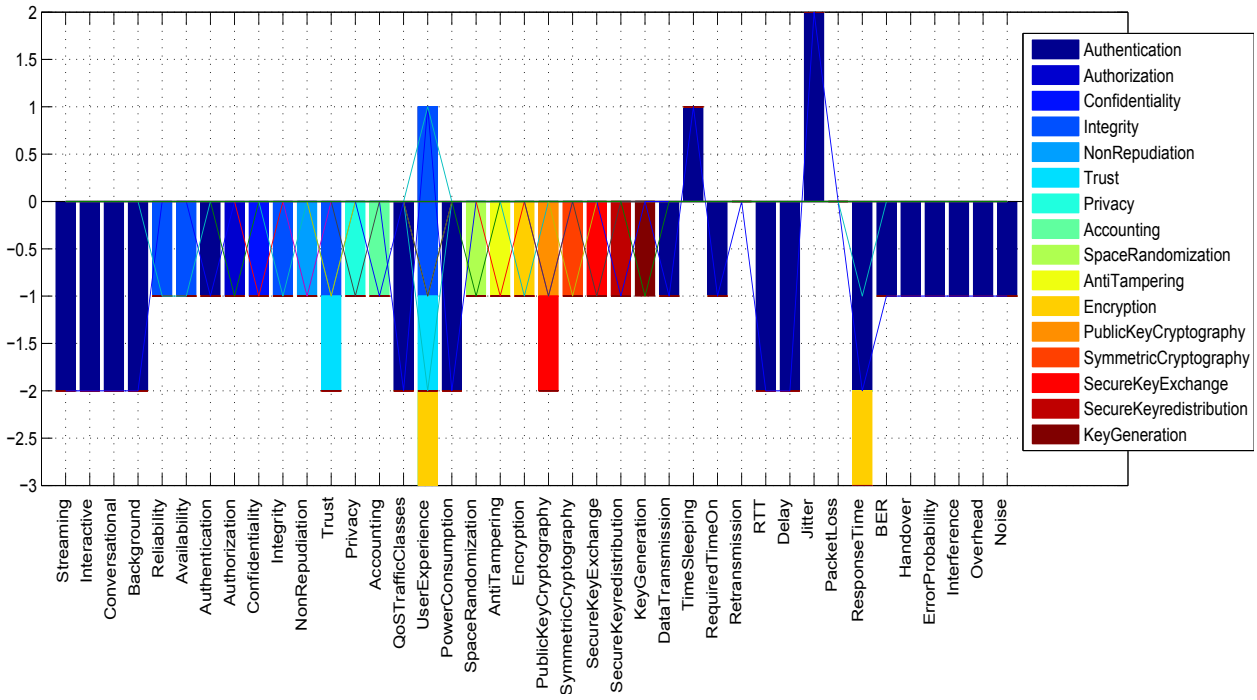
R	Operation on x	
(Symbol)	Increase x (Δx)	Decrease x (∇x)
$+$	$+\omega(+, a)$	NTD
$-$	$-\omega(-, a)$	NTD
$\neg+$	NTD	$-\omega(\neg+, a)$
$\neg-$	NTD	$+\omega(\neg-, a)$
c	$+\omega(c, a)$	$-\omega(c, a)$
t	$+\omega(t, a)$	$-\omega(t, a)$
$\neg c$	$-\omega(\neg c, a)$	$+\omega(\neg c, a)$
$i+$	$+\omega(i+, a)$	$+\omega(i+, a)$
$i-$	$-\omega(i-, a)$	$-\omega(i-, a)$

Ω is defined based on Table 3.1. Therefore, Ω decides whether or not, given a relationship defined in the PRM (R), the parameter in the consequent has to be increased or decreased, or instead, there is nothing to do (NTD, value 0 in the current model)[4]. Therefore, in general, the value in $\Omega(R, x)$ is given based on a weight ω that depends on R, but can also depend on x. Note that, $\omega(R, x)$ defines the weight that the parameter x has in the relationship. At this instance, ω is equal to 1 for all the relationships and parameters. The variation of this parameter would provide different contexts and ways of interpreting the information. However, it would require several testing proofs and lead beyond the scope of this work to describe this process in detail.

[4]For example, if aD^+b and ∇a, then the value of b is not modified, because b is only affected when a increases.



(a) Providing Security Requirements



(b) Retrieving Security

Figure 3.14: Influence of security on the system

For example, Figure 3.14(a) shows the negative impact that security mechanisms have on network performance and therefore on the QoS parameters. Note that this behaviour decreases but also increases the user's experience. This is because in our model the security requirements enhance the user's opinion about the system, because he/she feels safer. However, the performance degradation cannot be ignored, even more so when the power consumption is vastly increased according to the model. Again, in a context where the links are marked with weights, this behaviour should be modified based on the importance to the system of QoS traffic classes against security requirements. This could vary, depending on the context where the user is (at home or walking around).

Moreover, in Figure 3.14(b), we can see two main points: first, it has to be noted that the user's experience increases and also decreases as in Figure 3.14(a). This is because in this case the overhead caused by the security mechanisms is not present. Moreover, the only negative influence on performance is that the jitter increases, but only because the delay is fluctuating[5].

However, the user's experience decreases because the environment is not secure and the user may have noticed. This situation is highly dependent on the context and also on the user. Once again, the user's opinion about the security parameter is very subjective.

3.2. Context-based Parametric Relationship Model

PRM provides a language where different parameters can be represented, as well as their relationships. Thus, using that model, and given a relationship $A \rightarrow B$, it is possible to extract the influence on parameter B, the dependence on parameter A, and the impact on the system when a parameter increases or decreases its value.

This is very useful in order to determine the Security and QoS tradeoff in a system. However, the main problem with such a solution is that the context is not considered. Therefore, all the parameters, relationships and dependencies have the same relevance in the system, but this is unrealistic because all systems depend on a context. More specifically, the systems have general parameters and specific parameters that are, in the end, considered together. So, defining a *Context-based Parametric Relationship Model* (CPRM) is, to our understanding, the natural step towards evaluating the security and QoS tradeoff in a system.

For that reason, we enhance the definition of PRM to allow the integration of the context, thereby allowing us to modify the behaviour of the system to work with a *General Context* (GC) and *Particular Contexts* (PCs). We call this improved model the *Context-based Parametric Relationship Model* (CPRM). Moreover, the proposed extension allows the exchange of contexts between CPRMs. In our opinion, the context in a system is defined based on the parameters present in the system (defined at different layers and of distinct types), the relevance of each parameter to the system (what really matters in the system, administrative decisions) and the real impact of each parameter on the system (depends on the mechanisms and technologies used, and cannot be subjective).

One possible way of setting up the context in a system is to target the relevant components with weights in order to show their importance and impact. Based on the previous definition

[5]The jitter is increased because the delay is continuously decreasing while the test. Once the delay is stable, then jitter remains stable too.

of context, it can be understood that we need, as minimum requirements, weights for each parameter (A and B), to measure their relevance, and for each relationship, to measure the impact that, for example, an increment of A has on B. But, as we will show later, additional weights need to be considered in order to correctly define a context.

3.2.1. Requirements for a Context-based PRM

As mentioned, CPRM is an enhancement of the previous model so as to be able to interpret different types of contexts. This can be understood from various points of view. First, it is able to distinguish between GC and PC. Second, in order for the final CPRM-based system to be extensible and allow a grain-fine configuration, it has to take into account different types of weights: for each parameter (w_p), dependency $\rightarrow$ (w_d), type of parameter (w_t), layer (w_l), and operation (w_o).

It is understood that, while some of these weights can never change, others may vary according to a specific context. So, in the solution proposed, GCs define the general behaviour of the system, and PCs define the parameters used to support the requirements and other general parameters described in GC.

The PRM has to be updated in order to consider the GC and the PC. To do so, and also to enable separation of the GC and PC from the PRM, we define the CPRM structure, which is a PRM with general weights, and also the instance of a CPRM ($CPRM_i$), which is the instantiation of a CPRM based on a PC. Figure 3.15 shows the components in a CPRM-based system, as well as the process followed in order to obtain a $CPRM_i$. $CPRM_i$ represents the final behaviour of the system, in which all the mechanisms and technologies that are relevant are finally chosen by the administrator and taken into account when extracting relevant information of the model. The user/administrator sets the contexts using the GUI. The $CPRM_i$ can integrate various PCs, but only a GC.

Finally, dividing the overall context into general and specific, enables the context to be modified separately. So, the GC in a $CPRM_i$ can be changed, as well as the PCs. Subsequently, rules are applied to maintain the coherency in the model. In other words, the behaviour of the model can change based on the *Action Rules* (AR), that are used when a rule is not satisfied, hence making the model consistent again. AR are defined according to the rules shown in Table 3.8. Note that A3 enables a parameter to collaborate with any other parameter defined as a consequence of the dependence of any of its parents. However, in the current implementation, A3 is applied if and only if there is no child of k related with x ($\nexists p | k \in P(p) \wedge d(x,p)$). By doing so, the user can force any instance of the parameters to be solely related to a set of instances of another parameter. In any case, R2 and the property of inheritance in R3 have to be maintained.

The modification of the equations defined for the PRM was carried out in order to consider the integration with the GC. The integration with the PC is defined for the CPRM and is based on a set of rules for performing the instatiation of the CPRM based on the PC. We show both processes in the following sections.

Table 3.7 shows a summary of the formulation to PRM-based systems provided in Section 3.1 (the same identifiers are used). In the following paragraphs, the focus of the reasoning will be directed to enhance the model in order to be able for integrating contextual information dynamically.

Table 3.7: Parametric Relationship Model definitions.

Basic Formulation Set (BFS)		Complex Formulation Set (CFS)	
$D^{+} :: aD^{+}b \Rightarrow (\Delta a \rightarrow \Delta b)$	(3.1)	$D^{c} :: (\Delta a \rightarrow \Delta b) \wedge (\nabla a \rightarrow \nabla b) \equiv aD^{+}b \wedge aD^{\neg+}b$	(3.5)
$D^{-} :: aD^{-}b \Rightarrow (\Delta a \rightarrow \nabla b)$	(3.2)	$D^{t} :: aD^{c}b \wedge bD^{c}a$	(3.6)
$D^{\neg+} :: aD^{\neg+}b \Rightarrow (\nabla a \rightarrow \nabla b)$	(3.3)	$D^{\neg c} :: (\Delta a \rightarrow \nabla b) \wedge (\nabla a \rightarrow \Delta b) \equiv aD^{-}b \wedge aD^{\neg-}b$	(3.7)
$D^{\neg-} :: aD^{\neg-}b \Rightarrow (\nabla a \rightarrow \Delta b)$	(3.4)	$D^{i+} :: (\Delta a \rightarrow \Delta b) \wedge (\nabla a \rightarrow \Delta b) \equiv aD^{+}b \wedge aD^{\neg-}b$	(3.8)
		$D^{i-} :: (\Delta a \rightarrow \nabla b) \wedge (\nabla a \rightarrow \nabla b) \equiv aD^{-}b \wedge aD^{\neg+}b$	(3.9)

Acumulative Influence (ι)		Acumulative Dependence (δ)	
$\iota(a) = \lvert I_a \rvert = \lvert \{x \lvert x \rightarrow a \vee xRa, x \neq a, x \in P\} \rvert$	(3.21)	$\delta(a) = \lvert D_a \rvert = \lvert \{y \lvert a \rightarrow y \vee aRy, y \neq a, y \in P\} \rvert$	(3.22)
		$xRy \iff x \rightarrow y \vee \exists k \lvert k \in D_x \wedge k \in I_y$	(3.23)

Impact Increasing (Δ) and Decreasing (∇) a Parameter x

$$u(x,\omega) = \begin{cases} \Delta x & \text{if } \omega > 0; \\ \nabla x & \text{if } \omega < 0; \end{cases} \quad (3.26)$$

$$\Delta x \implies \forall y \lvert xRy, v(y) = v(y) + \Omega(R, \Delta x) \wedge u(y, \Omega(R, \Delta x)) \quad (3.24)$$
$$\nabla x \implies \forall y \lvert xRy, v(y) = v(y) + \Omega(R, \nabla x) \wedge u(y, \Omega(R, \nabla x)) \quad (3.25)$$

$\Omega(R, op(x))$, $R \in \{+, -, \neg+, \neg-, c, t, \neg c, i+, i-\}$, $op \in \{\Delta, \nabla\}$: (Table 3.6)

	$+$	$-$	$\neg+$	$\neg-$	c	t	$\neg c$	$i+$	$i-$
Δx	$w_{x,+}$	$-w_{x,-}$	ntd	ntd	$w_{x,c}$	$w_{x,t}$	$-w_{x,\neg c}$	$w_{x,i+}$	$-w_{x,i-}$
∇x	ntd	ntd	$-w_{x,\neg+}$	$w_{x,\neg-}$	$-w_{x,c}$	$-w_{x,t}$	$w_{x,\neg c}$	$w_{x,i+}$	$-w_{x,i-}$

3.2.2. Modifications in the Model: Setting up a General Context

The expresions (3.21)-(3.23) contained in Table 3.7 do not change, because these formulations are independent from the context.

$$\Delta x \implies \forall y \lvert xRy, v(y) = v(y) + w_T \wedge u(y, w_T) \quad (3.27)$$
$$\nabla x \implies \forall y \lvert xRy, v(y) = v(y) + w_T \wedge u(y, w_T) \quad (3.28)$$

However, Equations (3.24)-(3.25) change into (3.27)-(3.28), in order to add the *total weight* (w_T, Exp. (3.29)) based on the weight for the parameter in the antecedent (w_p), the type (w_t) and layer (w_l) of that parameter, and the weight for the operation w_o. Regarding w_o, in this definition it is independent from the weights in Ω ($w_{x,R}$). Ω shows the information given by the definition of the BFS and CFS (3.1-3.9), while w_o takes a subjective value.

$$w_T = (\Omega w_d)\frac{w_p + w_o + w_t + w_l}{max_p + max_o + max_t + max_l} \quad (3.29)$$
$$w_d(a,b) > 0 \ \forall a, b \lvert aRb \quad (3.30)$$

Note that w_p, w_t, w_l and w_o can be subjective. However, w_d should be defined according to the true effect that the parameter in the antecedent (aka x) has on the parameter in the consequent (aka y) given the relationship R and the action performed a ($\Delta or \nabla$). Moreover, w_d should be propagated to the rest of the dependencies in the chain.

The formulation (3.30) adds an additional condition to build the GC: w_d is at least equal to 1, because if there exists a relationship between two parameters then there is an effect on the system to be measured. The current implementation of the model permits setting up any value for weights but, when w_d is equal to 0, then the w_T is also 0.

Using the aforementioned formulation it is feasible to perform the operations previously used in the PRM, but using a given context based on weights. However, it is necessary to provide new rules and action rules in order to keep the model coherent when certain

parameters are introduced by an administrator during the execution of the CPRM-based tool for administration.

In order to provide the functionality needed to introduce a PC and match it to the CPRM, we provide definitions for a PC, rules to make it consistent with the current GC in the CPRM, and rules to decide how the different types of parameters cohexist and are taken into account.

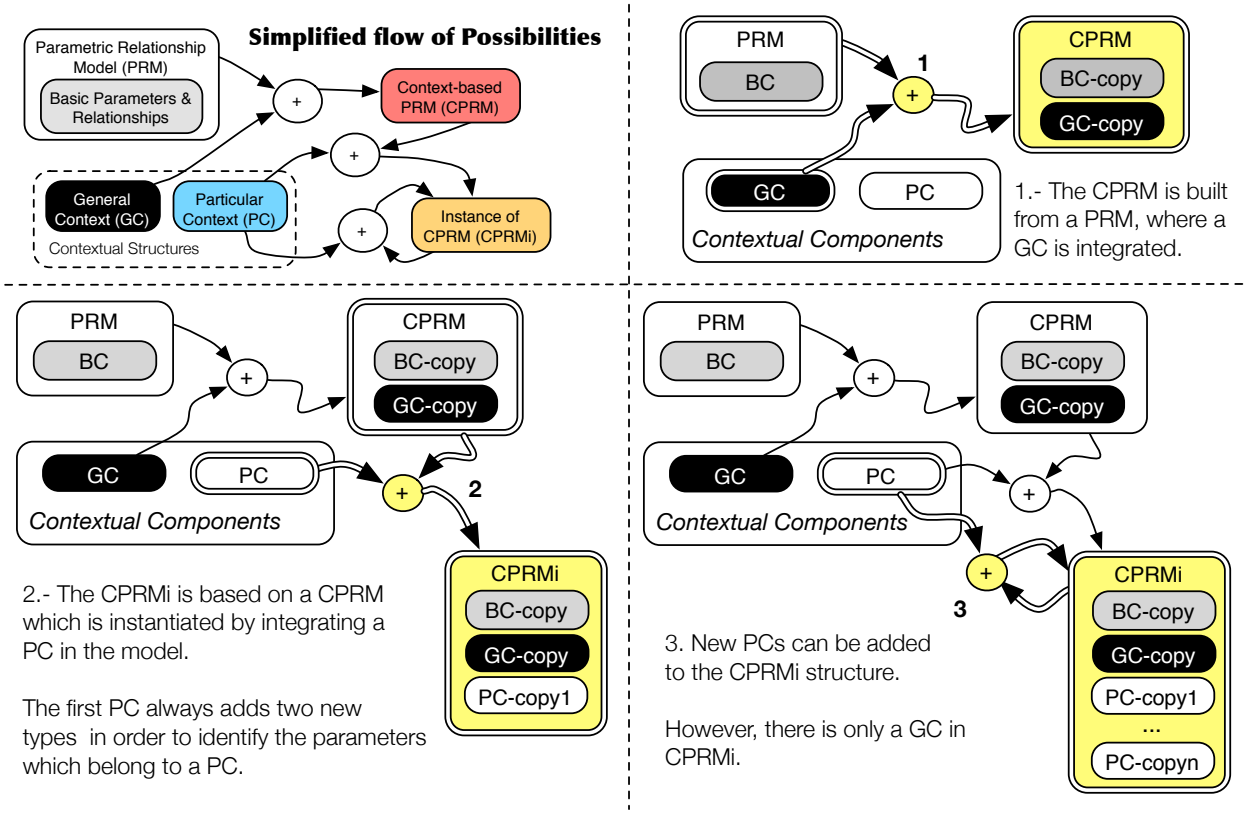


Figure 3.15: Steps in integrating contexts.

3.2.3. How the Model Schemes are Built

The integration of parameters, is decomposed into different steps, associated with what is called *model* schemes. Then, each BC is defined as part of a Parametric Relationship Model (PRM), while GCs are present in *contextual* schemes, as CPRM or $CPRM_i$, and, finally, PCs are present in *instantiated* models, denoted as $CPRM_i$ (that is, PRM $\subset$ CPRM $\subset$ $CPRM_i$). In particular, GCs and PCs are used as pieces, integrated into the model schemes so as to build new model schemes. The steps, for integrating contexts in order to have a $CPRM_i$ from a PRM, without going into more detail, are shown in Figure 3.15.

A PRM is a non-contextual structure, which defines a set of parameters and their relationships. To improve the analysis of the influence degree between parameters, these are defined based on a type and a layer. For example, Figure 3.7 shows a section of a PRM

model[6]. In this, the parameter *Privacy* is of type *Security*, and can be found at the *High Layer Requirements* layer. This separation into layers and types, improves the classification of parameters, and makes the selection of a set of parameters based on a type or a layer possible. Moreover, the model works with a set of predefined relationships in order to show the impact that the increasing/decreasing of a parameter has on the rest of parameters. Note that, what is useful for us, is to be able to detect what happens when a parameter is increased, decreased, enabled or disabled.

However, in a PRM, all the parameters have the same relevance. That is to say, in the relationship defined as $A \xrightarrow{+} B$ and $A \xrightarrow{+} C$, when A increases it has the same effect in B as C. This is impractical, and is unrealistic, because there are some parameters that are more sensitive than others. For example, imagine a communication protocol, denoted as $C1$, that increases the energy consumption E and the computation time CPU, when used. We know that $C1 \rightarrow E$ and $C1 \rightarrow CPU$, however, in scenarios where E is a valuable resource, maybe it is preferable to have a protocol $C2$ that minimises E even at the expense of increasing the CPU.

Therefore, CPRM-based models were defined to take these differences into account. A CPRM can be built based on an existing PRM, and a GC, that defines the context, given as weights, for the elements in the PRM. Then, as a result, we have a contextualised PRM, denoted as CPRM. By that point, our schema knows that there are parameters which are more relevant than others and also that the relationships can have different values. However, it is not possible to improve the model in order to define mechanisms that implement a parameter. For example, let us look at the previous relationship, which is built on the basis of the general behaviour of the communication protocols. If we define a new communciation protocol $SecureC3$, with additional relationships, then, intuitively, $SecureC3$ can inherit the relationships of its generic predecesor, $C1$. Moreover, the scheme should learn that a communication protocol can be secure, and then considers the relationships between the *father*, $C1$, and some of the consequents of the *child*, $SecureC3$, in order to improve the general behaviour of the model.

In order to allow this concept, where some new parameters are defined in order to provide a specific knowledge of the environment, we define the *instantiation* as the process of defining mechanisms, as new parameters, that implement parameters in the current CPRM (Figure 3.16). The new model or scheme is denoted as $CPRM_i$.

As Figure 3.15 shows, a $CPRM_i$ is built, based on a CPRM and a PC. Moreover, a new $CPRM_i$ is always possible from an existing $CPRM_i$ and an additional PC. In fact, PCs represent the dynamic nature of the model, as a result of a wider understanding of the environment. In other words, while the weights in the GC can be subjective, or even approximate, in a PC it is expected that the weights in the relationship will be accurate. Diverse examples are shown in Chapter 6. In particular, Table 6.2 shows an example of PC, where the parameters *Authentication* and *SignatureScheme* are instantiated using the instances CAS and DAS (Authentication) and ECDSA and PairingBased (SignatureScheme).

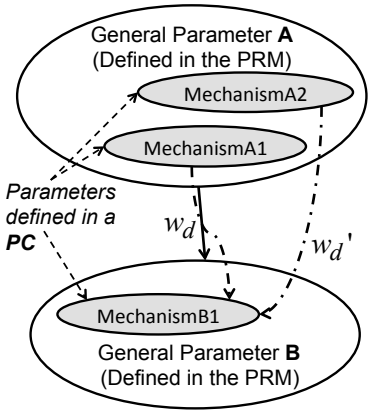


Figure 3.16: Instantiation of parameters A and B.

Table 3.8: Rules (R) and Action Rules (AR).

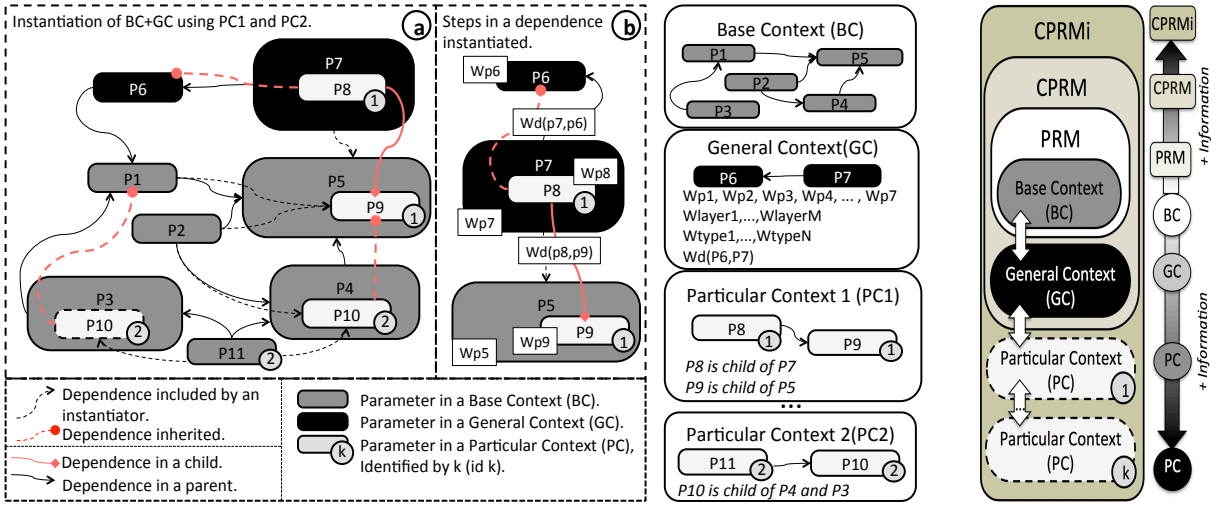
Rule	Action Rule
R1 A parameter in the PC is related to at least one parameter in the PRM (parents).	AR1 If not, the independent parameter is considered as a new parent and added to the PRM to make it consistent.
R2 Given P(x) and P(y) the list of parents of x and y, respectively $\|P(x) \cup P(y) \subset PRM$. If d(x,y), exists, then $\exists k \in P(x) \wedge \exists z \in P(y) \| d(k,z)$.	AR2 Otherwise, said relationship between parents has to be added to the PRM in order to make it consistent, with $w_d(k,z) = 0$.
R3 A parameter in the PC inherits the relationships of its parents, by default: if z belongs to P(x) and d(z,k), then d(x,k) is possible, with weight w(z,k) by default.	AR3 The relationship d(x,k) is added with w(z,k) $\forall k$. If $\exists p \| k \in P(p)$ and therefore, according to R2, $d(x,p)$, w(x,p) don't change.
R4 A parameter x inherits the layer of its parents and the type of its parents. When $\exists k, z \in P(x) \| type(k) \neq type(z)$, then $type(x) = [type(k)type(z)]$.	AR4 The decision model can fix the type of the layer of a parameter, but, even so, the final layer and type match with the layer and type of a parameter p in P(x).

3.2.4. Rules & Action Rules

The previous steps for integration of contexts are not possible without the definition of a set of rules to maintain the coherence of the definition of the model, through the process of instantiation of the CPRM based on the PC. In other words, the behaviour of the model can change according to *Action Rules* (AR), that are used when a rule (R) is not satisfied, hence making the model consistent again. ARs are defined according to the rules shown in Table 3.8, which express the correct behaviour expected in the model. In order to clarify the application of ARs, we use the example shown in Figure 3.17, where the final relationships in a $CPRM_i$, in which two PCs are integrated (using ARs), is illustrated.

For example, rule R1 is set to ensure that the PRM contains a basic set of parameters, so as to define the common relationships in heterogeneous environments. Then, if a new

[6]In general, these schemes are very complex because of the large number of relationships defined.



(a) Final CPRM instance ($CPRM_i$) (b) Components

Figure 3.17: Contexts as integration components

PC contains parameters without a parent in a PRM, it can be interpreted in two ways: the parameters in the new PC have been bad defined, or the parameter without parents defined is a new parameter to be considered as part of the PRM basic set (as part of the BC). We have to assume that the PC is well defined, so, when a new parameter is identified, it is added as part of the BC in the PRM (AR1), making it a new parent to be considered. If not, the definition of BC would be unstable. Then, in accordance with AR1, in order to integrate PC2, the parameter P11 is added as a new PRM parameter. However, usually such additions occur at an early stage of recognition of the environment, or as part of the aggregation of a GC (e.g. P6, P7).

On the other hand, rule R2 protects the inheritance of the parent's relationships. As the parents are considered as the most general parameters, it is expected that these parameters define all the possible relationships. Moreover, as the instances inherit their relationships from their parents, then, the relationship between two parameters is not possible if their parents are not related. For this reason, the relationship between two parameters whose parents are not related, makes the model inconsistent, and is avoided by adding new relationships between the parents (AR2).

However, new relationships added between parents to make the model consistent according to rule R2, cannot have the same weight as the relationships between their instances, because any new parameter that instances the parent, inherits this new relationship (and the default weight). So the weight for this new relationship between parents, set by AR2, is 0 ($w_d(k, z) = 0$). This, for example, is the case of the new relationship added between P7 and P5.

Note that, the previous relationship, $d(P7, P5)$ is different from $d(P4, P5)$ which was defined in the PRM. Therefore, R3 defines the inheritance of relationships from the parents to the instances. For example, P9 and P10 are defined in PC1 and PC2, respectively. Therefore, as P4 depends on P5, so P10, defined as a child of P4, inherits this relationship

with the weight defined by default (AR3), $w_d(P4, P5)$. So, this relationship implies that P10 will be related to P9 as a new possible combination[7].

Moreover, when the parameter $P10$ is added, it inherits the relationships of its parents based on AR3. It means that $d(P10, P1)$, $d(P10, P9)$ are possible, with $w_d(P10, P1) = w_d(P3, P1)$ and $w_d(P10, P9) = w_d(P4, P5)$, respectively. Furthermore, the relationship $d(P2, P10)$ is added as a consequence of the relationship $d(P2, P4)$ defined in the PRM. Then, according with AR2, $d(P2, P3)$ has to be added too, because P10 is also a child of P3, and P2 is related to it. However, $w_d(P2, P3) = 0$, and this does not affect $P10$, which is affected only by the inheritance of the relationship $d(P2, P4)$.

Finally, R4 determines that all the parameters inherit the type and layer of its parents, and AR4 enforces this. So, it is unnecessary to provide this type of information in the PC. However, note that, when AR1 is applied, the creation of the new parameter may need this additional information. In this case, the type of the new parameter is *instantiated*, because it was built as part of a PC. For this reason, it is very important that new parameters will be added in an early step (in a BC or GC), because, although the model considers these types of events, adding new parents from a PC is not the objective and breaks the initial purpose of maintaining a stable context separate from the dynamic context.

Some examples of the application of these rules in specific environments are provided in Chapter 6. However, in the use cases that have been considered, AR1 is not applied because the new parameters in the PCs are instances of the parameters in the CPRM. The second use case shows some examples where the instances introduces new information in the model. For example, in the PRM used to the second use case (See Appendix A.2.1), the relationship between TransmissionCapacity and Eavesdropping does not exist. However, after the instantiation, this relationship is stablished, because the instance of TransmissionCapacity named EavesdropperCapacity is related with Eavesdropping (Figure 6.21). Moreover, in the first use case we provide an example where AR3 is forced. This occurs because the parameter SignatureScheme is related with PowerConsumption (e.g., see Figure 6.2), and the instances of SignatureScheme (ECDSA and PairingBased) inherit this behaviour (e.g., Figure 6.4) that is not included into the PC. This can be seen in Figure 3.18, where the relationships that are inherited are highlighted.

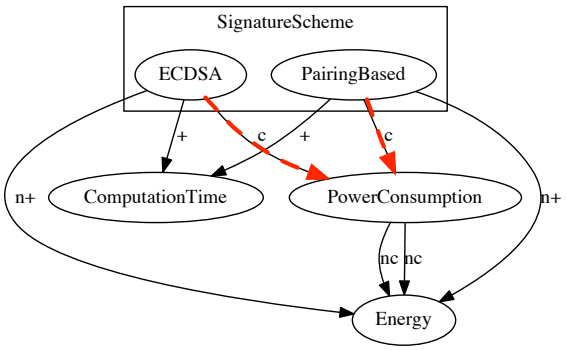


Figure 3.18: SignatureScheme (Chapter 6).

[7]Note that Figure 3.16 illustrates R3 to $MechanismA1$.

3.2.5. Requirements for the Integration in a $CPRM_i$

The action rules shown in Table 3.8 are taken into account when a PC is used to instantiate a CPRM. However, before setting up the rules, a new structure is needed in order to represent an instance of the CPRM. The $CPRM_i$ has to be defined taking into consideration a set of requirements:

- Independence from the original CPRM.
- Coherence between parameters in the model (Table 3.8).
- Adaptation capability: acceptance of new PC to be added to the existing one.
- Ability to return to the original CPRM behaviour.

Although CPRM is an extension of PRM in order to consider context-based behaviour, $CPRM_i$ has to be considered as an *instantiation of a CPRM based on a PC*. The $CPRM_i$ is not a new version of PRM or CPRM, because its purpose is not to define a model or implement functions. In fact, its purpose is to change its behaviour based on different contexts. Thus, a $CPRM_i$ is always built based on a CPRM and a PC, but when it is created, the original data in the CPRM should be cloned in the $CPRM_i$ to make it independent.

The current definition of $CPRM_i$ has been built, based on the following steps:

1. The $CPRM_i$ adds the special types *Instance* and *Instantiated* to the model.
2. The type Instance will be the type in the model for the parameters included from the PC. Of course, during the inference process the given parameters take the original type of their parents according to a set of rules. With these tags in the model it is possible to properly identify which parameters belong to a given PC. As a result, it is possible to return to the original CPRM behaviour.
3. If a parameter y is Instantiated, that is, if $\exists x \in PC | y \in P(x)$, then in the $CPRM_i$ the parameter becomes a new layer. The new layers which represent instantiated parameters are separated from the rest of the layers, defined in the model. When the PC is retrieved, the parameters which cease to be instantiated return to the original list of parameters in the model.
4. When the $CPRM_i$ receives a new PC, the new parameters are integrated in order to maintain the coherence in the model, based on the rules in Table 3.8.

When an instantiated parameter is represented as a layer, this adds the possibility of calculating the effect that the whole layer has on the performance of the system, thereby making it possible to evaluate the tradeoff between different mechanisms. The composition of parameters is shown in Figure 3.16, where the parameters A1, A2 and B1 are instances of A and B. The approximate weight for a general relationship $A \Rightarrow B$, w_d, is replaced by the specific weight, w'_d defined for $A2 \Rightarrow B1$. Moreover, $A1 \Rightarrow B1$ uses the general weight w_d because there is not a specific weight defined for it.

3.3. Summary

In order to combine the analysis of diverse mechanisms to provide Security and QoS under a common framework, a *Parametric Relationship Model* (PRM) was defined. The model

defines the structure that dependencies-based systems should have in order to provide useful information to be analysed from a tradeoff perspective. Moreover, this model is extended to provide a *Context-based Parametric Relationship Model* (CPRM) where the steps required to integrate new dependencies based on new conditions, for providing a new context, are defined. So, finally, the superposition of contexts, defines the new behaviour of the system, and this behaviour can be analysed based on a set of well defined dependencies.

The structure of a CPRM system is based on a set of parameters and the relationships between them, a set of operations (*op*) which define the effects on the dependent parameters, and a set of weights which define the relevant subjective and non-subjective of the components in the model. So, an administrator could consider, subjectively, that *trust* is a key parameter for the system's survival at a specific time or in a particular context (e.g. when the nodes have to vote whether or not to reject a node from the network due to malicious activity). Under said assumption, trust would have a higher weight than the rest of the parameters in the dependencies system. Of course, if the purpose of the network changes, as a consequence of the security policies, or motivated by the environment of the user, then, the subjective values should also change.

For example, when the environment is well know, like at home, the user could relax the relevance of the mechanisms of trust, because the knowledge of the devices of their environment shifts their concerns onto other issues, maybe the performance. In this sort of scenario, the user could consider the security as a consecuence of his location. However, fortunately, not all users have to have the same perception, and for this reason, the subjective values are needed.

Moreover, the model considers non-subjective values; devised to define the impact of the cascade effect that a dependence can trigger. These values, are defined as weights linked to the dependencies in a *General Context* (GC), whose value is calculated, in the first place, based on an approximation. Once the parameters are *instantiated*, that is, at least one mechanism has been defined which implements the parameter, then, the weight for the inherited dependence is updated to the weight defined in the *Particular Context* (PC). For example, the mechanisms implementing trust, are seen as parameters built on a PC (how the real environment implements the parameter), and then, these parameters can add real values, measurements of the impact that said solutions have on a real sytem, for a context. These values are specific, non-subjective values, and can replace previous non-subjective values defined in the model for the general parameter (trust).

To reach this point, where different contexts are integrated or even interchanged, the definition of a clear set of structures is fundamental. To do this, the model starts from a basic set of parameters, which define, in a general way, the scenario to be evaluated. This *Base Context* (BC) does not change, it is fixed [8]. However, the interpretation of the model, and the weights/relevance of the parameters can change, according to the integration of new contexts. So, the BC is always waiting so that their parameters, relationships, layers, types and operations, can be taken as the values of context (weigths) defined in a GC, and, after that, in the subsequent PCs that are integrated.

The BC is the result of an exhaustive analysis of the current architectures and the en-

[8]The BC is considered as an empty context with regard to GC and PC. So, in the following, the term *context schemes* makes reference to GC and PC, but not BC.

vironment where the CPRM-based tool defined in Chapter 4 will be deployed. In our case, the BC that we use is taken from the analysis of the Security and QoS mechanisms in the Future Internet defined in Chapter 1 and Chapter 2. Therefore although the tool proposed allows the definition and the use of BC personalised by the user, and is hence extensible to various areas of study, our main objective is to use it to assess the Security and QoS tradeoff. In fact, our BC defines these types of relationships and no other, although they could be added, depending on the area of study desired.

CHAPTER 4

Security and QoS tradeoff Tool (SQT)

In this chapter, the steps for implementing a handler for CPRM-based components is detailed. The prototype is implemented in MATLAB following the specification and rules defined to CPRM-based systems, and the set of Security and QoS Tradeoff parameters provided, from which the tool, SQT, is named. A component-based architecture is previously defined in order to enable the steps for the integration, to satisfy the previous requirements of independence between models, coherence and adaptation capability. Moreover, the steps in the integration of components using SQT is detailed, as well as the functionality implemented which is accesible using the Graphical User Interface (GUI) developed with this purpose. Finally, the performance of the solution is evaluated based on the cost for integrating new and dynamic information in the model. This is the cost of the behavioural changes.

4.1. Motivation for a Security and QoS Tradeoff Tool

In this chapter a tool for assessing the Security and QoS tradeoff in heterogeneous networks (SQT) is defined, taking as the basis, a set of well defined security and QoS parameters and their relationships, expressed as part of a CPRM. SQT is built based on that we consider to be the key requirements for analysing the Security and QoS tradeoff in future networks. This tool is considered as a handler for CPRM-based systems, and in this chapter the steps for developing similar tools to SQT are provided. In our case, a prototype built in MATLAB is provided. This prototype is not directly applicable to all the environments, but that we consider is very useful to understand the basic usability of SQT.

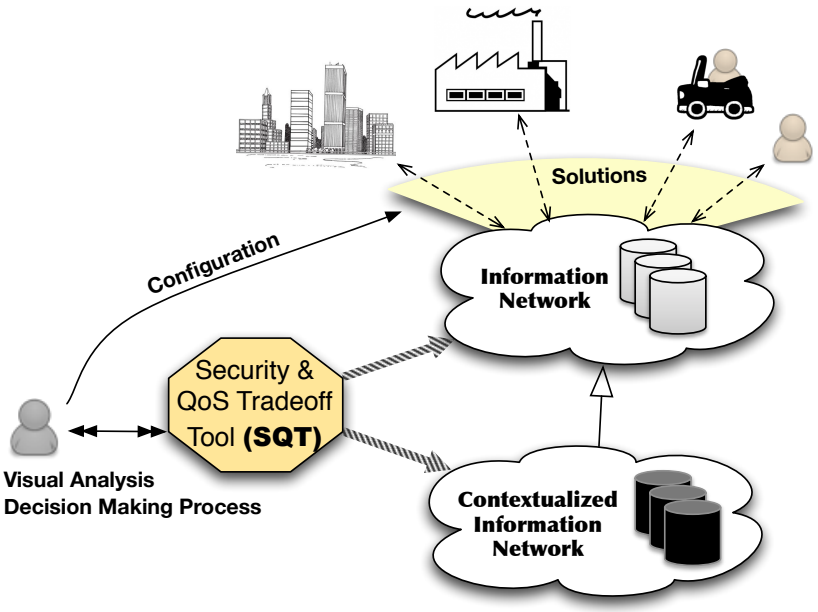


Figure 4.1: Security and QoS Tradeoff Tool.

Figure 4.1 shows the idea behind SQT is developed. SQT takes advantage of the multiple information and data collected on the environment by the different devices, users and sub-environments (e.g. buildings). This information usually can be contextualized, as for example in Smart Cities scenarios [116], where the coexistence of different type of devices using multiple interfaces will be a reality. Therefore, when the information is contextualized, deriving usefull information about security and QoS tradeoffs for the analysis based on the test fields in the environment is feasible. So, the final aims are (1) to provide a model to translate the description of a system to an easy and understandable languaje to be procesed by a generic tool capable for integrating, dynamically, the new information collected by the information system (parameters, requirements, statistical results on the performance of a given solution in devices), and (2) to be able for taking decissions with the information taken from the information system, and show a visual representation of said data.

For example, Wireless Sensor Networks (WSN) can be considered as critical infrastructures, because they are very limited by the resources and capabilities of the smallest sensors. The aim of this solution is also to provide a valuable estimation on the impact that security

mechanisms have on this type of devices to improve the decissions on their configuration. Indeed, how to protect resource-constrained devices without severely affecting the lifetime is key when these are used as part of the backbone of the CPSs. CPSs are used to collect the information about the physical environement, something that is native using sensors. When new techniques open the door to collect and contextualize this data [117], the processing of the information gathered from these kind of environments can be very useful to the deployment of solutions in knowleadge-based heterogeneous scenarios [118].

4.2. Arquitecture

As a starting point, Figure 4.2 shows the components in a CPRM-based system, as well as a simplification of the process followed in order to obtain a $CPRM_i$ (Figure 4.2 (b)). $CPRM_i$ represents the final behaviour of the system, in which all the mechanisms and technologies that are relevant are finally chosen by the administrator and taken into account when extracting relevant information of the model. The user/administrator sets the contexts using the GUI (Figure 4.5). Note that, the PRM defines all the initial components which will be inherited by the CPRM and the $CPRM_i$, except the contexts, that are only defined to be in the CPRM or the $CPRM_i$. In particular, a CPRM only has one GC, while a $CPRM_i$ has one GC, and, as minimum, one PC.

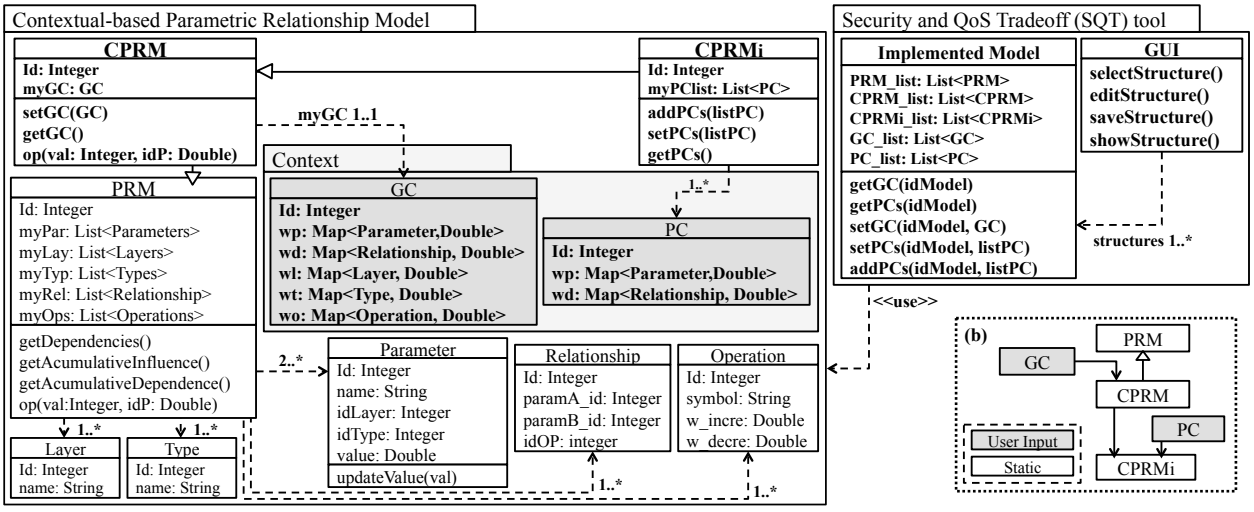


Figure 4.2: Contextual-based parametric relationship model classes and behaviour (b).

The design of SQT is based on components, from the arquitectural point of view (Figures 4.2, 4.3 and 4.4), as well as from the integration of contexts, considered as interchangeable components (Figure 3.15). So, considering that the same CPRM can have only one GC, and as many PCs as needed, any model structure can be enhanced using SQT, by amending or replacing the GC, or any/all of the PCs integrated in it.

Also, in order to facilitate the analysis (perform comparisons and so on), it is necessary to be able to return to the previous version of the model, for example, by removing the last context added, or even build new contexts, by removing any of the integrated contexts (not necessarily the last one added). Moreover, these changes should not affect the definition of

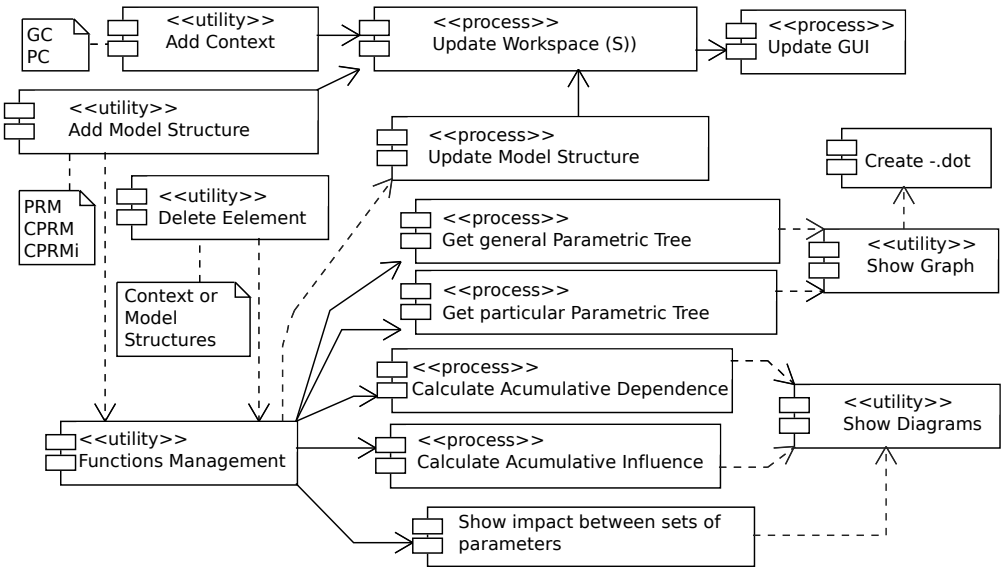


Figure 4.3: Components diagram.

the model itself; the model has to be consistent. So, following the rules, and ARs defined in Chapter 3.2, the multiple integration of interchangeable contexts, consistently, is guaranteed. In addition, the integration chain and the data structures also need to be defined, following the set of criteria that will be described in this chapter.

To provide the functionality of interchangeability, taking contexts as components, the integration of contexts is performed in SQT based on the diagram shown in Figure 4.4. Specifically, the activity diagram shows the creation of a $CPRM_i$ from a PRM. To do that, at each step, the intermediate structure necessary to complete the model definition is built. Note that this procedure is defined in such a way, so as to build a default model, used as the example to be modified. In general, the GC or PC are previously defined, not randomly generated.

Moreover, as our aim is that any context can be extracted from an existing model, the functions used to generate contexts to be used as examples, also are able to extract contexts when the input received is adequate. This is the case for the functions getGC and getPC, that are used to extract (if it exists) or generate (if it doesn't exist) a context based on a given parametric model structure. This behaviour depends on the input given to these functions. In the case of getGC, when the input is a PRM, that is, a model without a preassigned context, then, the function generates a default GC for the parameters in the PRM [1]. Otherwise, if the input is a CPRM or a $CPRM_i$, lets say, any structure with a GC preassigned, then, the GC of the structure will be returned.

Similarly, getPC only returns the PCs assigned to a structure when they are defined, a condition that only a $CPRM_i$ may satisfy. Therefore, when getPC receives any other model structure, it returns a new, randomly generated PC, in accordance with the parameters given in the model. Intuitively, the parameters *instance* will be fictitious, only for testing purposes[2].

[1] After that, the GC can be applied to the PRM, generating a new CPRM, as Figure 4.4 shows.

[2] M and P indicate the number of parameters of type *instance* that will be generated for each parameter

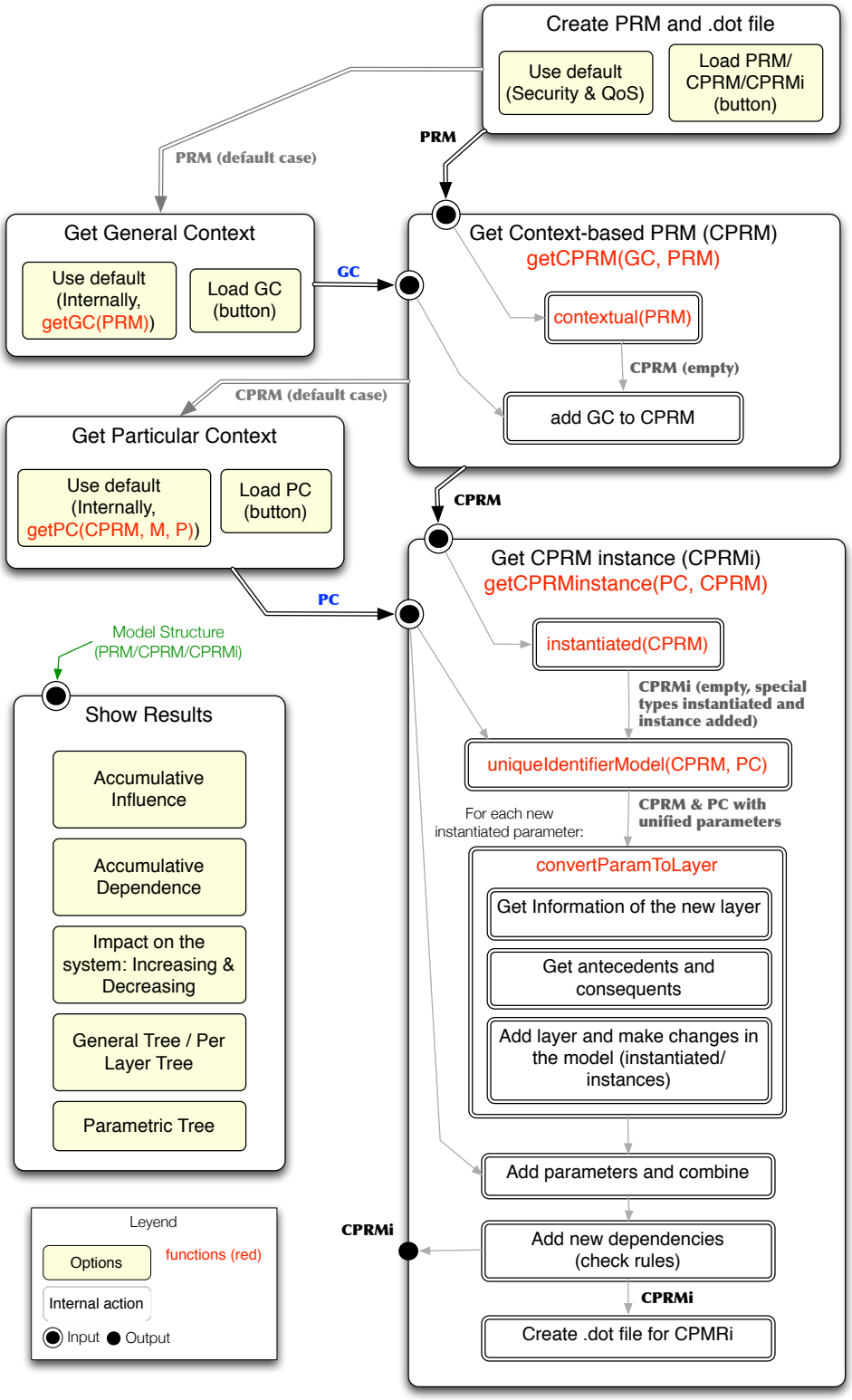


Figure 4.4: Activity flow.

of the model given as input.

Then, the next step is the assignation of contexts to models, which is done by the functions getCPRM and getCPRMinstance. First, getCPRM generates a new CPRM, based on the GC and PRM/CPRM provided. In order to assign the GC to the model, the latter is previously converted to a contextual-based model, if it is not one already. Second, the function getCPRMinstance generates a new $CPRM_i$, given a PC and a PRM/CPRM/$CPRM_i$. In this case, when the model given as input is not an instantiated model (not a $CPRM_i$), then, the model is converted to a contextual-based model. After that, it is converted to make it compatible with an instantiated model. Finally, the PC is integrated in the model. However, if getCPRMinstance gets a $CPRM_i$ as input, the process followed is more complex, since various PCs can coexist in the same $CPRM_i$. So, although the main objective is the same, to get a new structure given a context and a model, getCPRMinstance requires a much more detailed analysis than getCPRM.

Note that it is expected that an instance of a CPRM, that is, a $CPRM_i$, will be more dynamic than a CPRM. It is also motivated because, if the system is well defined, the PCs should be interchanged more frequently than a GC, that, although it can be modified, it is assumed that will be the most stable definition of context. Moreover, when the GC is retrieved, the PCs (if exist) are also removed from the model, because they were set up based on the existence of a GC.

It can be considered that the integration of new PCs into a $CPRM_i$ is the most delicate step. For example, the new PC defines parameters that, being different from those in the $CPRM_i$, can be taken as the same set of parameters and this has to be taken into account. This scenario could occur, given that the same parameter can be part of several PCs. For this reason, defining the parameters based on a set of identificators (instead of only one) it is very important, in order to perform the mapping between parameters during the integration step, thus avoiding the overlapping between different types of parameters that could be taken to be the same one. This is only an example of the issues that have to be considered when integrating a PC into a $CPRM_i$. These issues are managed by SQT, using the function getCPRMinstance.

This function, given a CPRM, builds the skeleton of a $CPRM_i$ structure (using *instantiated*), adding the new types and additional fields (Table 4.1) required to satisfy the properties of the model. After that, it completes the model by adding the information in the CPRM, and, finally, it integrates the new PC in the model. If the function receives a $CPRM_i$ as input, then, it is only necessary to perform the final step of integrating the new PC in the $CPRM_i$. Moreover, this last step is the most important and the most complicated, because it is responsible for avoiding the overlapping of parameters and the additional issues related to the integration of parameters. If this step is not properly achieved, then, probably, the context will not be retrieved once it has been integrated, or, even worse, the final model may be incoherent.

Therefore, as Figure 4.4 shows, the steps after the function *instantiated* are critical. After this point, the unification of parameters, that is, the mapping, is done, considering the definition of the parameters given in the model. Then, once the model has been unified, getCPRMinstance identifies the parameters that will be instantiated, and generates new layers for them. In other words, this means selecting those parameters p (instantiated), so that there are parameters defined in the PC which instantiates p (instances), lets say $p|\exists p2 \in PC, p \in P(p2)$. In order to do this, the function convertParamToLayer is used,

Table 4.1: Fields for Data Structures in Model Schemes

Row,Column: Purpose	PRM definition	CPRM (changes regarding PRM)	$CPRM_i$ (changes regarding CPRM)
1,1-2: Info. layers	Number of layers (NL) + Properties of layers	Adds the property *weight of the layer* w_l.	Defines special layers for the parameters instantiated (parents).
2,1-2: Info. types	Number of types (NT) + Properties of types	Adds the property *weight of the type* w_t.	Adds two special types: *instance* and *instantiated*
3,1-2: Info. operations	Number of operations (NO) + Properties of operations	Adds the property *weight of the operation* w_o.	-
4,1-2: Additional information	Directory by Default (DD)	-	Adds information on the number of PCs integrated.
5,1: NP	Depends on the model.		
5,2: NProp	5	6	6
5:(5+NP),1-NProp: Parameters	Properties of parameters	Adds the property *weight of the parameter* w_p.	The parameters instantiated changes its layer by the new one created as a consequence of the instantiation.
6+NP,1: Dependencies	Dependencies before being processed (DB) or Matrix of dependencies processed (MD)		
6+NP,2-3: Once the dependencies have been processed	Matrix of zeros NPxNP + DB	Matrix of weights NPxNP + DB	-

to return the definition of the new layer. These layers store all the information needed so that in the case that the PC is removed, the model will be able to return to its previous behaviour, prior to the instantiation using the PC.

The last step in the integration is to determine the relationships that will be inherited by the instance of a parameter, and, moreover, generate those new dependencies required for maintaining the coherence in the model. This step is done based on Table 3.8, as defined in Chapter 3.2. After that, the $CPRM_i$ is complete, and can be exported to a *.dot* file, that is interpreted as a graph using GraphViz.

Finally, regarding the analysis, all tests that are possible to do on a PRM are possible on a CPRM or a $CPRM_i$. The difference is, that while a PRM is static, a CPRM also presents a subjective view of the context of the network, giving more relevance to parameters, relationships or transactions in accordance with the management priorities or deep knowledge of the network.

In addition, a $CPRM_i$ enables the integration of dynamic contexts, based on the specific information of the current state of the network. These final contexts, the PCs, are not only more variable and fleeting contexts than the GCs, they are also more specific. Once the set of devices that will form the network is known, so as to recognize its dependencies and set up non-subjective values, but also the closest to reality, parts of the GC can then to be instantiated using the PC.

4.3. Components in the Model

SQT handles two type of structures or schemes, given as scripts: data model structures (PRM, CPRM, $CPRM_i$), and contextual structures (GC, PC). Both are provided separately, because the context can be extracted from the first ones to be integrated in the second ones.

In the following the specification chosen in SQT for handling these components is detailed.

4.3.1. Data Model Structures

As we can deduce from the previous sections, any structure PRM, CPRM, $CPRM_i$, PC or GC, has a predefined format in which they are created and used. SQT holds all these structures as part of a general structure, S, that can be saved, as workspace. So, from the implementation point of view, it is possible to assume that S (Exp. 4.1) represents the data model, composed by the model structures and the context structures.

$$S = \{D1, D2, D3, D4, D5\}; \tag{4.1}$$
$$D1 = \#prm, nxtID, \{\{prm1, id, info, file\}, ...\}; \tag{4.2}$$
$$D2 = \#cprm, nxtID, \{\{cprm1, id, info, gcid, file\}, ...\}; \tag{4.3}$$
$$D3 = \#cprm_i, nxtID, \{\{cpmi1, id, info, gcid, pclist, file\}, ...\}; \tag{4.4}$$
$$pclist = [pcid1, pcid2, ...]; \tag{4.5}$$
$$D4 = \#gc, nxtID, \{\{gc1, gcid, info, file\}, ...\}; \tag{4.6}$$
$$D5 = \#pc, nxtID, \{\{pc1, gcid1, info, file\}, ...\}; \tag{4.7}$$

In Table 4.1 the physical differences between the three types of model schemes are shown. Based on these discordances, SQT can identify when a model structure is a PRM, a CPRM or a $CPRM_i$, and then, manages the operations defined according to the type of structure and its definition. Moreover, the commonalities between the models, makes the component-based integration feasible.

Note that, any element in a PRM has to be identified by two identifiers minimum: the unique identifier value (numerical value, $id\_$), and the name of the element (string). Both identifiers are present in the properties of the elements, that is, in the properties of layers, types, operations and parameters. Moreover, the properties of the elements also identify the visual representation of the elements in the Matlab diagrams, or in GraphViz (color and shape).

Moreover, once the particular dependencies of a parameter with the rest have been calculated, this matrix, specific to the parameter, denoted as *Parametric Map* (PM), is stored as a property of the parameter. This matrix is recursively generated, and defines all the relationships where the parameter is involved (that is, a NPxNP matrix). The drawback is that these PMs (one per parameter) may require a large amount of space in the data structure; which is the price to be paid for not having to re-calulate the PMs more than once. The PMs, are calculated based on the *Parametric Table* (PT), defined in Chapter 3.1, where all the direct relationships between parameters are shown. So, if $\exists a, b \in PRM | d(a, b)$, then, $PT(a, b) > 0$. Initially, this knowledge is expressed throught *Brute Dependencies* (DB), $A \xrightarrow{op} B$, using the numeric identifiers. So, for example, $A \xrightarrow{op} B$ is expressed as id_A, id_op, id_B, and, in a contextual-based model, the weight of the dependence is included: $id\_A, id\_op, id\_B, w_{d(A,B)}$.

Finally, it is important to remark, that the PRM provides the basis for building the CPRM models and the $CPRM_i$ instances (or instantiated models). However, there are differences in the data structure which poses huge changes in the calculation process achieved

by SQT. So, while the CPRM structure represents a turning point between a PRM and an instantiated model, the really significant changes are found in the definition of the $CPRM_i$ structure. This is due to two key factors: the definition of the special types *instance* and *instantiated*, and the conversion of instantiated parameters to layers. These factors, coupled with the ability of the model to modify or restore itself, by means of the elimination and the aggregation of contexts, are a big change from the non-instantiated models, which are relegated to a more static function.

4.3.2. Context Structures

The definition of specific data structures for GCs and PCs is necessary in order to manage them as interchangeable components. Two skeletons for GC and PC are provided using Exp. 4.8-4.14 and Exp.4.15-4.17, respectively.

Some fields in the context structures, GC and PC, are the same as those defined in the model structures, PRM, CPRM and $CPRM_i$. This is necessary, because the context structures will be part of the contextual model structures, and, therefore, a common part between them will be required to match them, like pieces of a puzzle. Indeed, the context structures are intended to change the properties in the elements of the model (parameters, relationships, types ...), so, at least, they have to be able to identify these elements and the new values.

$$GC(1,1:2) = \{NL\{id\_nivel1\ weight1;\ id\_nivel2\ weight2; ...\}\} \tag{4.8}$$
$$GC(2,1:2) = \{NT\{id\_tipo1\ weight1;\ id\_tipo2\ weight2; ...\}\} \tag{4.9}$$
$$GC(3,1:2) = \{NO\{id\_op1\ weight1;\ id\_op2\ weight2; ...\}\} \tag{4.10}$$
$$GC(4) = \{\}; \tag{4.11}$$
$$GC(5,1:2) = \{NP, NProp\}; \tag{4.12}$$
$$GC(6:(5+NP),1:NProp) = \{id\_param1\ weight1; ...\} \tag{4.13}$$
$$GC(6+NP,1:2) = \{ND, \{id\_dep1\ weight1; ...\}\} \tag{4.14}$$

However, in the GC and the PC, the fields NProp and NP refer to the context structure itself. A context structure defines its own extension. For example, in the current version, in a PC, the number of properties for a parameter, is equal to five ($NProp = 5$): the list of identificators of the parents of the parameter ($id\_Parents$), the identificator of the parameter (that can be modified if the AR are applied), the name of the parameter, and the weight (w_p).

$$PC(1,1:4) = \{NP, Nprop, ND, \{IDpc, descrip.\}\}; \tag{4.15}$$
$$PC(2:(1+NP),1:Nprop) = \{id\_Parents, id, name, weight\}; \tag{4.16}$$
$$PC\{3+NP\} = \{id\_ParamA, idOp, id\_ParamB, weight; ...\}; \tag{4.17}$$

Given a PC, when it is integrated in a CPRM, the parameter $p \in PC$, will be of type *instance*, and its type and layer are inherited from its parents, which take the type *instantiated*. In both cases, these modifications are marked with the identifier of the PC ($IDpc$ in Exp. 4.15) which introduces the changes.

4.4. Correlation based on a Common Set of Parameters

The structure given by C_{NP,MAX_c} stores all the parameters that affect the same set of parameters as another given parameter. We denote this relationship as correlation of parameters based on the influence of a common set of parameters. The correlation based on a common set of parameters allows identifying the maximum number of parameters that are affected simultaneously by the modification of a set of parameters.

$$C_{NP,MAX_c} = \left\{ \begin{array}{ccc} \left\{ \begin{array}{cc} \{pa1_{1,1}\} & [pr1_1, ...pr1_n] \\ \vdots & \vdots \\ \{pa1_{m,1}\} & [pr_{m_1}, ...pr_{m_q}] \end{array} \right\} & cp_{1,2} \cdots & cp_{1,MAX_c} \\ cp_{2,1} & cp_{2,2} \cdots & cp_{2,MAX_c} \\ \vdots & \vdots \ddots & \vdots \\ cp_{NP,1} & cp_{NP,2} \cdots & cp_{NP,MAX_c} \end{array} \right\}$$

We define the correlation degree based on the size of the set of parameters that a group of parameters have in common. For example, the position $cp_{1,1}$, brocken down in C_{NP,MAX_c}, shows the set of parameters with correlation degree equal to 1 given the parameter with identifier 1. Thus, $\{p_{a1}\}$ represents any parameter which is affected by the increasing/decreasing of the parameter with id 1, and, according to C_{NP,MAX_c}, this parameter is also affected by the increasing/decreasing of the parameters with identifiers in the list $[p_{r1_1}, ...p_{r1_n}]$.

$$cp_{NP,MAX_c} = \left\{ \begin{array}{cc} \{pa_{NP_{1,1}}, ..., pa_{NP_{1,MAX_c}}\} & [pr1_1, ...pr1_s] \\ \vdots & \vdots \\ \{pa_{NP_{z,1}}, ..., pa_{NP_{z,MAX_c}}\} & [pr_{z_1}, ...pr_{z_t}] \end{array} \right\}$$

Note that the correlation degree represents the size of the sets $\{p_a\}$. For example, the information of correlation for the parameter with identifier NP, given the maximum correlation degree is cp_{NP,MAX_c}. This structure, stores sets of type $\{p_a, ...\}$ with size MAX_c.

4.5. Graphical User Interface for Administration

The first prototype of SQT provides a *Graphical User Interface* (GUI) for administration purposes, designed to show all the options available to the user in a single window, divided into seven sections or panels (Figure 4.5):

1. PRM Panel. Load a PRM from an ".m" file, or load a default PRM. It is possible to load a CPRM or a $CPRM_i$, because both of them are PRM (but a PRM is not a CPRM).
2. Panel for operating with GCs. Intended to select GCs and model structures that allow a GC to be added, or the GC to be extracted.
3. Panel for operating with PCs. Intended to select PCs and model structures that allow adding the PCs, or extracting them.

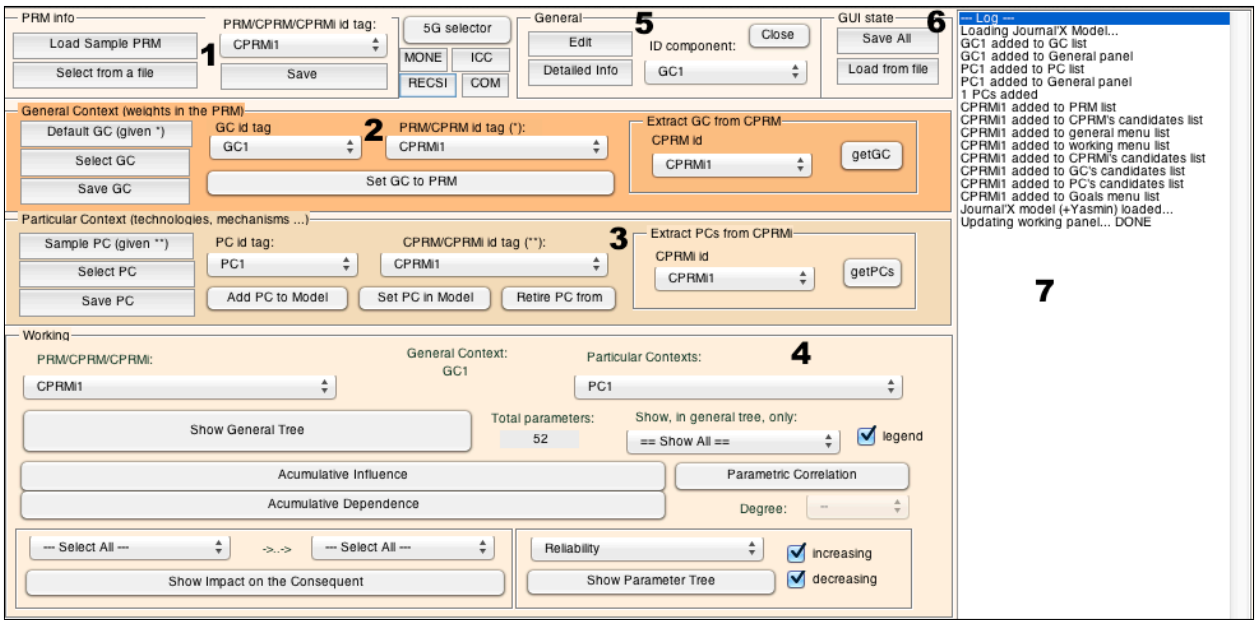


Figure 4.5: GUI for administration.

4. Working panel. This panel can be used once the first PRM has been loaded. Includes the operations defined by the model, that are shown in Figure 4.3.
5. General Panel. Intended to act on any of the elements defined in the workspace (S). Through this panel, it is possible to close or delete any element in S.
6. State Panel. Intended to save the workspace (all the context and model structures in S), as well as to load an entire workspace, that had been previously saved in a ".m" file.
7. Informative Panel. Shows information about the operations performed using the GUI, and any errors that may have occurs.

It is important to note that any model structure can be saved in a new file with extension ".m". So, the user can modify an existing example (e.g. default sample) so it builds its own models and contexts. The functionality for saving the default models and contexts, is aimed at simplifying the learning process of the user so they can use SQT.

4.6. Evaluation of Dynamic Instantiation

The aim in this section is raise awareness of the impact of a PC based on the location of the instantiated parameters in the dependencies graph. When the PC defines a type of sensor or device, it is usual that the parameters involved will be leaves or parameters with low accumulative dependence. In this case, the parameters in the consequent are instantiated parameters whose impact on the rest of parameters is null. Hence, the impact of the relationships defined for these parameters, when they are set up in our model do not provide much information as regards from where there are extracted. So, the impact of a new PC is greater, as the new parameters defined (instances) affect parameters far away from the leaf nodes, and with a high number of relationships.

Table 4.2: Set-based definitions in a CPRM-based system.

Acumulative Influence (ι) and Acumulative Dependence (δ)	
$\iota(a) = \|I_a\|, I_a = \{x \| x \to a \vee xRa, x \neq a, x \in P\}$	(3.21)
$\delta(a) = \|D_a\|, D_a = \{y \| a \to y \vee aRy, y \neq a, y \in P\}$	(3.22)
$xRy \iff x \to y \vee \exists k \| k \in D_x \wedge k \in I_y$	(3.23)

This conclusion can be analysed using the formulation in Table 4.2, which is taken from Chapter 3.1. Considering N parameters in a CPRM, and Y the parameter that will be instantiated by K number of instances, which means that there are k parameters which satisfy that Y is their parent: $K = |\{x|Y \in P(x)\}|$. This set is known as the set of instances of a parameter Y, and is denoted as H in Exp. 4.25. Moreover, P defines the set of parents of a parameter, which is the information provided by the PC. Whether Y is a leaf node, that is, $\delta(Y) == 0$, and the accumulative influence on Y preceeding the instantiation of the CPRM is $\iota(Y) = M$, then the new accumulative influence on Y after the instantiation is $\iota(Y) = K * M$, considering that Y is the only instantiated parameter.

Moreover, if the instantiated parameter Y is not a leaf node, it impacts on other parameters in the PRM ($\delta(Y) > 0$). So, in the case that prior the instantiation the accumulative dependence on Y was $\delta(Y)$, and that $\delta(Y)^t$ defines the dependence degree of Y once the instances 1 to $t <= K$ have been added, and therefore $\delta(Y)^0 = \delta(Y)$, after the instantiation the accumulative dependence on Y is given by Exp. 4.24 (δ'):

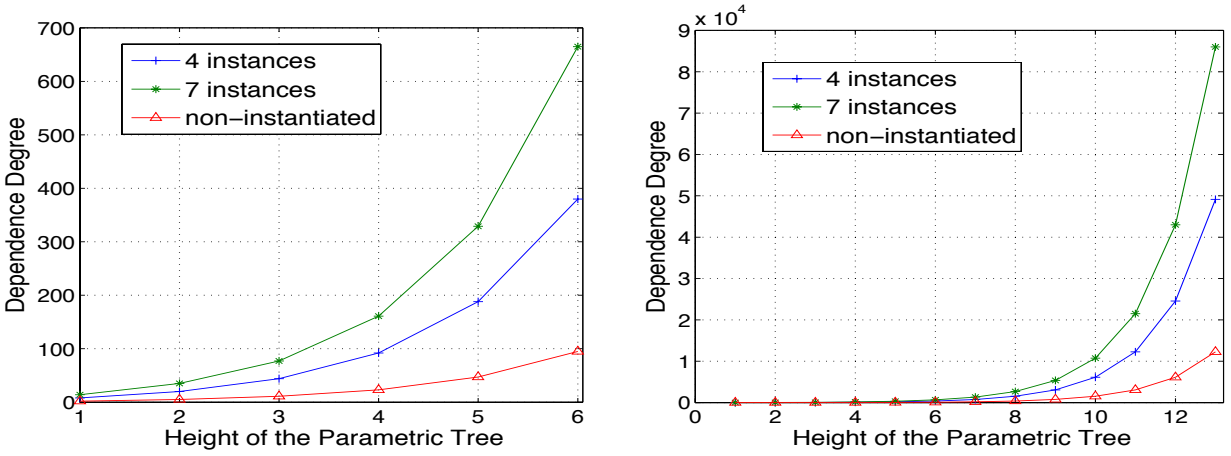
$$\delta(Y)' = \delta(Y) + \sum_{i=1}^{K} (\delta(y_i) - \delta(Y)^{i-1}) \tag{4.24}$$

$$H = \{x|Y \in P(x)\}, K = |H| \tag{4.25}$$

Therefore, the new accumulative dependence is calculated based on the new dependencies that are included because the instances can define new relationships, so new parameters can be affected. Therefore, the complexity when a new PC is added depends on the number of parameters and relationships but also the location of the instantiated parameters in the general parametric tree.

Figure 4.6 shows the increasing number of dependencies (dependence degree) for a parameter regarding the length of the branch, considering 6 as the length of the longest branch in Figure 4.6(a) and 13 in 4.6(b). The problem is simplified considering that the new instances only inherit the relationships defined by their parents, and do not define new relationships, and that the number of dependencies per parameter is fixed at 2.

Considering these restrictions, note that, although the instances do not define new dependencies, the number of dependencies for the parameters at upper layers increases. According to Table 4.2, the accumulative dependence is higher in those parameters far away from the leaves. Specifically, it depends on the position of the parameter and the number of dependencies behind it. When the parameter is instantiated, the accumulative dependence increases if new parameters appear in the parametric tree as a result of the new information provided in the instantiation.



(a) Branch: 1 (leaf) to 6.

(b) Branch: 1 (leaf) to 14.

Figure 4.6: Instantiation of one parameter

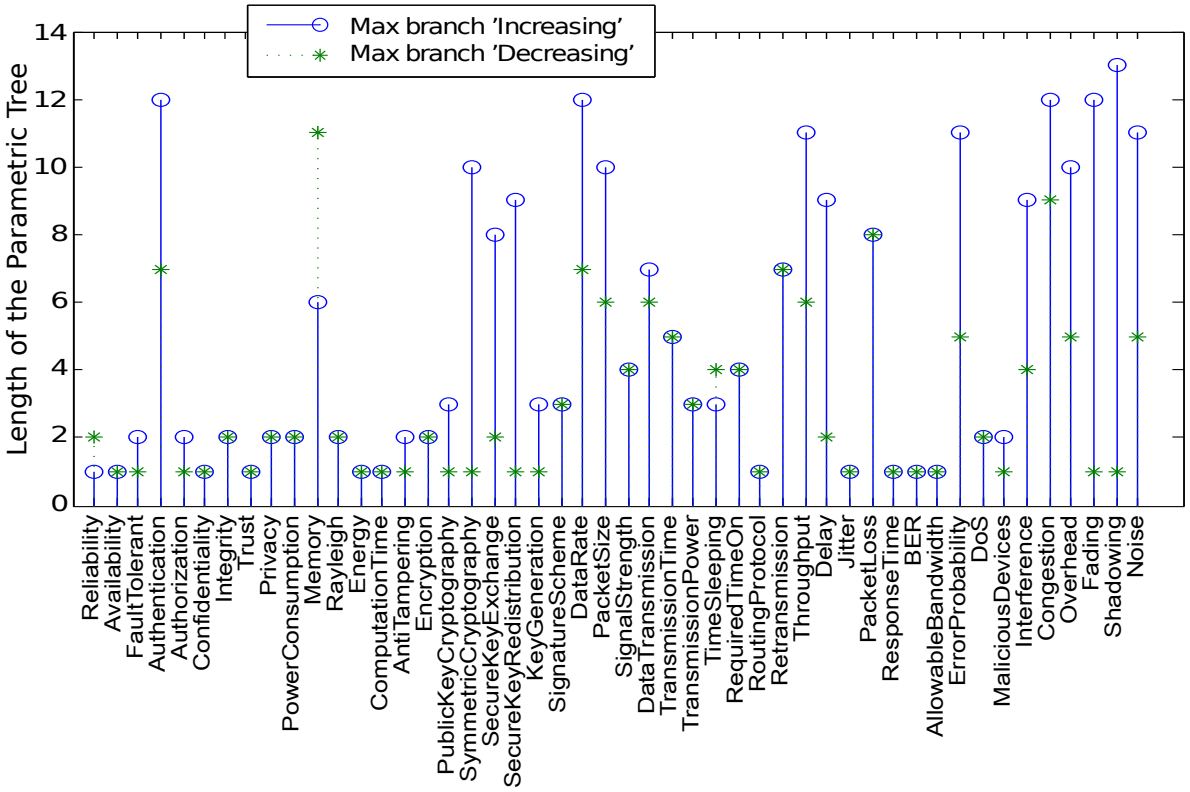


Figure 4.7: Example: Length of parametric trees in a CPRM-based system.

The values chosen to show the previous results have been chosen only for testing purposes. Indeed, in a PRM, the number of dependencies, parameters and instances is free. For example, Figure 4.7, shows the length of the parametric trees in a PRM with real parameters used for testing, where the longest branch may vary considerably, depending on the parameter, precisely because there is no fixed limit for the number of relationships. Moreover,

in Figure 4.7, two types of results are shown: the results for the increasing and decreasing parameters may generate different parametric trees according to the formulation in Table 3.7 and the weights. If a dependence is weighted 0, then the effect of the parameter that is in the antecedent of the dependence is not propagated by the tree.

Finally, the accumulative dependence degree depends on the number of dependencies defined for the system, and the instantiation may add new dependencies not defined by the parents. It must be remembered that Figure 4.6 has been built, taking into account that the instances do not define new relationships. When new relationships are defined in the PC, which is generally the case, the impact on the final accumulative dependence increases. Moreover, it must be appreciated that, even if the instances only inherit the behaviour of their parents and do not include new dependencies, the calculation of the impact on the model of the instantiated parameters takes more time, because the process is repeated per instance.

For example, when the accumulative dependence increases, the complexity of the integration of parameters carried out by a handler of CPRM-based systems, is higher. An intuitive conclusion is that a possible way to reduce the time of integration of PCs, is to select the PCs which describe the physical layer components from the very beginning, because, in general, the physical description of components has a low dependency degree, when it concentrates on, for example, the characteristics of the battery. These components are usually instantiated with instances that are affected by other components (appear as consequents in the formulation).

However, given the nature of our model, this is not always the best choice, because new relationships, that is, new behaviour, defined in PCs may completely change the behaviour of the final $CPRM_i$.

In other words, if the instances with new relationships, not defined in the model for the instantiated parameter, are not integrated at the begining, and the instantiated parameter has a large number of instances, then, the new relationships have to be added to all the instances. The cost for this is very high, because it implies triggering action rules to maintain the coherence in the model, so, the whole parametric system is checked again when some incoherence is found. If the instances with new relationships are added at the begining of the instantiation process, then, this new behaviour is taken by the parents which transfer the sum of the whole behaviour learned (the parent's behaviour and the new relationships) to the instances. Note that this criteria concerns the order in which the instances in the PC are to be set up in the model, so it can be done independently from the order in the selection of PCs or parameters to be instantiated in the CPRM.

As a result, the instantiation process may be enhanced to mitigate the adverse effects which impact on performance, considering:

R1. Dependency degree and accumulative dependence of parameters to be instantiated.
R2. The effect of the changes on the rest of the system (length of the max. parametric tree).
R3. The order of the new instances and relationships defined in PCs, because the new behaviour may totally change the final $CPRMi$.

4.6.1. Classifications and Mitigations

In this section, the focus is to provide a classification based on the information in the CPRM-based system and the type of the PC that will be integrated in order to adapt the integration process, to mitigate the effect on the performance because of the dynamic instantiation. Some conclusions that can be drawn from this analysis are sumarised in Table 4.6, where order means the priority in the instantiation process.

Table 4.3: Recommendations in the Integration Process.

Component	Characteristic	Order
CPRM	Parameters with shortest branch first (SBF).	1
CPRM	Parameters with max numer of instances.	2
PC	Parameters that define new behaviour: new relationships to be added that maximises max δ'.	1
PC	Parameters that maximises the final longest branch (if available).	2

In order to test these conclusions, the CPRM-based system described in Table 4.4 is processed taking into account Table 4.3, to mitigate the effect in the integration of the PC, that is also structured according to the recommendations in Table 4.3[3]. Note that the only relationships that are marked with a weight are those defined by the instances to make changes in the behaviour of the model.

Table 4.4: CPRM and PC definitions.

Parameter	Direct Relationships	Branch	Instances	New Relationships
A	3: B,C,E	6	A1, A2	-
B	0	1	B1, B2, B3	$B3 \rightarrow F1$
C	0	1	C1, C2, C3, C4	$C4 \rightarrow F1$
D	3: B,C,E	8	D1	-
E	2: F,I	7	E1, E2	$E1 \rightarrow C3$, $E1 \rightarrow B2$
F	1: H	3	F1	-
G	1: H	2	G1, G2, G3	$G1 \xrightarrow{w=0} H1$, $G1 \xrightarrow{w=3} H2$
H	0 -	1	H1, H2	-
I	2: J,K	9	I1	$I1 \xrightarrow{w=2} J1$, $I1 \xrightarrow{w=0} K2$
J	1: L	8	J1,J2	-
K	0 1	1	K1,K2,K3,K4	-
L	1: M	7	L1,L2	$L2 \xrightarrow{w=4} M2$
M	2: J,A	6	M1,M2	$M1 \xrightarrow{w=2} J1$
N	1: B	2	N1	$N1 \rightarrow A$, $N1 \xrightarrow{w=2} B3$

[3]Information about default weights omited for sake of clarity. It is assumed that the relationships that are completely new have weight 1.

For example, I is related to J and K according to Table 4.4. Then, the instances of I inherit the relationships of I, which means that they will be related to J and K, and, therefore, with their instances. However, the relationship $I1 \xrightarrow{w=0} K2$ redefines the weight in the relationship inherited to avoid the relationship between the instances I1 and K2. These relationships do not add additional relationships to the model, because the parents of the instances are already related to each other, so it is unnecessary to apply action rules. Therefore, the relationships that add new information in the model are $B3 \rightarrow F1$, $C4 \rightarrow F1$, $E1 \rightarrow C3$, $E1 \rightarrow B2$, and $N1 \rightarrow A$. These require, respectively, the relationships $B \rightarrow F$, $C \rightarrow F$, $E \rightarrow C, B$ and $N \rightarrow A$ to be included.

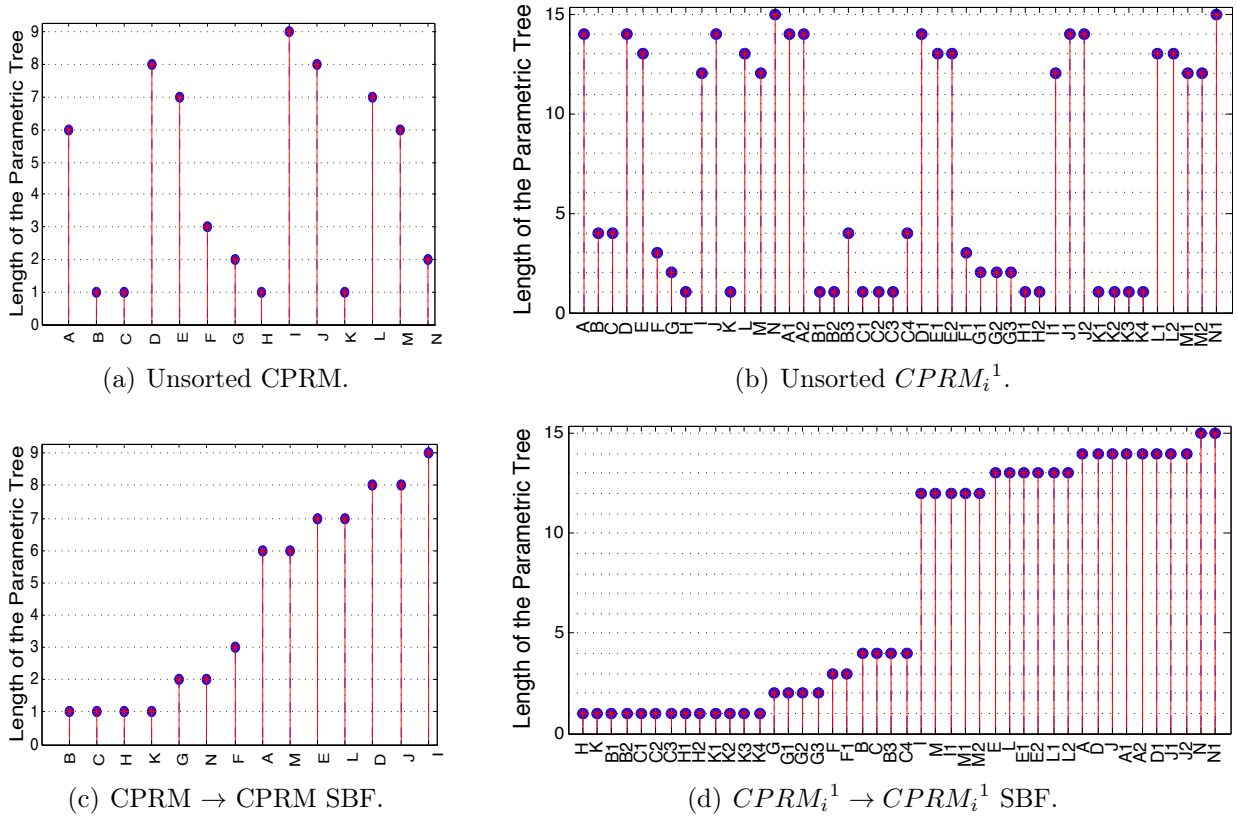
Consider that the order $A - N$ corresponds to the unsorted or default distribution where letters describe non-instantiated parameters and the instances for each parameter are described using the letter of the parent and a number. Note that one of the recommendations in Table 4.3 suggest the *Shortest Branch First* (SBF) criterion, that is the opposite to the *Longest Branch First* (LBF) criterion. Using the CPRM and the PC, 9 $CPRM_i$ are generated:

- $CPRM_i{}^1$: Unsorted CPRM. Unsorted PC.
- $CPRM_i{}^2$: Unsorted CPRM, PC SBF.
- $CPRM_i{}^3$: Unsorted CPRM, PC LBF.
- $CPRM_i{}^4$: CPRM LBF, Unsorted PC.
- $CPRM_i{}^5$: CPRM LBF, PC SBF.
- $CPRM_i{}^6$: CPRM LBF, PC LBF.
- $CPRM_i{}^7$: CPRM SBF, Unsorted PC.
- $CPRM_i{}^8$: CPRM SBF, PC SBF.
- $CPRM_i{}^9$: CPRM SBF, PC LBF.

Figure 4.8 shows the length of the branches of the original CPRM (before the instantiation process) and $CPRM_i^1$, before and after the instantiation. Note that a simple ordering in the CPRM before the instantiation is not sufficient, because the PC adds changes in the behaviour of the model that have to be considered. Indeed, the order of the parents (parameters of type *instantiated*) before and after the instantiation is not the same. For example, in Figure 4.8(c), in a CPRM SBF it is recommended C before G. However, as can be seen in Figure 4.8(d), the instance $C4$ introduces more changes than the instances of G. So, this information cannot be considered if only the parameters in the CPRM are taken into account. Specifically, Figure 4.8(d) has been built from Figure 4.8(b), but this order is different from the final order that can be generated following the recommendations in Table 4.3, they may vary depending on the recommendations.

In order to check the different alternatives and the suitability of the recommendations, the instantiation process (add CPRM to PC to generate $CPRM_i$) is repeated 60 times per $CPRM_i$, and the average execution times for these processes are shown in Figure 4.9[4]. Note that the CPRM and PC chosen as seeds, integrate diverse relationships, defined only in order to force the maximum number of operations to action rules.

[4]Results calculated using MATLAB and our handler of CPRM systems for Security and QoS Tradeoffs (SQT). During the process, the handler plots different graphs and information for testing. Note that this additional funcionality increases the overall time in all the integrations.



(a) Unsorted CPRM.

(b) Unsorted $CPRM_i{}^1$.

(c) CPRM $\rightarrow$ CPRM SBF.

(d) $CPRM_i{}^1 \rightarrow CPRM_i{}^1$ SBF.

Figure 4.8: Changes in the parametric trees after the instantiation

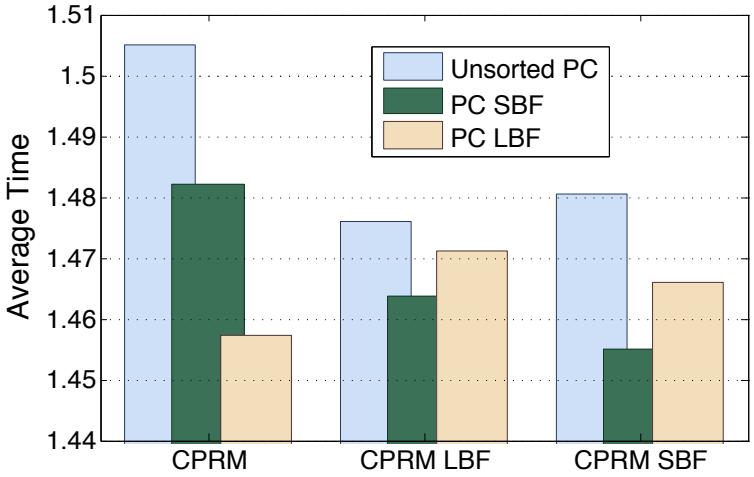


Figure 4.9: Average times in $CPRM_i{}^{1-9}$ calculation.

Considering the results, when restrictions in Table 4.3 are applied to the CPRM before the instantiation, the performance (measured in terms of computation time) is improved with respect to the case where the CPRM is not sorted, or is sorted acording to LBF. Moreover, in this case, the PC does not integrate a high number of relationships. Despite this, given the results, it is possible to appreciate that the advantages for sorting the parameters in the

PC are conditioned by the max branch of the parameter that will be instantiated and the number of new relationships that the parameter introduces in the CPRM, and, therefore, the length of the new branches that have been added as a consequence of the instantiation process. Moreover, as Figure 4.8(c) and Figure 4.8(d) show, the order in CPRM LBF and PC LBF or CPRM SBF and PC SBF is not the same, because the order in the PC is generated considering the new information that will be integrated in the model. It is the reason because CPRM LBF + PC LBF is not optimal, instead CPRM LBF + PC SBF. Note that, in the absence of información (case CPRM), the integration of the PC LBF is the best option, in this case.

It can be observed that the effect when sorting the parameters (instances) in the PC is higher than the effect in sorting the parameters in the CPRM. This is precisely because the integration process is costly, and, especially so, when the instances and relationships define a behaviour that is very different from the behaviour defined by the parents. It is expected that the new instances inherit most of the behaviour of their parents, so in these cases maybe a better definition of the properties of the system to be instantiated, or a new classification in different layers of the parametes would be very helpful to mitigate the effect of the instantiation process. The knowledge of the system can be very helpful in order to identify the order of the parameters in a PC, because, when the environment is very dynamic, it is not always possible to define the adequate order in the integration of PCs or the order of their componets.

Finally, CPRM schemes are used to identify Security and QoS tradeoffs. In Chapter 2, a set of parameters and relationships which define general behaviour of security and QoS parameters extracted from a wide range of studies and mathematical formulation is provided. With this purpose, the original scheme used to compose these systems, classifies the parameters based on their type and abstract layers. The classification of types includes: security (e.g. authentication), performance (e.g. delay), QoS (e.g. streamming), resources (e.g. battery), characteristics (e.g. type of antenna), *Quality of Experience* (e.g. mechanisms for measuring the user's experience), etc.

The objective therefore, is to have different PCs and instantiate the CPRM when the information about the final composition of the system becomes available. Given the previous results, and our classification, the integration of PCs should be performed as follows, based on the dynamisms of the parameters and our focus[5]:

1. High-layer relationships. Security and QoS requirements are described at this layer. So, given the nature of our analysis, the parameters in this layer should be instantiated at the end of the process, because, probably, they change frequently.
2. Local properties. At this layer the resources in the devices of the network and the characteristics are described. For example, this layer describes physical characteristics of the components. So, at this layer there are several parameters that are leave in many parametric trees.
3. Communication and Measurements layers. The most of the parameters defined at these layers are of type performance. Most of these parameters are defined based on mathematical formulation, and are influenced by other parameters at the same layer or from

[5]These are conceptual / abstract layers, not physical layers.

other layers. It is expected that the instances for the parameters at this layer do not include additional relationships that affect the behaviour of the model. It is expected that the new changes redefine the weight to some parameters and relationships.

4. Environmental conditions. The effect of the environment in the system is described at this layer. As these are restrictions given by the environment, some of these parameters may not vary for a long time (e.g. the probability of wireless eavesdropping in a wired system; average of devices in an office, etc.). These parameters depend on the dynamism of the system.

Moreover, the preferences in the selection/integration of PCs, should be adapted based on all information about the system where this mechanism will be used. In general, this solution may be useful when the system provides a rich variety of information that is stored in data bases and may be defined using parametric relationships. Then, the integration / extraction of contexts may help to understand the different problems in the integration of security and QoS mechanisms in the environment, before their deployment.

4.7. Summary and Final Remarks

In this chapter the steps for implementing a handler of CPRM-based systems have been detailed. This has resulted in a tool for assessing the Security and QoS tradeoff (SQT) in the *Future Internet* (FI). This tool is independent of the applications to be evaluated, because it is based on a CPRM, which considers the properties/configurations taken by the applications or systems to perform the parametric relationship analysis. So, it is not necessary to change the specification of the tools measured or adjusted. Instead, a rich background of the system to be measured is required. In this approach, we have assumed that the final administrator knows enough about the system to take advantage of SQT. Moreover, the tool provides an initial set of parameters and relationships to perform the Security and QoS tradeoff in an FI environment, and also offers the possibility of modifying these parameters and the entire definition of the components, as needed.

Finally, the impact on the performance when dynamic information is agregated to the model is evaluated considering the formulation given in Chapter 3.2, and the final implementation of SQT. The results show that it is possible to mitigate the impact during the intantiation process when the particularities of the environment are considered together with the behaviour defined to CPRM.

The instantiation consists in providing mechanisms for general parameters, which are defined in the model. It is a fact that when the number of parameters increases, so does the complexity of the CPRM-based system. Therefore, the complexity in CPRM-based systems can be analised using the size of the parametric tree defined for the parameters to be instantiated. In this last section of the chapter, the impact on the influence and the dependence degree based on the position of the instantiated parameter in the dependencies tree has been discussed, and alternatives to mitigate this impact are proposed and analysed.

Note that, at this point, there is an open issue. Indeed, the dependence degree affects the visualisation of the data handled by these models, because the number of relationships increases the complexity of the final diagram. Therefore, despite the problems of having a

large number of dependencies, the benefit is that, the more dependencies and parameters there are, the more information there is available to extract useful information about the security and QoS tradeoff. For this reason, the next chapter defines a recommendation system for extracting information from CPRM-based models where large numbers of parameters coexist.

CHAPTER 5

CPRM-based Recommendation System for SQT

In this chapter the steps necessary to develop a recommendation system for SQT (SQT-RS) are detailed. The recommendation system is developed considering the properties and definitions given to CPRM-based systems. Moreover, the aim is to provide recommendations dynamically, based on the behaviour defined in the CPRM-based system at a given time. To satisfy this objective, a set of steps for extracting the information from CPRM is defined, generating dynamic facts, that are the inputs for the recommendation system. Finally, SQT-RS is integrated in SQT (Chapter 4), and a discussion about the cost on generating the dynamic facts based on the CPRM selected is provided.

5.1. Overcoming the Limitations in SQT

SQT implements a CPRM-based systems handler and provides samples of CPRM-based systems, based on a pre-defined set of parameters and relationships defined at a high-layer of abstraction, focusing on Security and QoS tradeoffs. That means that SQT depends on the set of parameters chosen to operate with the model, and, thus, on the pre-defined behaviour based on the current literature. This has proved to be useful from a research point of view, at a high layer, for example, in the use case of instantiation of Authentication mechanisms in WSNs.

However, the current version of SQT can be difficult or impractical for untrained users. The reason for this, is that the final results provided by SQT (graphs) have to be carefully analysed prior to making a decision, and the growing number of parameters complicates the analysis which, therefore, requires much more time. To make SQT useful for users, it needs to be adapted to provide real-time recommendations based on goals and the current state of the model.

Some improvements are possible by adding new functionality in SQT, in order to set up requirements, and provide recommendations. Specifically, in this chapter:

- The concepts of *requirement*, *goal* and *recommendation* for CPRM-based systems are defined.
- *Facts* and *rules* to perform the inference process to identify the best configuration or recommendations given the requirements and goals are defined.
- An example of the generation of facts and recommendations given a predefined set of Security and QoS parameters is discussed.
- The effect of the contextualised parameters on the final numer of facts generated by SQT-RS is analysed.

To implement these characteristics, in this chapter an expert system based on CLIPS is defined and integrated inside SQT. The information in CPRM-based systems (parameters and relationships), and the results inferred from them (impact and influence of parameters) are processed to produce the facts needed for the expert system to work. The rules are defined and implemented in accordance with the properties of CPRM-based systems.

Similar approaches for providing recommendations considering Security and QoS are tend to focus on service composition, as is the case of [119] and [120], or on providing recommendations but to be implemented in devices, as in [13]. In [121] a friendly tool to simplify the decisions of the user when selecting security goals is proposed. However, these approaches have been developed for services, and do not provide dynamic recommendations based on heterogeneous contexts formed by different *things*.

5.2. Prior Formulation to be Considered

The SQT tool is based on the definition of a CPRM-based system, where a set of operations on the parameters of the model are explained. Said formulation is needed to define the recommendation system, because the operations on the parameters help us to determine the

final set of recommendations. In the following sections, the terminology in Table 5.1 will be used. The greater part of this formulation has been taken from Chapter 3.

Table 5.1: Recursive operations in a CPRM-based system.

Acumulative Influence (ι) and Acumulative Dependence (δ)	
$\iota(a) = \lvert I_a \rvert, I_a = \{x \lvert x \rightarrow a \vee xRa, x \neq a, x \in P\}$	(3.21)
$\delta(a) = \lvert D_a \rvert, D_a = \{y \lvert a \rightarrow y \vee aRy, y \neq a, y \in P\}$	(3.22)
$xRy \iff x \rightarrow y \vee \exists k \lvert k \in D_x \wedge k \in I_y$	(3.23)
Impact Increasing (Δ) and Decreasing (∇) a Parameter x	
$\Delta x \implies \forall y \lvert xRy, v(y) = v(y) + w_T \wedge u(y, w_T)$	(3.27)
$\nabla x \implies \forall y \lvert xRy, v(y) = v(y) + w_T \wedge u(y, w_T)$	(3.28)
$u(x,\omega) = \begin{cases} \Delta x & \text{if } \omega > 0; \\ \nabla x & \text{if } \omega < 0; \end{cases}$	(3.26)
${A_{D_o p}}^p = \bigcup Dep(k)_{op'} \lvert k \in D_p, op, op' \in \{\Delta, \nabla\}$	(5.1)

The *Accumulative Influence* (ι) and the *Acumulative Dependence* (δ) are functions to be applied on a parameter $a \in CPRM$. They are based on respectively, the cardinality of the sets I_a (Exp. 3.21) and D_a (Exp. 3.22). I_a groups those parameters that are related with a, being a the consequent in the relationship, while D_a is formed by the parameters that are related to a, being a the antecedent in the relationship. So, these parameters *depend on* a. In both cases, I and D, the parameters are related directly $x \rightarrow y$, or indirectly through intermediary parameters k (Exp. 3.23).

Moreover, the impact produced by increasing (Δ) or decreasing (∇) a parameter can be measured by Exp. (3.27)-(3.28). Note that the impact of the operations (Δ, ∇), will be propagated through all the parameters affected by the dependencies. Note that this recursive effect is produced by Exp. (3.26), based on the value of ω. These steps are explained in more detail in Chapter 3. For what follows, it is enough to understand that w_T depends on the type of relationship defined between the parameters in the dependence (+,-,...). We will come back to this point when defining the recommendation set.

Finally, our tool SQT implements the CPRM model using matrixes to work with the relationships. So, the total number of dependencies generated by the increasing/decreasing of a parameter can be summarised using the *accumulative matrix of dependencies* ${A_{D_o p}}^p$ (Exp. 5.1). This matrix is calculated as the union of the dependencies generated by the recursive increasing and decreasing of the parameters in the parametric tree. When the value of a parameter p changes, the effect of the change is propagated throughout the rest of the parameters which depend on p, and this influence is stored in this matrix to derive information that will be used, for example, to easily identify the conflicts between parameters. Thus, the matrix Dep_{op} reflects the direct dependencies enabled for a parameter when the operation op (increasing/decreasing) is carried out.

5.3. Approaches for Deployment

In this section, two questions are discussed. First, how it is possible to use SQT in a real system and, second, how developers can use SQT in combination with their own tools. Considering these questions, there are two possible alternatives to help developers use SQT in a real system: the SQT daemon and the application collaboration approach, depicted, respectively, in Figures 5.1 and 5.2.

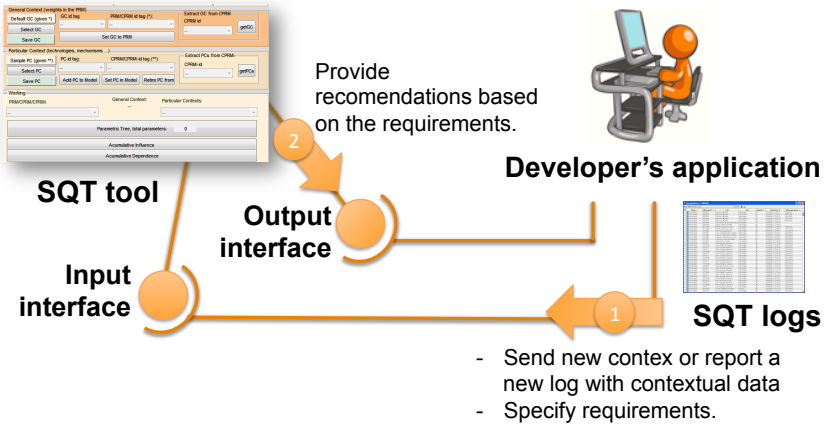


Figure 5.1: Application Collaboration deployment.

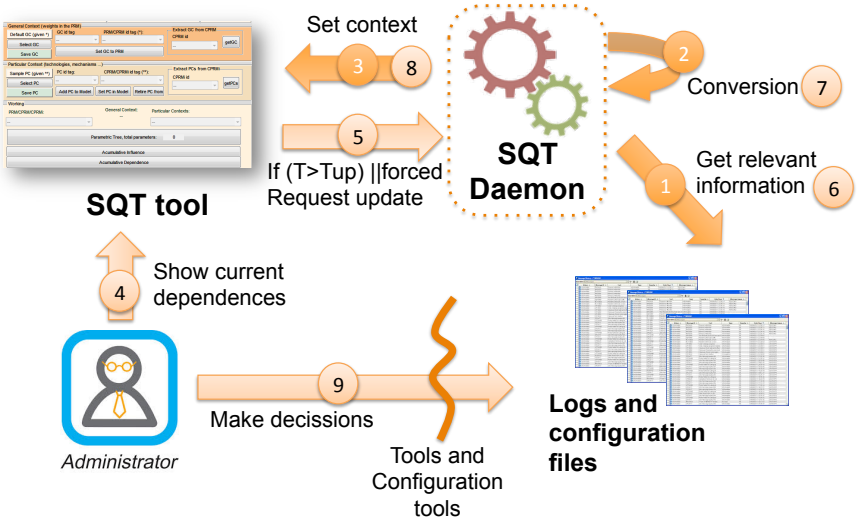


Figure 5.2: SQT daemon deployment.

The application collaboration approach understands the syntax of SQT and sends defined files to be analysed. Then, SQT gives the answer, i.e. the recommendations, to the developer's application. The intuitive problem with this solution, is that the developer's application has to be adapted to collaborate with SQT. Therefore, the complexity depends on the flexibility of the application to be adapted in order to receive feedback from SQT

and, of course, whether or not the main functionality of the application is unaffected by this integration.

The alternative approach, is to use an SQT daemon, i.e. a service which investigates the user's equipment and detects the configuration files, taking the parameters from them. This approach is more complex and can even be intrusive. In general, the steps for the latter approach are as follows:

1. Extract parameters from the configuration files.
2. Generate models based on the parameters.
3. Show these models to the administrator, who decides possible changes.
4. Regularly consult these files to check on the changes.

Note that the extraction of the parameters from a configuration file depends on the configuration file itself. So, as not all configuration files follow a schema, this extraction process can be very difficult to apply in a real scenario. Indeed, the requirements needed to deploy SQT change depending on the selected approach.

5.3.1. Discussion

The main requirements for deploying the SQT daemon, are based on the dynamic extraction of parameters from the configuration files and logs in the system. This functionality, requires:

R1. Identifying how configuration files are built.
R2. Identifying how logs are built.
R3. Defining the concepts of *requirement* and *recommendation*.

Moreover, the prototype of an SQT daemon has to be built, based on a specific platform, because understanding the whole set of configuration files significantly increases the complexity of the solution, and the approach should be clarified in order to define how to extend this prototype to other platforms (if possible). Indeed, the concepts of requirement and recommendation depend on the actions that must be taken in the final application. As a result, it is very difficult to apply this approach without increasing the complexity of the system.

Unlike the SQT daemon deployment, the requirements for Application Collaboration seem more achievable:

R1. Define the concepts of requirement and recommendation.
R2. Define SQT logs.
R3. Define SQT interfaces with applications.
R4. Proceed to developer collaboration.

The developer needs to collaborate, and adapt their applications in order to generate the SQT logs, and also to be able to incorporate the recommendations received from SQT.

We note the major difficulty in this approach is whether or not the developer is able to map the syntax used in SQT with the parameters in their applications, in other words, understand how a CPRM-based system works. Moreover, defining the SQT logs and the SQT

interfaces with applications, may be trivial once the concepts of requirement and recommendation have been defined. This definition has to be done before deploying any approach. In this chapter a recommendation system based on requirements and goals introduced by the user is defined and implemented.

The SQT models can be stored, edited by the application, and then loaded again. Moreover, various interfaces have been defined to assist in the input of requirements and goals. These interfaces are graphical, as we believe that a graphical representation is useful for a large number of users, because it helps clarify how CPRM-based systems work. Furthermore, automatic responses may have adverse effects on the final system. So, in our prototype, the interfaces for the input of requirements are provided as GUIs, and the results are also stored as files so they can be managed by external applications when needed.

5.4. CPRM-based Structures in SQT-RS

In this section we present the concepts of goal, requirement and recommendation according with the definition of CPRM, and how they are used in SQT-RS.

5.4.1. Goals and Requirements

Goals and requirements are closely related to the user. SQT-RS provides a GUI for selecting goals and requirements (Figure 5.3), and, internally handles these values using two structures: *GOA* and *REQ* (Exp. 5.8-5.11). The information in these structures is used to provide general descriptions, about the number of elements (goals/requirements) and, after that, the complete information of the goals and requirements requested by the user.

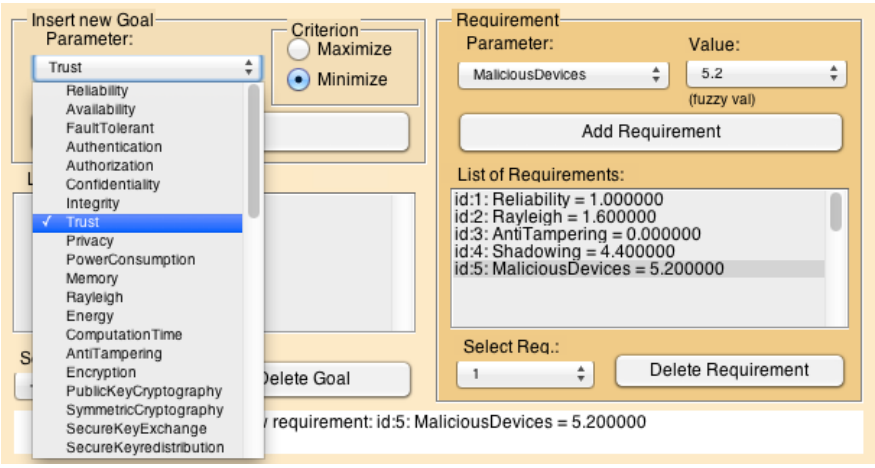


Figure 5.3: GUI for requirements and goals defined by the user.

The set of parameters shown in the GUI depends on the $CPRM_{id}$ selected in SQT, so any goal g_K is defined for a specific $CPRM_i$ at a given time, and are classified based on their identifier (id), an objective parameter given by its identifier (P_{id}), the objective or criterion to be applied, and a list of recommendations to satisfy the goal (S_{id}), that initially is set to

null. Moreover, the list of conflicts identified for S_{id} is included in C_{id}. Both, S_{id} and C_{id} are uploaded into SQT-RS after the execution of the inference process in CLIPS.

$$GOA = \{\{\#G, CPRM_{id}, \#Rec\}; g_1; ...; g_{\#G}\}; \tag{5.8}$$

$$g_k = \{id, P_{id}, objective, S_{id}, C_{id}\}; \tag{5.9}$$

In the case of the requirements, the requirements selected by the user are forced in the $CPRM_{id}$ target, modifiying the values for the parameters. For this reason, the requirements do not store information about the recommendations. Rather, the requirements req_k are described using an identifier (id), the id of the parameter and the value taken by the parameter (val).

$$REQ = \{\{\#Req, CPRM_{id}\}; req_1; ...; req_{\#Req}\}; \tag{5.10}$$

$$req_k = \{id, Parameter_{id}, val\}; \tag{5.11}$$

The objectives criteria are the maximisation (max) or minimisation (min) of the values of a parameter. For example, based on the classification of parameters in CPRM, parameters of type *consequence* or *performance* are good candidates to be considered as part of a goal. In a 5G Green environment, some parameters that may be chosen are *outage probability*, *signal strength*, *energy*, *secrecy capability* and *secrecy rate* between others. However, SQT-RS allows the selection of any parameter in the model as objective goal.

5.4.2. Recommendation

A recommendation is described by the set of parameters and the operations to be performed in order to satisfy the goals requested by the user in the previous step. The mathematical formulation for recommendations in SQT-RS, defined based on one or more goals, is shown in Table 5.2. Specifically, the formulation for considering only one goal and one recommendation set as output (Exp. 5.12-5.13, 5.16-5.17, and 5.19), several outputs or different recommendation sets (Exp. 5.14, 5.18, 5.20-5.21), and multiple goals (Exp. 5.15), is given.

The latter set of expressions for defining sets of recommendations usually occur when satisfying multiple goals is not possible simultaneously, or different combinations with different weights can be applied. For this reason, the recommendations in a recommendation set S are ordered depending on the final impact on P, such that Exp. (5.20) is satisfied.

The following definitions are used in Table 5.2:

- $id1...idN$ are identifiers of parameters in CPRM.
- op_i is the type of operation through which the objective is satisfied, while op_{jP} means that from the results of op_j, only the results for P are considered.
- $\square$: the goal can be achieved by applying either operation Δ or ∇.

Note that Exp. (5.20) shows a property based on Exp. (5.16)-(5.17). So, when the recommendation is built considering Exp. (5.16)-(5.17), the final set of recommendations is composed by parameters that belongs to the influence set of P (I_P) by definition (Exp. 3.21).

Table 5.2: Formulation for Recommendations.

Generic Definitions		
Goal:$g \in \{max, min\}, g :: CPRM \rightarrow [0,1]$		(5.12)
Recommendation:$R = \{id1_{op_1}, ...idN_{op_N}\}$		(5.13)
Recommendations Set:$S_g(P) = \{R_1, ..., R_k\}$		(5.14)
Multiple objectives:$RS = \{S_{g1}(P_{id1}), ...S_{gq}(Pidq)\}$		(5.15)
Goal, $g(P)$	Recommendation	
$max(P)$	$R\|id(x_j) = idj, x_j \in I_P, op_j(x_j) \rightarrow \Delta P$	(5.16)
$min(P)$	$R\|id(x_j) = idj, x_j \in I_P, op_j(x_j) \rightarrow \nabla P$	(5.17)
$\Theta(R) = sum_{j=1}^{N} op_{jP}(x_j)\|idj_{op_j} \in R, op_j \in \{\Delta, \nabla, \square\}$		(5.18)
$\square x \Rightarrow Itdoesnotmatter \Delta xor \nabla x$		(5.19)
Prop. $R_i \in S_g(P), idj_{op_j} \in R_i, op_j(x_j) \rightarrow g(P)$		(5.20)
Prop. $S_g(P) = \{R_1, R_2\} \Leftrightarrow \Theta(R_1) > \Theta(R_2)$		(5.21)

Moreover, for multiple goals, it is probable that a single recommendation set S is unable to satisfy all the objectives and so, multiple recommendation sets can be provided, where the final aim for each set is to satisfy a subset of the overall set of goals. These types of multi-objective problems are very complex to solve and, depending on the number of parameters can take very long time to get one answer.

According to the definition of a CPRM-based system, the information in a $CPRM_i$ is enhanced when more mechanisms are integrated in the model, that is, when more parameters of type *instance* are inserted in the model. In particular, if all the parameters are of type *instantiated* or *instance*, the model is completely instantiated, and the final recommendation shows a configuration of services, mechanisms or tools.

Therefore, SQT-RS considers this last point, and, in order to optimise the generation of recommendations, the instantiated parameters are treated, based on their instances. That means that the parameters of type *instantiated* are not shown, instead, their instances are processed. Moreover, SQT-RS only processes the information of those parameters that are related with one or more objectives. An additional benefit this adds is that it minimises the information stored, and the processing time.

Consequently, the final recommendation depends on the best configuration of parameters which not only satisfies the list of objectives defined by the user, but also depends on the instance of parameters, for example, the specific mechanisms available in the system that provide the requirements.

Moreover, it must be noted that a recommendation is an output of the system, while the goals and recommendations are inputs. Based on these outputs or recommendations, the user receives feedback, and can change the inputs or the composition of the model. Therefore, the way in which this output is perceived by the user is crucial.

It is important to understand that the parameter S_{id} in the goal (Exp. 5.8) is updated once the recommendation process has ended, that is, once the recommendation sets have been calculated using CLIPS. In the same way that CLIPS returns the results as strings, these may

be provided as MATLAB expressions to be directly loaded into SQT-RS to automatically show some results. We have used this feature to show the new state of the CPRM once the recommendations have been given. Therefore, the user can accept the changes suggested by the recommendation system and update the new state of the CPRM in SQT-RS without additional effort.

5.4.3. Facts and Rules

In the following sections, how to convert the information from the CPRM/$CPRM_i$ into facts is discussed, and the result of the operation on the CPRM parameters inside rules is detailed. The complete sequence used to build the facts and rules to be used by the expert system from a CPRM, is shown in Figure 5.4.

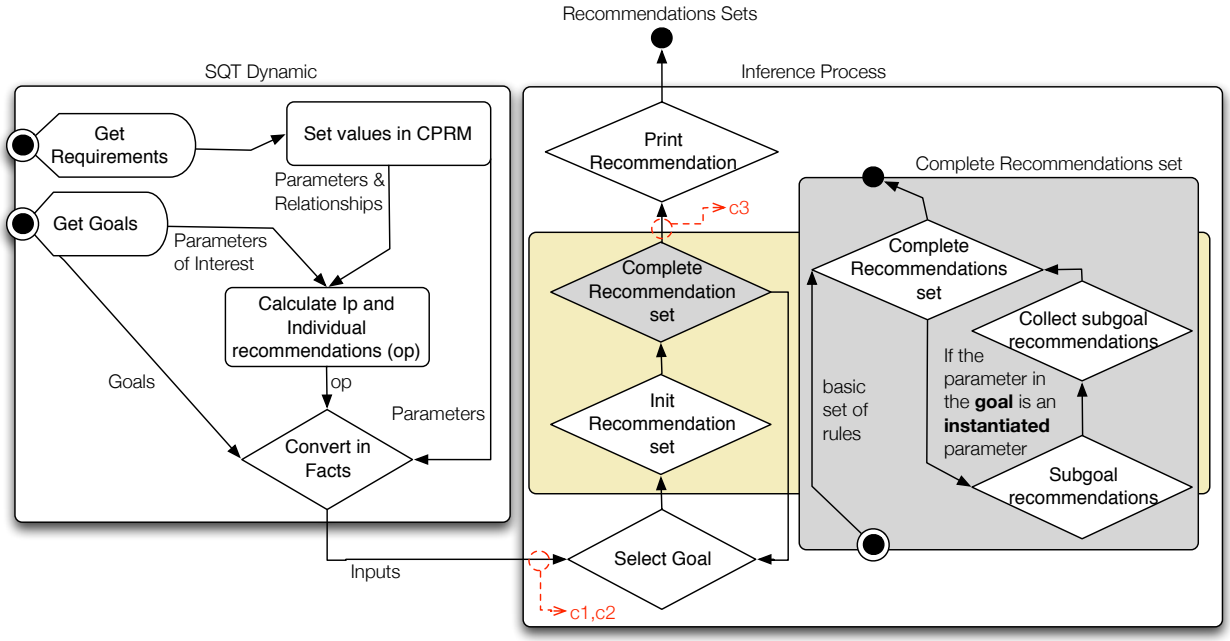


Figure 5.4: Recommendation chain.

5.4.3.1. CPRM-based facts

The list of facts generated by SQT-RS represents the current state of the model. This is built considering those parameters and relevant information extracted from the model which is related to the goals and preferences selected by the user. So, the list of facts provided by SQT-RS to infer information in real-time, will be delivered from the requirements and goals selected by the user, which are values required and max/min parameters, and the current information in the CPRM/$CPRM_i$, which are parameters and relationships.

The process followed to build the list of facts is as follows:

i. The user selects the requirements for the parameters, and the goals.

ii. These requirements (values for the parameters) are set up in the model, so the parameters in the requirements change their values to those selected by the user.
iii. After that, the list of goals (parameters and target) is processed to identify those parameters that are relevant for the individual recommendations.
iv. Then, the individual recommendations, denoted as *op* are calculated based on Exp. (5.16)-(5.17).
v. Finally, the list of facts is formed by the definion of those parameters of relevance, goals and individual recommendations calculated based on the information in the model.

Note that following this approach, the requirements, although not considered as facts, do influence the final recommendation by changing the value of the parameters in the CPRM. Besides, unlike the requirements, the goals selected by the user are converted to facts without being processed by SQT-RS. The same occurs with the definition of parameters.

As a result, SQT-RS processes three types of input-facts, considered dynamic because they are generated when the preferences/inputs given by the user change the behaviour of a model: goals, individual recommendations (*op*, Exp. 5.16-5.17) and parameters. Goals are given by the user; defined, based on a criteria (maximisation or minimisation) and a parameter. Additionally, SQT-RS generates *subgoals* when the parameter in a goal is instantiated.

Furthermore, individual recommendations reflect the operation to be carried out on a parameter in order to satisfy the goal imposed by the user. The parameters are given based on the information in the model. Therefore, regardless of whether or not the user introduces new requirements or new contexts this information is static. The scripts in the model may contain non-contextual parameters (non-instantiated and non-instance), or parameters with type instantiated or instance, that are contextual parameters.

5.4.3.1.1. Facts for identifying conflicts. In addition, the current version of SQT-RS is able to process conflicts between the individual recommendations. So, SQT-RS use special facts to identify these conflicts. In a first step, SQT-RS has been implemented to show simple conflicts between individual recommendations. This approach is very simple in practice and avoids direct conflicts without requiring much more memory and computation.

However, to provide a grain fine identification of conflicts, it is necessary to identify those parameters in the chain of an individual recommendation and identify the coincidences between the *internal operations* to achieve the goals.

For this reason, we define the internal operations as facts ($internal - op$) for identifying those intermediary parameters i, that effect the target/goal parameter g because of their relationship with a parameter a provided in an individual recommendation. In other words, $a \rightarrow ... \rightarrow i \rightarrow ... \rightarrow g$. Therefore, the intermediary parameters can be identified using the accumulative matrix of dependencies ${A_{D_{op}}}^a$ between the parameters in the CPRM chosen.

The drawback when including these operations is that the number of facts increases, and therefore the memory and processing time that SQT-RS needs to work also increase. However, the benefit is that it avoids adverse effects that are produced because the application of individual recommendations affect the same parameters in the chain in opposite ways.

Furthermore, in those cases in which both actions, increasing or decreasing a parameter, produce the opposite effect in the goal, the fact *avoid* is included to indicate that the rec-

ommendation that has been applied is the least damaging to achieve the objective, but the modification of the parameter should be avoided because it affects the goal.

Finally, the dynamic input-facts are processed by the static rules defined for SQT-RS, and finally generate output-facts: recommendations based on goals. It is important to remark that, the individual recommendations *op* are based on the results for the increasing/decreasing of parameters something which provides the best results based on the goals, measured with four types of tests defined in CPRM: accumulative dependence, accumulative influence, and increasing or decreasing a specific parameter.

So, the CPRM-based rules, by using the previous input-facts, will provide recommendations based on the result of these operations, based on the set of parameters defined in the model and the values to be enhanced as required by the user. Once the user receives feedback from the tool, new values can be introduced and then the model generates new facts and the inference process starts again.

5.4.3.2. CPRM-based rules

The CPRM-based rules are static and never change. They are meant to satisfy the requirements detailed in the previous sections. So, the inference process is divided into four steps, addressed in three phases, detailed in Figure 5.4:

1. Selection of a goal. Repeated for as long as there are goals to be processed.
2. Calculation of the set of recommendations, given the goal. The set of recommendations for a goal has to take into account the type of parameter (contextual or non-contextual).
3. Calculation of conflicts. This could be performed before goal selection or after the calculation of the recommendations, depending on the types of conflicts considered.
4. Print results. All the results are printed at the end of the inference process.

While 1 and 4 are basic steps, and step 3 depends on the concept of conflict determined, the greatest processing time is in step 2. Indeed, due to the set of conflicts considered in SQT-RS, step 3 has to be done at the end of the identification of recommendations, before printing the results. Step 2 is considered as the second phase in Figure 5.4, where rules are applied based on the type of parameter to be considered. The simplest rules in this phase are those which consider non-contextual parameters. However, when an individual recommendation is provided by an instantiated parameter, the following property, satisfied by any CPRM-based model has to be considered:

Definition (Prop.1): If there is an individual recommendation for an instantiated parameter, that is $\exists op_j(x_j), x_j \in I_P, type(x_j) == instantiated | g(P) = 1$, then, there is a recommendation for each instance of this parameter.

This property is satisfied because of the coherence rules on which the construction of any CPRM model is based. Simplifying, as any parameter instance inherits their parent's relationships, which, by definition is an instantiated parameter, then, if x_j belongs to I_P, then, $\forall x, y, x \in P(y)$ [1], that is, x belongs to the set of parents of y, then, as y inherits the relationships of x, then if $x \in I_P$, then y belongs to I_P too.

[1]Not to be confused: the set of parents of y, $P(y)$, with the parameter P, in this case it is only a variable.

Note that Prop.1 is assumed when the recommendation system is considering an individual recommendation which consists of increasing/decreasing (that is, operating on) an instantiated parameter. This is different from those cases where the instantiated parameter is part of a goal. When the instantiated parameter is part of a goal, Prop.1 is irrelevant. In a CPRM model, as stated, all the parameters of type instance inherit the relationships of their parents, which, by definition, are instantiated parameters. But, all the instantiated parameters have to inherit relationships of their instances, when the model required has to be coherent. These new relationships inherited by the instantiated parameters have weight 0 so as not to interfer with the instances, as is thoroughly explained in Chapter3.2.

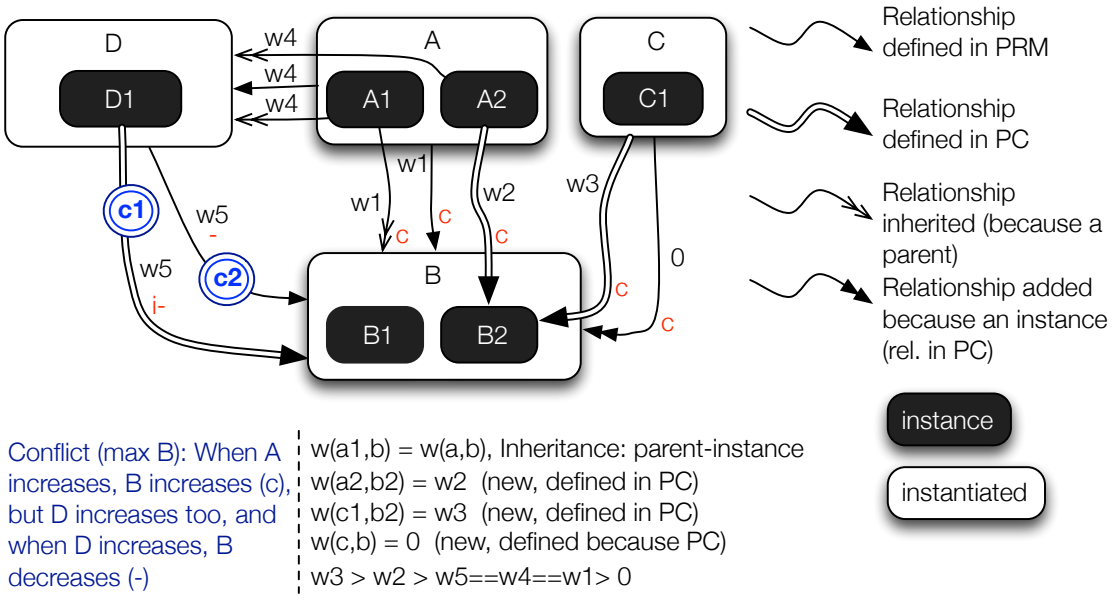


Figure 5.5: Example of inheritance relationships.

As a result of this, the inference process cannot interpret an instantiated parameter in a goal as a non-instantiated parameter, because the weights of the relationships for an instance are more specific than or equal to the weights defined by the parent for the same instance, and different weights provide different individual recommendations. Hence, the expert system, when an instantiated parameter is part of a goal to be satisfied, breaks this goal down into *subgoals* in order to consider all the instances of the parameter as goals to be satisfied, but within the context of maximising/minimising the instantiated parameter. This is necessary for identifying the instances that help to maximise the property/parameter. So, the following property is satisfied by our recommendation system:

Definition (Prop.2): A CPRM-based recommendation system considers the problem of maximising/minimising an instantiated parameter as the problem of maximising/minimising the instances of the parameter. That is: if $\exists g(p)|type(p) == instantiated$, then, $R_p = R_{p_i}|p \in P(p_i)$, with p and p_i parameters.

It must be observed, that while Prop.1 is satisfied by the properties of a CPRM, this new property, Prop.2, is implemented in the recommendation system, and so does not come from

the model behaviour. In order to clarify the relevance of this property we use the example depicted in Figure 5.5. Consider that the goal is in B (maximise or minimise). Then, the inference process will return recommendations about possible modifications in A1, A2, C1 and D1, in order to maximise B. However, without taking into account Prop.2, the final recommendation will consider the weights defined by these parameters with B, and not the weights defined with their instances.

If the goal is, for example, to maximise B, considering only A, B and C, if the relationship between C and B defines a possitive impact [2] (as is the case of the complete (c) relationship), then a good recommendation is to increase or provide the capability C1 in order to maximise the property B through the improvement of B2. In addition, if the goal $max(B)$ does not imply $max(B1)$, $max(B2)$, and, without considering D, then, the main recommendation will be to increase A1, and A2 and C1 will not be mentioned.

To satisfy Prop.2, the recommendation system must implement, in addition, the property, Prop.3:

Definition (Prop.3): The information about an instance is provided as a fact when any of its parents is considered in a goal or in an individual recommendation. In addition, if the parent is in a goal, then the individual recommendations for the instance are also provided.

This property is required so as to provide the inference process with the information about the instances for creating the subgoals, and infering information. Independently of Prop.3, the information about an instance can be provided as fact based on the steps detailed in Section 5.4.3.1.

The complete set of rules defined for SQT-RS are shown in Figure 5.6. A,B and C represent different states related with parameters: non-instantiated and not-instance (A), instantiated (B) and instances (C). In the built recommendation phase, when the parameter is instantiated, their instances are considered, and for this reason there are no specific subrules for this state. The subrules r31 and r32 of type B, manage objectives for instantiated parameters. Hence, in these cases, the final recommendation is based on maximising/minimising the instances of the instantiated parameter. These rules work under the assumption that Prop.2 and Prop.3 have been satisfied. In other words, for any goal defined for an instantiated parameter, there are facts defining the subgoals of the goal, and these subgoals are defined for the instances of the instantiated parameter. In addition, the individual recommendations for subgoals are added as facts too, based on Prop.3.

Finally, the rules are taken directly from a .clp file, and this file is completed with the dynamic facts generated by SQT-RS. For this reason, SQT-RS uses a .jar file, developed to assert, dynamically, the facts provided by SQT-RS, in a CLIPS environment where the rules and templates are defined using Jess [122]. The recommendation sets are stored in a temporary file and processed by SQT-RS in order to show the results to the final user. Moreover, the conflicts that avoid the satisfaction of multiple goals are collected by the rule

[2]The final effect depends on the symbol of the relationship. In this example, all the relationships are complete (c). That means that when the antecedent increases, the consequent increases, and when the antecedent decreases, the consequent decreases.

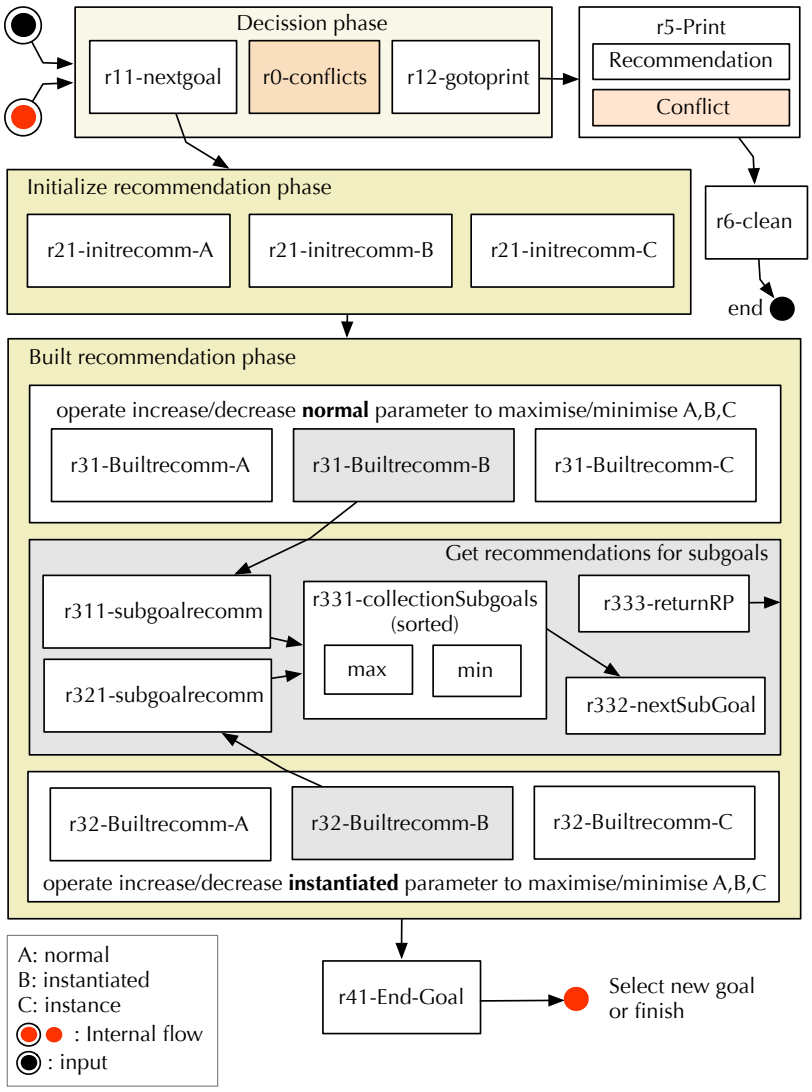


Figure 5.6: CLIPS rules and phases.

r0-conflicts and stored in an additional file that is shown to the user, using SQT-RS.

5.4.3.2.1. Conflicts considered in the rules. The meaning of conflict depends on the type of problem to be considered. In our implementation, we consider that the following conflicts are possible:

c1. When the modification of a parameter x, either increasing or decreasing, that is $\Box x$, avoids the goal.
c2. When there is an intermediary parameter i that requires increasing and also decreasing to achieve the goal (opposing operations).
c3. When different goals require opposing operations to satisfy their respective recommendations.

These conflicts are considered together, and, therefore, it is possible to print them at the same time as the recommendations are printed. The recognition of the conflicts based on the analysis of the attributes in simple facts (c1-c2) is available at the begining of the recommendation process, while others, being more complex, because they are based on the analysis of the final set of recommendations or multiple goals (c3), have to be analysed at the end of the decision process. For this reason, and because our definiton of rules allows it, the identification of conflicts is performed before the selection of the next goal to be processed when c1 and c2 are considered, and at the end of the decision process, just after the recommendations for all the goals are ready, when c3 has to be considered (multiple goals). As, at the end of the decision process, the number of facts is higher than at the beginning, it is useful to consider when the evaluation of c3 is not required.

Figure 5.5 shows an example of conflict because of c1 and c2. The conflict c2 occurs when the relationships between A and D and A and B occur. The complete relationship c implies that when A increases, then B increases and when A decreases, B also decreases. Therefore, to maximise B, the possible modifications in A have to be considered. However, when A increases, D also increases, and D has a negative relationship with B (-), which means that when D increases, B decreases. So, from the point of view of considering the modifications in A, D is an intermediary parameter that introduces a conflict when maximising B.

Another example is provided by the relationship defined by the instance of D, D1. This instance redefines the behaviour of D, because uses the independent negative ($i-$) relationship, that means that regardless the modifications in D1, B always decreases when D1 is modified. So, this is a conflict of type c1 which affects maximising B. As this type of conflict is visible when the individual recommendations are calculated, these types of conflicts are considered as facts of type *avoid*. Moreover, a recommendation is included to suggest the modification that last affects the objective.

Furthermore, as A1 inherits the weight from A (w1), and it is the same weight as w4 and w5, the final recommendation considering D does not consider A1, and, althought A2 is considered (w2>w1), it is targeted as a conflict (avoid) because of the relationship with D.

```
f-50    (conflicts (toprint "
    Conflicts:
    1: when the parameter A changes, the target maximize B is unreached.
    2: when the parameter D changes, the target maximize B is unreached.
    3: when the parameter A1 changes, the target maximize B is unreached.
    4: when the parameter A2 changes, the target maximize B is unreached.
    5: when the parameter D1 changes, the target maximize B is unreached.
    ") (idn 6) (toMATLAB "{{1,avoid,1,-1.0};
    {1,avoid,4,-1.0};
    {1,avoid,5,-1.0};
    {1,avoid,6,-1.0};
    {1,avoid,10,-1.0};
    }"))
```

Figure 5.7: Example: Internal fact in CLIPS generated for conflicts. All the facts are processed by SQT-RS before being shown to the user in an suitable format.

Both examples for c1 and c2 are very simple. Indeed, these relationships are identified in very long parametric trees, and therefore considerably increase the calculation of the dynamic facts.

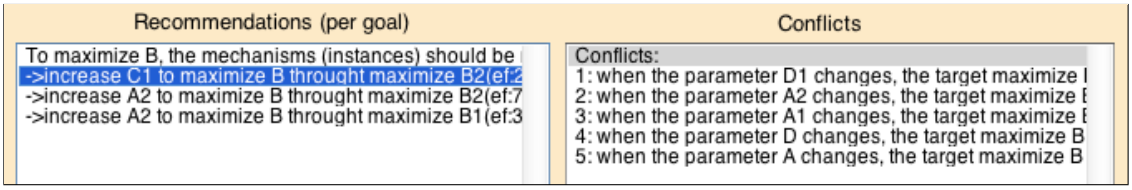


Figure 5.8: Example: Recommendations and conflicts in GUI.

Finally, the user should be able to interpret the results provided in Figure 5.8. In this case, SQT-RS recommends increasing C1 and A2 to maximize the instances of B (and therefore the property B). But also a list of conflicts is displayed. The conflicts in this case indicate that the user must decide what parameters are more relevant in the system. As is appreciated in Figure 5.8, A is a conflictive parameter because when increases, B increases too, and also D, which in turn decreases B. As can be seen, is through C1 that B could be increased without conflicts, since there are not conflicts that affect the parameter C. The results also indicate that another possibility is to analyse the effect that increasing A has on B compared to the effect that decreasing D (given A) has on B. However, this is part of a more detailed analysis, given as a consequence of new tradeoffs to be considered.

5.5. Number of Facts Based on the Knowledge

In this section we provide some results of SQT-RS based on the goal maximise Energy (*max Energy*), using the PC analysed in the first use case (Chapter 6.2), which instantiates the parameter Authentication, in a Wireless Sensor Network (WSN) scenario.

Figure 5.9(a) shows the file of facts generated in this example by SQT-RS. We focus on the goal *max* Energy because the file of facts generated is smaller compared with the goal *min* Energy. This is because of the types of relationships defined in the PRM source. We also select Energy because Authentication, which is an instantiated parameter, affects it. So, the file of facts also shows the facts generated for Authentication and the instances of this parameter. Moreover, the facts are shown only for testing purposes. The aim, is that this file will only be used by SQT-RS, the final user can only see the final recommendations. The recommendations set given the facts in Figure 5.9(a), are shown in Figure 5.9(b).

In Figure 5.9(a) only Authentication is instantiated. Specifically, CAS, DAD, IDS and IMBAS are instances of Authentication. The rest of the parameters remain within the default value of effect 1, while the effect for increasing/decreasing the instances is conditioned to the values given in the PC. In addition, the final recommendations set, does not consider instantiated parameters. Instead, it takes the results given by the instances of the parameters. The results shown in Figure 5.9(b) first shows the individual recommendations which affect the goal to a greater extent, given the context.

For example, given the first individual recommendation, it is possible to estimate that, to maximise the parameter Energy, if we are providing Authentication mechanisms, we should choose IDS instead of DAS. Because if DAS is avoided, the effect (ef) on Energy is reduced by 106. This effect is the result of increasing and decreasing the parameters in the individual recommendations.

```
(goal (priority 1) (criterion maximize) (parameter 13))
(op (todo decrease) (on 4) (to maximize) (p 13) (val 66.000000))
(op (todo decrease) (on 10) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 12) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 23) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 24) (to maximize) (p 13) (val 1.000000))
(op (todo increase) (on 25) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 26) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 27) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 28) (to maximize) (p 13) (val 1.000000))
(op (todo increase) (on 29) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 30) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 32) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 36) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 44) (to maximize) (p 13) (val 1.000000))
(op (todo decrease) (on 49) (to maximize) (p 13) (val 102.000000))
(op (todo decrease) (on 50) (to maximize) (p 13) (val 106.000000))
(op (todo decrease) (on 51) (to maximize) (p 13) (val 38.000000))
(op (todo decrease) (on 52) (to maximize) (p 13) (val 66.000000))
(parameter-instantiated (id 4) (name "Authentication") (layerP "High-Layer") (typeP "Security"))
(parameter (id 10) (name "PowerConsumption") (layerP "LocalProperties") (typeP "Performance"))
(parameter (id 12) (name "Rayleigh") (layerP "LocalProperties") (typeP "Performance"))
(parameter (id 13) (name "Energy") (layerP "LocalProperties") (typeP "Performance"))
(parameter (id 23) (name "DataRate") (layerP "Communication") (typeP "Performance"))
(parameter (id 24) (name "PacketSize") (layerP "Communication") (typeP "Performance"))
(parameter (id 25) (name "SignalStrength") (layerP "Communication") (typeP "Performance"))
(parameter (id 26) (name "DataTransmission") (layerP "Communication") (typeP "Performance"))
(parameter (id 27) (name "TransmissionTime") (layerP "Communication") (typeP "Performance"))
(parameter (id 28) (name "TransmissionPower") (layerP "Communication") (typeP "Performance"))
(parameter (id 29) (name "TimeSleeping") (layerP "Communication") (typeP "Characteristics"))
(parameter (id 30) (name "RequiredTimeOn") (layerP "Communication") (typeP "Characteristics"))
(parameter (id 32) (name "Retransmission") (layerP "Communication") (typeP "Consequences"))
(parameter (id 36) (name "PacketLoss") (layerP "Measurements") (typeP "Performance"))
(parameter (id 44) (name "Congestion") (layerP "Environment") (typeP "Consequences"))
(parameter-instance (id 49) (name "CAS") (layerP "High-Layer") (typeP "Security") (id-parent 4))
(parameter-instance (id 50) (name "DAS") (layerP "High-Layer") (typeP "Security") (id-parent 4))
(parameter-instance (id 51) (name "IDS") (layerP "High-Layer") (typeP "Security") (id-parent 4))
(parameter-instance (id 52) (name "IMBAS") (layerP "High-Layer") (typeP "Security") (id-parent 4))
```

(a) Facts for maximising Energy.

```
To maximize Energy:
-> decrease DAS(ef:106.0);
-> decrease CAS(ef:102.0);
-> decrease IMBAS(ef:66.0);
-> decrease IDS(ef:38.0);
-> decrease Congestion(ef:1.0);
-> decrease PacketLoss(ef:1.0);
-> decrease Retransmission(ef:1.0);
-> decrease RequiredTimeOn(ef:1.0);
-> increase TimeSleeping(ef:1.0);
-> decrease TransmissionPower(ef:1.0);
-> decrease TransmissionTime(ef:1.0);
-> decrease DataTransmission(ef:1.0);
-> increase SignalStrength(ef:1.0);
-> decrease PacketSize(ef:1.0);
-> decrease DataRate(ef:1.0);
-> decrease Rayleigh(ef:1.0);
-> decrease PowerConsumption(ef:1.0);
```

(b) Recommendation set.

Figure 5.9: Example: Facts and recommendation.

Note that Energy is a non-instance, non-instantiated parameter. The rules for building the recommendation are different from the contextual cases. Therefore, the selection of Energy as a parameter in the goal is only complicated by the number of relationships which affect Energy.

Intuitively, the presentation of the results can be enhanced depending on the user. We have chosen the representation in Figure 5.9(b) because we think that it is very intuitive given the formulation of the problem in this chapter. However, this text was generated by the rules in CLIPS, and can be modified, if required.

Furthermore, it is just as possible to provide additional feedback to SQT-RS from the .clp file, taking advantage of the powerful interpretation of Matlab of strings as commands, functions, etc. It is not complicated to provide automatic feedback to SQT-RS, simply by using the structures according to Section 5.4. Said structures can be interpreted from SQT-RS and set up in the CPRM chosen.

Finally, the file in Figure 5.9(a) has a reduced set of parameters. This is because SQT-RS only converts to facts, those parameters that are part of a set of interest. The set of interest is formed by any parameter in a goal, and, if the parameter in the goal is of type instantiated, then, their instances are added as subgoals. After that, the individual recommendations for the parameters in the set of interest are retrived. This criterion of selection of information, based on the set of interest, is to make it more efficient. Indeed, it is possible to enhance the rules for inferring additional information based on the whole set of parameters, if the conversion to facts is not restricted just to the parameters in the set of interest. However, the size of the files of facts can grow too much in these cases. In the current version of SQT-RS, the size of the file of facts depends on the number of parameters in goals, the number of parameters in goals that are instantiated, and the accumulative influence degree of the parameters in the set of interest. If all the parameters have been selected, the number of

facts in the file of facts, based on the number of parameters, and considering the additional facts produced by instantiated parameters, is defined as follows:

$$\#Facts = \sum_{j=1}^{M}(i(x_j)+1)(\iota^i(x_j)+2) \tag{5.28}$$

Where:

M: is the number total of parameters.

$i(x)$: is a function which returns the number of instances of a parameter x: $|H_x|$, $H_x = \{y|x \in P(y)\}$.

$\iota(x)$: is the accumulative influence of the parameter x, as defined in Table 5.1.

$\iota^i(x)$: considers the accumulative influence when the parameter x is instantiated.

$$\iota^i(x) = \begin{cases} \sum_{y \in H_x} \iota(y) & \text{if } type(x) \text{ is "instantiated"} \\ \iota(x) & \text{in other case} \end{cases} \tag{5.29}$$

Considering E, N and L, respectively, the total number of non-contextual parameters, instances and instantiated parameters, such that $M = E + N + L$. The number of facts based on the contextual number of parameters is shown in Figure 5.10. The average of instances per parameter instantiated can be calculated as:

$$\bar{i} = {}^1/_L(\sum_{i=1}^{L}|\{y|x_i \in P(y)\}|) \tag{5.30}$$

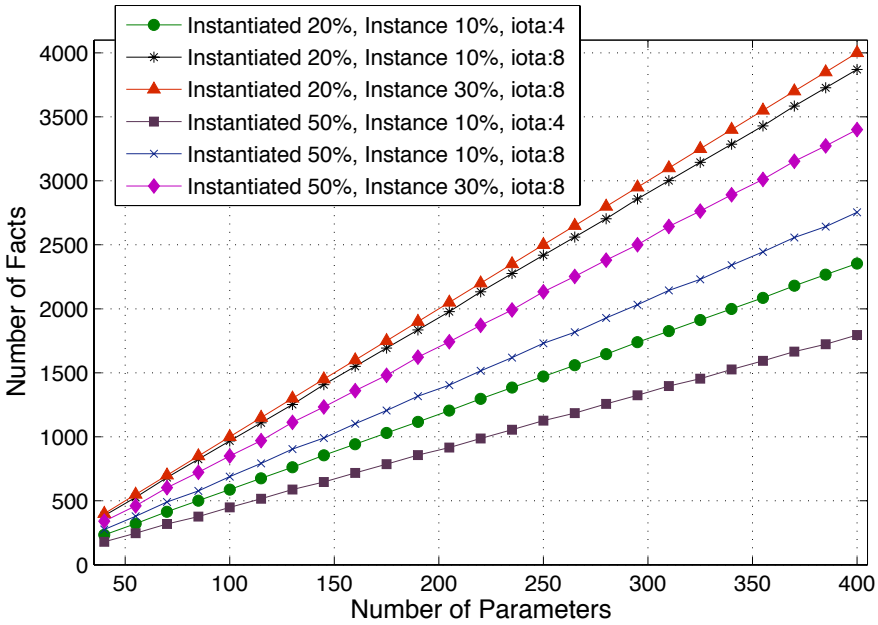


Figure 5.10: Increasing facts based on the context.

Figure 5.10 shows the effect that increasing the number of instantiated parameters, instances and accumulative influence (iota, ι) has on the set of facts. As can be seen, the worst cases are registered when the accumulative influence increases. Concretely, the worst case is when ι and the percentage of instances increases, followed by the increase of instantiated

parameters. So, the number of instances per instantiated parameter and the accumulative influence are key factors which affect the size of our files of facts.

Moreover, the values chosen for ι in Figure 5.10 are fixed, while it is obvious that ι varies according to each parameter in the model. Specifically, the parameters which provide a property, such as *Authorisation*, should have lower ι than the parameters which define a resource, such as *Memory* or *Energy*. While higher values of ι increase the complexity and the information to be stored and processed, a low value reduces the number of recommendations provided by SQT-RS for those parameter.

Finally, in Figure 5.10 those facts created to identify the conflicts are not considered because these depend on the dependencies in the system. Intuitivelly, if the facts generated to identify the conflicts are considered, the number of facts increases depending on the dependence degree. Furthermore, the higher the parametric trees generated, the greater the probability of having *opposite* conflicts.

5.6. Summary and Final Remarks

The SQT tool uses CPRM-based systems for measuring the Security and QoS tradeoff. However, SQT depends on the number of parameters. When the number of parameters increases, the final results are very difficult to analyse for a human. In this chapter we have overcome this limitation by defining a recommendation system, called SQT-RS, to advise the final user on the alternatives for characteristics, properties and mechanisms to be deployed. SQT-RS can be used with different types of parameters, not only Security and QoS. However, the basic set of parameters provided by the tool defines these types of relationships and no other. These can then be added dynamically by modifiying the files generated by SQT-RS. Furthermore, SQT-RS can be particularly useful for assessing Security and QoS tradeoffs in dynamic networks, where there is a great uncertainly about the final composition of mechanisms, services, applications and multipurpose entities.

SQT-RS is based on the definition of CPRM, so, in order to work, a set of parameters and their relationships are required so as to define the knowledge-based parametric system. There are other approaches for analysing complex decisions such as the *Analytic Hierarchy Process* (AHP) [123], or the *Potentially All PaiRwIse ranKings of possible Alternatives* (PAPRIKA) [124] approaches, however, these are fixed and do not define contextual parameters as CPRM does. Unlike these approaches, SQT-RS integrates the operations to enable a basic set of parameters to be enhanced, dynamically, based on the problem and the knowledge available at a given time, in the environment.

Additional approaches for providing recommendations considering Security and QoS are those that focus on service composition [119, 120], or on providing recommendations but to be implemented in devices [13]. The majority of these solutions are theoretical and do not provide solutions that consider the subjective opinion of the users on the parameters, nor do they consider how the information or the feedback is provided for the user. Solutions such as those provided in [121] are very interesting because of the relationship between the background system and how the information is traslated to the user's context. In this case, a friendly tool to simplify the decisions of the user in the selection of security goals is proposed. However, these approaches are still specific to a set of parameters, and the

dynamic adaptation of these or the integration of this information in other models is not trivial.

As the information in the model may change dynamically quite frequently, working with a dynamic model to provide dynamic recommendations based on heterogeneous contexts formed by different *things* or parameters is crucial to the convergence of the different technologies in future networks.

CHAPTER 6

Use Case Scenarios

In this chapter, the tool developed for assessing security and QoS tradeoff is evaluated with two use cases directly related to the three networks selected to be part of the Future Internet and Internet of Things. The first use case defines the integration of a particular context that provides authentication mechanisms for a Wireless Sensor Network scenario. This use case focuses on the validation of the initial prototype developed in Chapter 4. The aim of the second use case is to evaluate the recommendation system developed following the indications given in Chapter 5. In this second use case, the scope is 5G Green relay networks, to cover cellular networks and ad-hoc networks. In this case, we develop an additional component to easily generate the final schemes based on a set of requirements which define the final environment. Moreover, the particular contexts for eavesdropping and jamming may be integrated in the final scheme using said functionality.

6.1. Overview

In Chapter 1 a set of candidate networks to be part of the FI were chosen based on a set of requirements: (i) ad-hoc environments *where the user is present*, (ii) resource-constrained environments *with elements/sensors capable for collecting information*, (iii) environments whith a *communication infrastructure with more resources*, and (iv) the chosen environments have proved *their interoperability with each other*, or are taking several steps towards the networks convergence.

Finally, MANET (i,iv), WSNs (ii,i,iv), and cellular networks (iii, iv) were chosen, and analysed in Chapter 2 to extract information valuable for the analysis of the security and QoS tradeoff. The models defined in Chapter 3, PRM and CPRM allow to handle different types of parameters, and these models are implemented in the SQT tool provided in Chapter 4. Moreover, SQT was enhanced to provide recommendations, using the module SQT-RS defined in Chapter 5.

The use cases provided in this chapter are defined to cover all the contributions in this thesis. The first use case uses a basic set of parameters as discussed in Chapter 2, and evaluates the PRM defined in Chapter 3 and SQT (Chapter 4) using parameters for a WSN. The second use case, evaluates the enhances introduced by SQT-RS in Chapter 5 and uses parameters defined for Green 5G relay networks. 5G relay networks can be considered MANET networks and also cellular networks. So, we consider that these use cases are relevant to the problem proposed, since considers all the candidate networks chosen in Chapter 1, and, therefore, allow the validation of our approach (Chapters 2 to 5) to the analysis of the security and QoS tradeoff in FI environments.

6.2. Use Case 1: Authentication in WSN

Here we define the first use case to be implemented. It uses a scenario where a Wireless Sensor Network (WSN) may be deployed, and two authentication mechanisms are available, *Certificate-based Authentication Scheme* (CAS) and *Direct Storage based Authentication Scheme* (DAS) [125]. To consider sensors as possible devices involved in the network, it is necessary to take into account parameters such as energy or power consumption, amongst others. The following sections separate the parameters in the base context, considered to implement the behaviour of a WSN, from the parameters in the specific context, defined to implement the use case related with the authentication in WSN.

6.2.1. Parameters in a Base Context

The parameters in the base context, shown in Table 6.1, have been taken from Chapters 1 and 2, as part of the detailed study where the convergence of WSN with other networks in the *Internet of Things* (IoT) has been analysed. Here, the basic parameter set is reduced so as to be able to work with it from a theorical point of view.

Although the default GC, given these parameters, is set to 1 for all the parameters ($\forall p | p \in PRM, w_p = 1$), it is possible to set a subjective GC, based on our own priorities. For example, increasing the relevance/impact of the parameter *Encryption*, will cause

Table 6.1: Parameters for a Base Context

	HIGH-LEVEL REQUIREMENTS
QoS	Reliability, Fault Tolerance, Availability
Security	Authentication, Authorisation, Confidentiality, Integrity, Trust, Privacy
	LOCAL PROPERTIES
Resources	Power Consumption, Memory, Rayleigh Channel, Energy, ComputationTime
Security	Anti-Tampering, Encryption, Public Key Cryptography, Symmetric Cryptography, Secure Key Exchange, Secure Key redistribution, Key Generation, Signature Scheme, Certificate
	COMMUNICATION
QoS	Data Rate, Packet Size, Signal Strength, Data Transmission, Transmission Time, Transmission Power
Characteristics	Time-sleeping, Required-time-on, Routing Protocol
Consequence	Retransmmission
	MEASUREMENTS
QoS	Throughput, Delay, Jitter, Packet Loss, Response Time, Bit Error Rate (BER)
	ENVIRONMENT
QoS	Allowable Bandwidth, Error Probability
Attacks	DoS, Malicious Devices
Consequence	Interference, Congestion, Overhead, Fading, Shadowing, Noise

any parameter related to Encryption in the antecedent, that is, $\forall y | Encryption \rightarrow y$, to be affected more than the rest. The parameters affected by the increase/decrease of the parameter Encryption, can be consulted using the parametric tree. However, the final impact on the parameters may vary according to the type of the relationship defined between the parameters (see Chapter 3.1), and the weights assigned to them.

From now on, all the weights for the parameters (the relevance), are set to 1. Moreover, the relationships are defined in the default GC with value 1 for all the relationships ($w_d = 1, \forall d : A \rightarrow B | d \in PRM$). These values can be modified in the GC, but, in our case, the changes are made using an example of instantiation of parameters. In other words, we use PCs to assign final weights w_d to the dependencies in the model [1].

6.2.2. Parameters in a Particular Context

Once we have the CPRM, it is possible to integrate different PCs inside it. Then, it is said that the CPRM has been instantiated, and the new model is denoted as $CPRM_i$. For example, in the following sections, the PC shown in Table 6.2, is integrated in the CPRM which contains the parameters shown in Table 6.1. The weights in Table 6.2 have been fixed in accordande with the work done in [1], while the information about the specific relationships defined in Table 6.2 is extracted directly from the same, aforementioned paper. The rest of the relationships, defined in the PRM, are taken from Chapter 3.

[1] Given Table 3.8, changing the value of the relationships where a parameter *parent* is involved is of very little use when, in the next step, the instances of the parameter will define the same relationships using their own weights.

Table 6.2: Weights w_d according to [1]

General Parameter	Dependence			Weight
	Antecedent	R	Consequent	w_d
Authentication	CAS	+	ECDSA	1
	DAS	+	ECDSA	1
	CAS	$\neg c$	Memory	0
	DAS	$\neg c$	Memory	5
	CAS	c	PacketSize	5
	DAS	c	PacketSize	1
	CAS	c	Certificate	2
SignatureScheme	ECDSA	$\neg c$	Energy	1
	PairingBased	$\neg c$	Energy	5
	ECDSA	c	ComputationTime	1
	PairingBased	c	ComputationTime	5

So, the parameters to be instantiated are in this case *Authentication* and *SignatureScheme*, and the instance parameters are CAS, DAS, ECDSA and PairingBased. Note that both *parents*, Authentication and SignatureScheme, are defined in the BC in Table 6.1. Therefore, they have their own relationships in the PRM. The definitions for CAS and DAS can be found in [126]. CAS uses the user's public/private key pair to provide authentication. This requires the use of certificates, therefore CAS adds the relationship $CAS \rightarrow Certificate$. To the contrary, DAS, avoids the use of certificates to reduce the overhead. To do this, DAS stores the current user's ID information and their public keys. Both schemes use the *Elliptic Curve Digital Signature Algorithm* (ECDSA) to sign the broadcast messages.

So, the focus here is to integrate the specific information or context behaviour, represented in Table 6.2, in our predefined model. Once this new information has been integrated, the final instantiated model, will provide specific information about the authentication and signature schemes allowable in the final environment.

The tradeoff is in the effect that these mechanisms can have on the rest of the parameters already defined in the model, that were not previously considered in the specific papers or any other source from which the PC was extracted. Although the scenario proposed here has been simplified, the final scenario can be very complex, composed by several PCs. For example, the combination between the defined instances and new instances defined for another parameter, for example the authorisation parameter [127], is possible with the current definition of the model. Of course, this implies new tradeoffs to be considered, and complicates the basic analysis of tradeoffs. In what follows, the objective is to show how SQT can perform the analysis using the simplified case, in order to provide the basis for its use.

Note that, once the new PC has been integrated, the parameters Authentication and SignatureScheme will be new layers, so they can be analysed from this perspective. It is also possible to operate with these parameters using the type *instantiated*, and all the instances can be selected together, by using the type *instance*.

These options are enabled so as to test the impact that these changes have on the model. In the final model, when the parameters Authentication and SignatureScheme increase, the instance parameters defined for these parameters also increase, and, consequently, these

actions trigger the dependencies defined for these parameters. However, it is important to remember that, with each new integration in the model, the ARs defined in Table 3.8, can define new dependencies. So, when the parameters increase, the $CPRM_i$ can demonstrate a new behaviour, completely different from the original CPRM, which is normal, because the $CPRM_i$ is based on the context.

6.2.3. Setting up the model

In this first step we only load the model from a file, and show the influence and dependency degrees (Figure 6.1). Appendix A.1.1 shows the file *.m* in which the PRM, which contains the parameters shown in Table 6.1, was defined. It is important to note that the parameters in the PRM are parents, that is, non-instantiated parameters, but, even from this kind of scheme, we can extract some valuable general information shown in the next subsection.

Note that SQT allows the generation of a PRM by default, that can be saved (in a new file *.m*) and modified as we wish. The same can be done for any other structure or component used by SQT, even the workspace can be saved and loaded again.

Once the model has been loaded, the working panel can be used to show the accumulative influence and dependence degree. This is a general view of the impact that the parameters have on the model (Figure 6.1). For example, as can be observed, in the PRM chosen, the parameter Trust does not affect the rest, while in other scenarios, such as mobile platforms, this parameter is directly related to the user's experience. Indeed, the parameter Trust usually depends on cryptographic mechanisms, and reputation protocols. However, it is also possible to consider Trust as a subjective value imposed without taking into account cryptographic primitives. Here the model has been simplified, considering only some of the relationships that can be found in the general model provided in Chapter 2. To the contrary, as can be observed, the influence of the parameter Authentication has been especially considered, and these relationships are provided with special interest.

6.2.4. Analysis of parameters

The model allows the analysis of parameters before and after the instantiation. The differences between both are shown throughout this analysis.

6.2.4.1. Prior to the Instantiation

Here we show the parameter tree for those parameters that have not yet been instantiated, but will be instantiated in the next step. The following analysis focuses, in particular, on the parameter Authentication, whose parametric tree is shown in Figure 6.2, given according to the definition of the PRM in Section 6.2.1. Specifically, the figure illustrates the parameters that are affected when Authentication is provided (*increased*, that is Δ). This dependency tree is quite different from the parametric tree calculated for decreasing the Authentication (∇), shown in Figure 6.3.

Parametric trees should not be confused with the general parametric tree, where each parameter is noted with its type, showing a general view of all the relationships in the environment. Instead, the parametric tree for a parameter is a diagram, particularised to

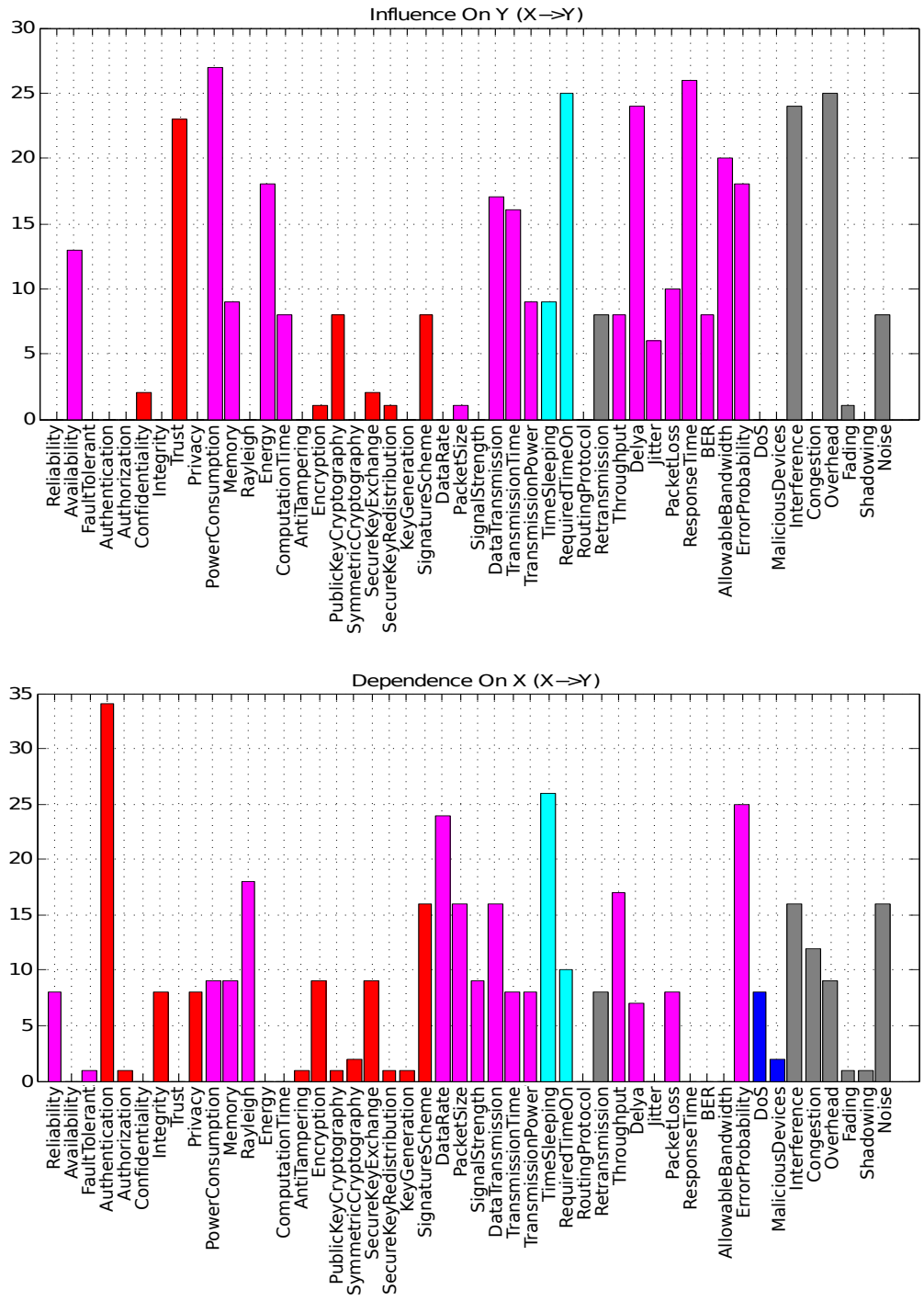


Figure 6.1: Influence and Dependence degree.

one parameter, as Figures 6.2 and 6.3 illustrate for Authentication. Therefore, it does not show all the possible relationships in the model, or layers, as the first one does.

Figure 6.2 can be interpreted as follows: when Authentication is provided, based on the current literature, the parameters ResponseTime, PacketSize, Memory and SignatureScheme

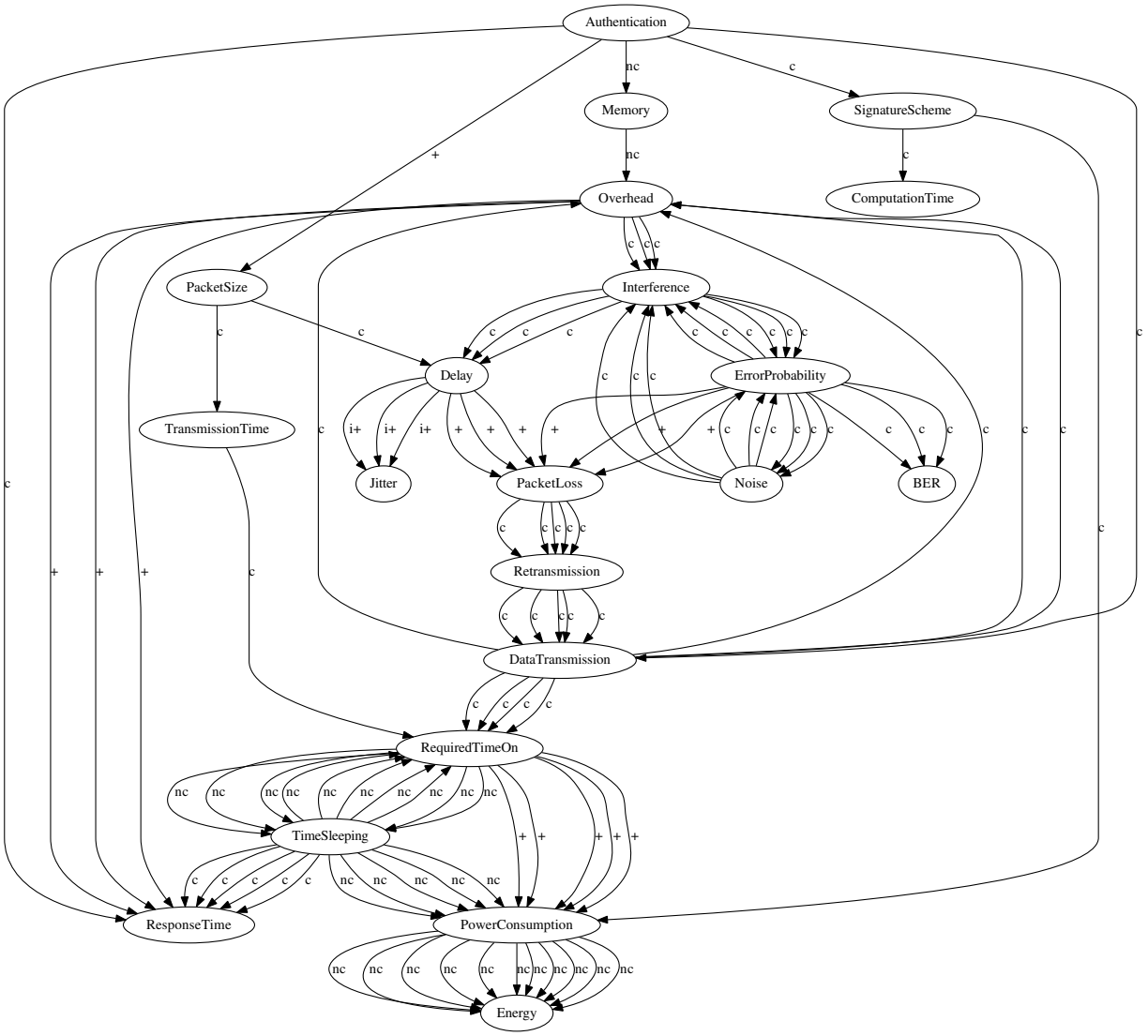


Figure 6.2: Parametric Tree: Increasing Authentication.

can all be affected by an increase, or, in other words, if they are properties, they should be provided. Moreover, each one of these parameters has its own dependencies that were originally defined in the PRM. So, for example, it is known that an increase in the memory can create an overhead in the system, if there are more services which need this memory in order to work properly. In this case, the interference is considered as the probability that the system will collapse, decreasing the performance. Intuitively, under these conditions, the probability of failures in the system increases, as does the delay. Both, delay and error probability may cause packet loss, even when they are considered at the local layer, because these finally affect the capacity of the system to respond to the neighbouring demand.

Once packet loss has been affected, the probability of retransmissions increases, and, in a WSN, this means that the nodes cannot enter into a sleep state to save energy as much as they should. Therefore, in a WSN, if the time for the antennas to be in power on increases, it implies less time in inactive mode, so, the power consumption increases, which results in

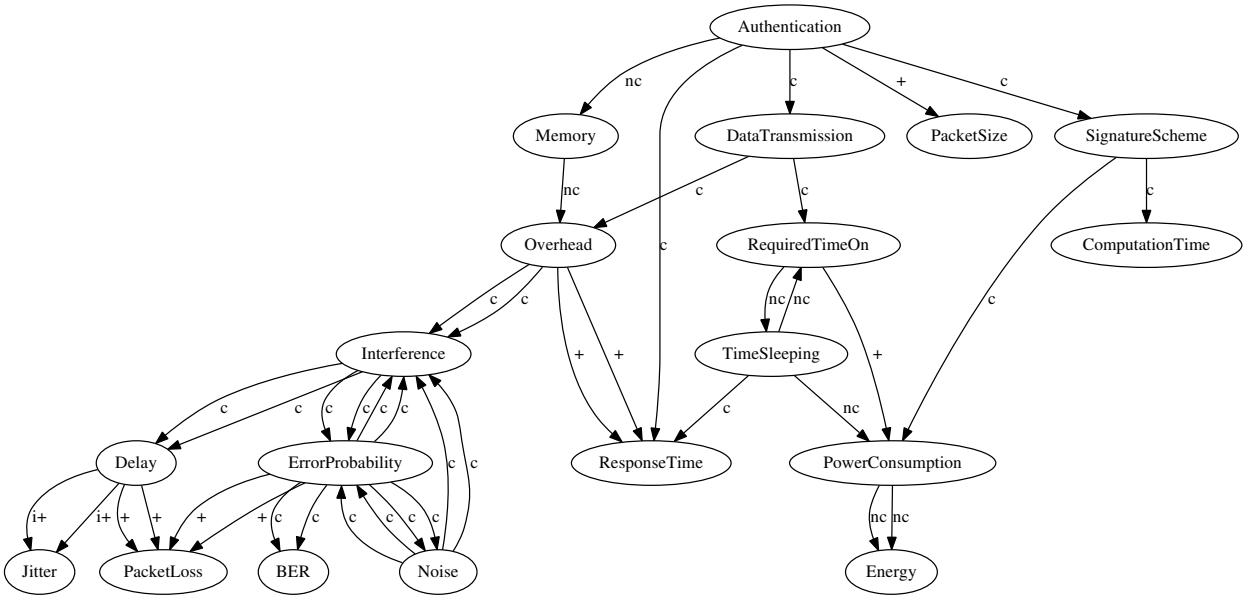


Figure 6.3: Parametric Tree: Decreasing Authentication.

a decrease in global energy.

When, in a parametric tree, there are leaf nodes, it can mean two things: first, there are no relationships defined for this parameter. Second, when this parameter is affected by the corresponding effect, increasing (Δ), or decreasing (∇), this effect has no result; there is nothing that can be done.

For example, note the case of $Authentication \xrightarrow{+} PacketSize$. As a positive relationship ($+$) was defined, then, PacketSize is only affected if Authentication increases. Otherwise, the system has no information, and in that case, the influence chain stops at PacketSize. This difference can be observed in Figure 6.2, where an increase in Authentication triggers an increase in PacketSize, while in Figure 6.3, a decrease in Authentication has no effect on PacketSize, so, in the latter case, there is no propagation through this branch.

So, as can be seen, the *complete* relationships (c and $\neg c$) are defined in both cases, for increasing and decreasing, and therefore can be observed in both trees.

In the following sections, in order to broadly outline the use case, the parameter tree for decreasing is considered. This is because the dependency trees, although simplified for testing, are very extensive, and the trees for decreasing are smaller in this case.

6.2.4.2. After the Instantiation

Here we show the new tree for the parameters instantiated, that is, the parents. Using the PC defined in Section 6.2.2, and the given relationships, Figures 6.4 and 6.5 show the parametric trees for the parameter Authentication, once it has been instantiated. Moreover, as Authentication is related to SignatureScheme, then both instantiated parameters appear in said figures.

Note that, the relationships are duplicated because CAS and DAS are shown in the diagram, and both inherit the relationships from their parent, Authentication. Moreover,

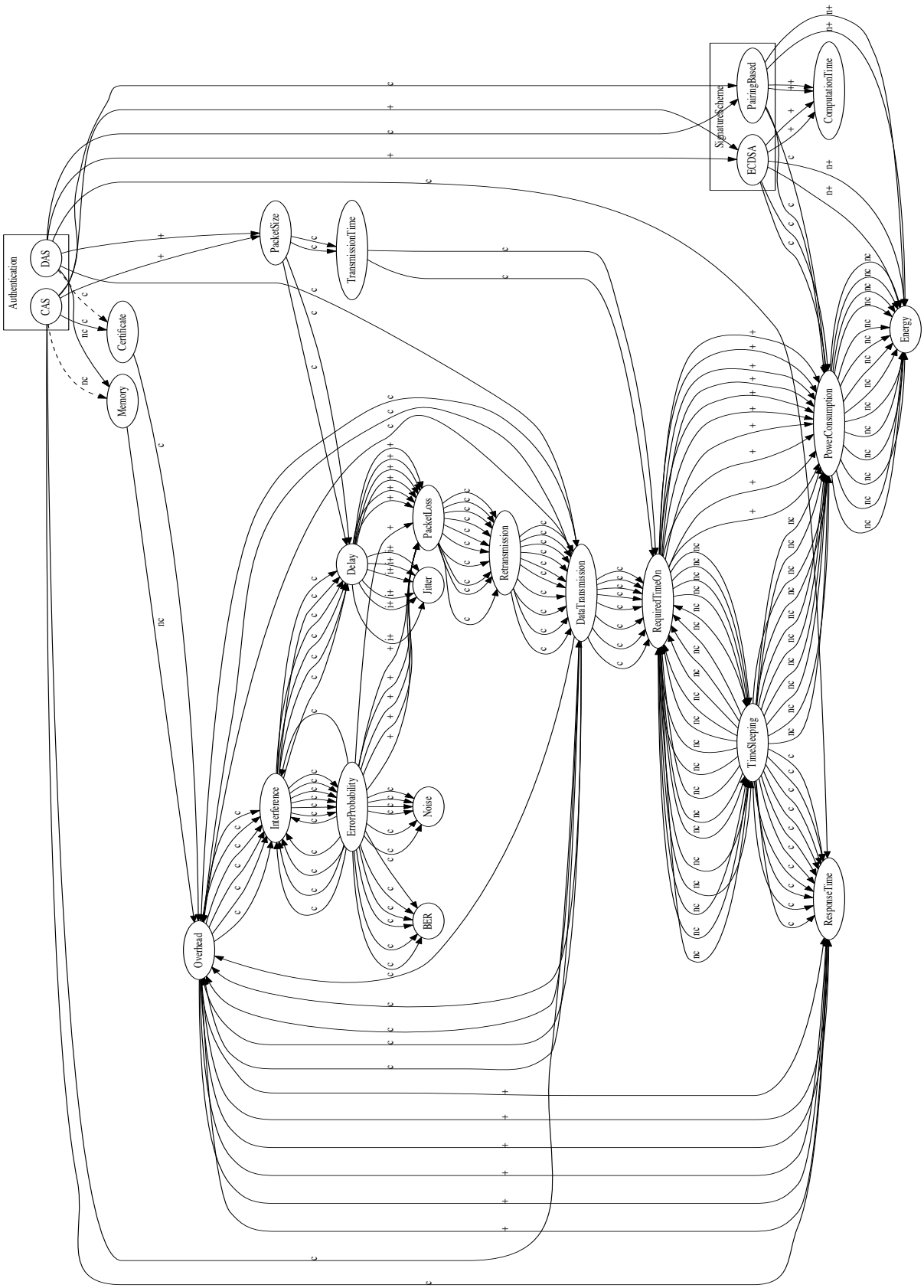


Figure 6.4: Parametric Tree: Increasing Authentication (instantiated).

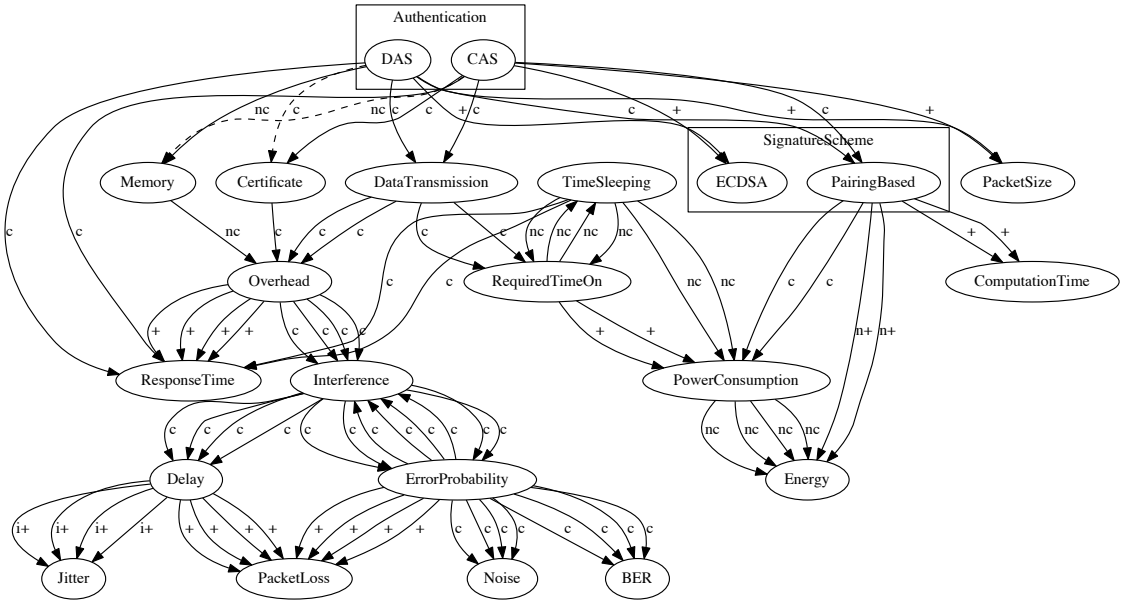


Figure 6.5: Parametric Tree: Decreasing Authentication (instantiated).

Authentication integrates the new behaviour defined by CAS, $CAS \rightarrow Certificate$. This new behaviour is integrated using weight 0, because according to rule AR3 any relationship for an instance must be supported by the instantiated parameter without affecting the pre-defined behaviour of the parent. Therefore, as the behaviour of Authentication is not modified by this new information, DAS inherits the new information but its behaviour is not affected by this new relationship because the weight is 0 (dashed line). Since Figure 6.5 (decreasing) is smaller that Figure 6.4 (increasing), we focus on the decreasing case showing the specific trees for CAS and DAS. If we select the parametric tree for both parameters CAS and DAS separately, these are very similar (Figures 6.6 and 6.7).

The differences between parametric trees for the same parameter (e.g. Authentication) are due to the meaning of the relationships. For example, although ECDSA has relationships defined, these are not applicable when Authentication is decreasing, because the relationship is positive (+). In that case, ECDSA is a leaf in the parametric trees for decreasing. A different case is when Memory is a leaf in Figure 6.6 (decreasing CAS) but not in Figure 6.7 (decreasing DAS). This is because the weight in the relationship between CAS and Memory is set up to 0 in Table 6.2, so, in this case, there is nothing to do. Therefore, our last implementation of SQT avoids to show branches with weights equal to 0 or without effect on the rest of parameters.

However, these similarities, do not affect the tradeoff analysis, which is performed based on the weights, which are not visible in the parametric tree diagram. Note that the weights defined in the PC for the relationships of CAS and DAS are different, so instead of inheriting the weights for the relationships of Authentication, both parameters have their own specific

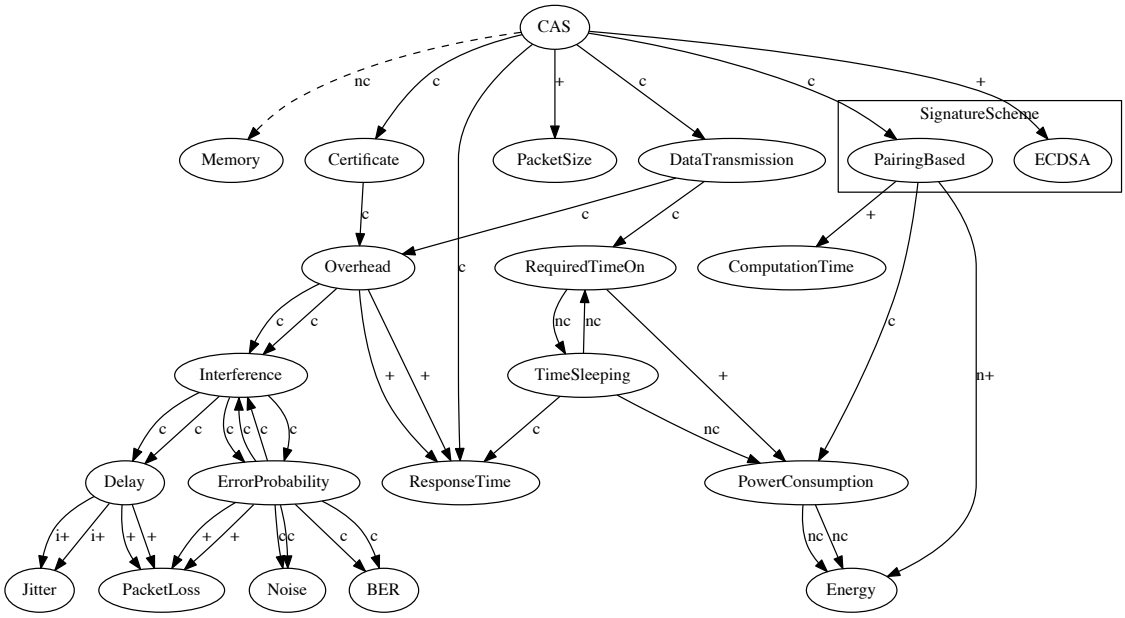


Figure 6.6: Parametric Tree: Decreasing CAS.

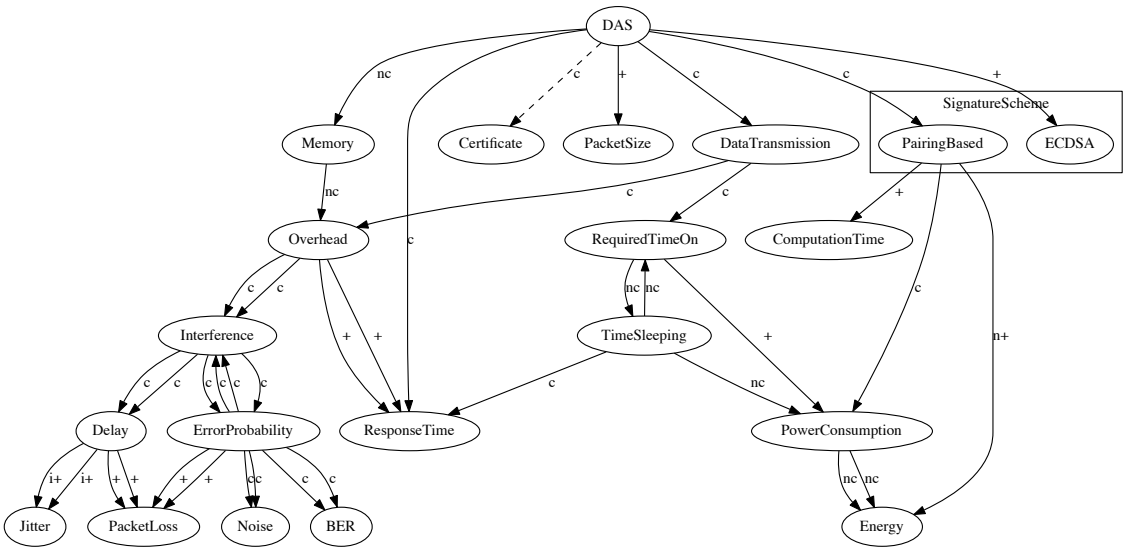


Figure 6.7: Parametric Tree: Decreasing DAS.

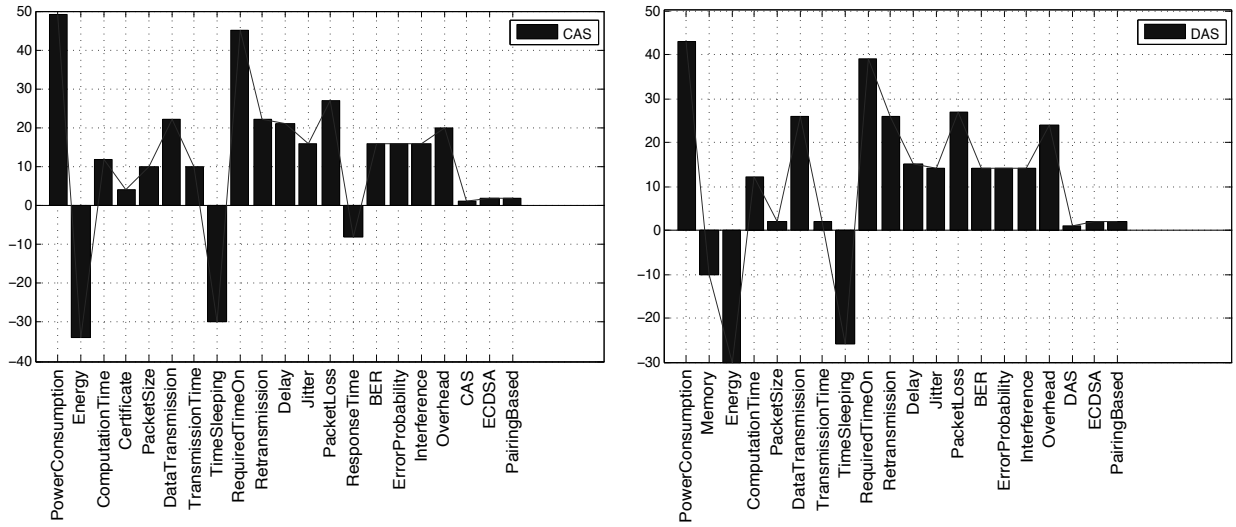


Figure 6.8: Impact of CAS and DAS on the Performance.

conditions and measurements. Therefore, the final impact of them on the final system is different, as Figure 6.8 shows.

Indeed, the results after increasing the parameters CAS and DAS differ, particularly, in the set of parameters: PowerConsumption, PacketSize, DataTransmission, TransmissionTime, Delay, ResponseTime and Overhead. A common graph for both mechanisms is shown in Figure 6.9. In this case, we consider it better to show the results for the increasing of both parameters, because it helps in the case that a system administrator wants to use SQT to assess the Security and QoS tradeoff, focusing on the two mechanisms used for implementing Authentication.

Remember that one of the objectives of DAS by avoiding the use of certificates is to reduce the overhead. However, in our scenario the value for Overhead is higher using DAS than using CAS. This is because we consider additional parameters in our analysis. Note that in our analysis, the parameter Memory affects the Overhead with weight 5. There are other parameters that also affect the Overhead, for example the PacsetSize. However, the final impact is higher using DAS than CAS. This can be different if the weight for the relationship $CAS \rightarrow Certificate$ is higher than the current value (2).

Moreover, note that the value of PowerConsumption in CAS is higher than in DAS, according to Figure 6.9. As can be noted in Figure 6.10, the increase in PacketSize affects PowerConsumption much more than the increase in Memory. Considering that the relationship $CAS \xrightarrow{c} PacketSize$ takes weight 5 (Table 6.2), the impact of this authentication mechanism on PowerConsumption is justified. Note that this conclusion is reached after considering the usual parameter set and the defined relationships.

Focusing on the main differences, in this use case, if we need to use CAS, PowerConsumption is a key value that has to be considered. Taking this into account, given the information in Table 6.2, which was set in the model, intuitivly CAS should be combined with ECDSA. However, it is possible to find new combinations in the intermediary parameters (in the parameter tree) in order to minimise the impact of CAS on PowerConsumption.

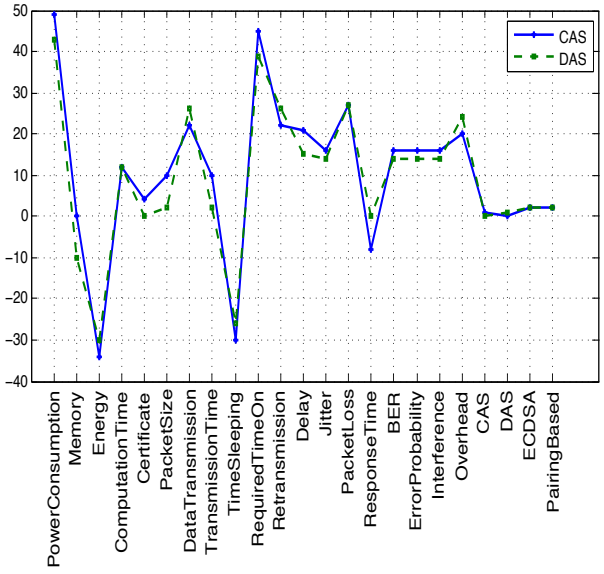


Figure 6.9: CAS versus DAS.

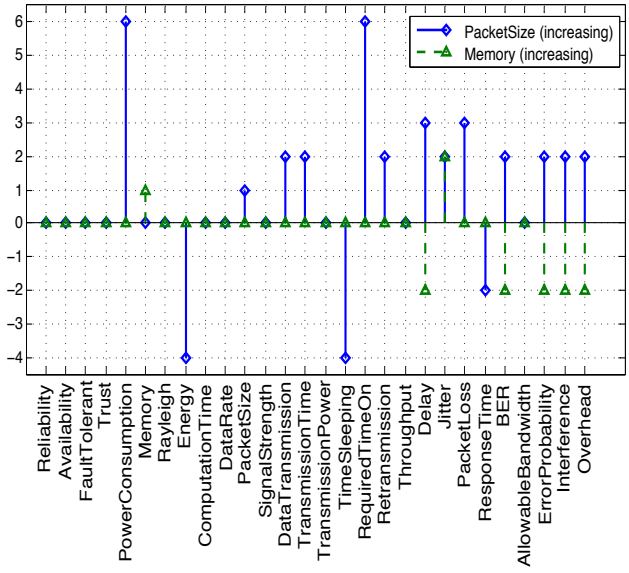


Figure 6.10: Packet size versus Memory.

This is only an example, given a known use case, but the idea is that this type of formulation of a system based on parameters and the dependencies between them, enables the evaluation of different mechanisms under a common language. Therefore, if these results are combined, for example, with authorisation mechanisms, we can extract new combinations of mechanisms to improve the configuration of the final system.

Finally, note that, while the relationships in Table 6.2 were extracted from [1], because they provided a good example to prove our approach, in our analysis we show the effect that these mechanisms can have on other parameters that have not been considered in Table 6.2. The idea behind SQT is precisely that we can combine different sources of information in order to evaluate the final set of data as a common behaviour, or context.

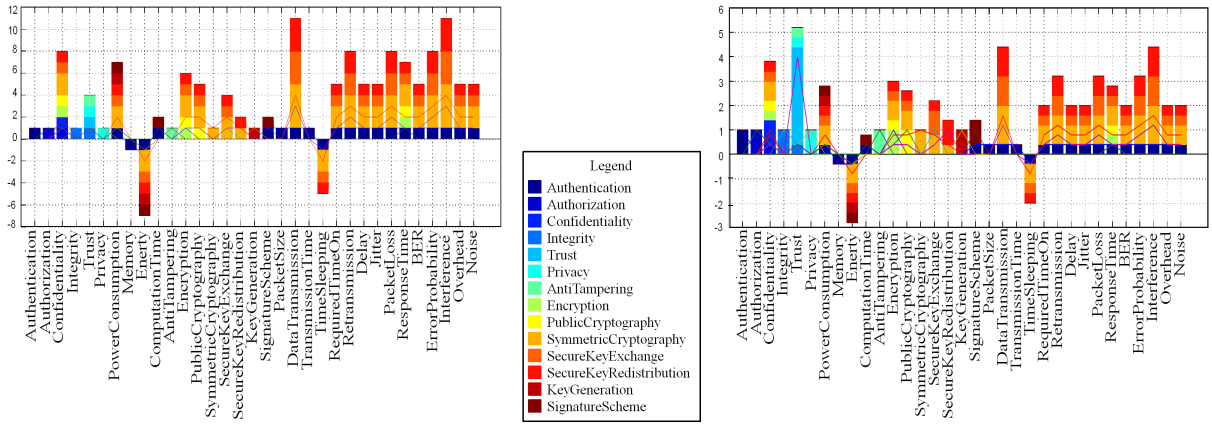
6.2.5. Setting up the Relevance (w_p)

In addition to the previous analysis, it is possible to change the relevance of a parameter.

For example, by considering that in the initial CPRM all the parameters have the same relevance ($w_p == 1, \forall p$). In order to indicate the relevance of a parameter in a CPRM, we target each parameter with a different weight w_p. This is necessary to set up the minimum context for the system: namely which of the parameters are most important to an administrator at a given moment.

For example, Figure 6.11(a) shows the effect on the system when security parameters are decreased, and all the parameters are weighted with 1. In Figure 6.11(c) the process is repeated with the weight for Trust modified to 4.

Increasing or decreasing the values for w_t and w_l have similar consequences. These weights are used in order to represent different contexts in the system (ej. increase the relevance of parameters of type *security*). Moreover, they can be considered as subjective values which can even represent specific needs or requirements in a network (e.g. increase the relevance of Authentication in the presence of guests in a network).



(a) $w_p == 1, \forall p \in PRM$

(c) $w_{Trust} == 4$

Figure 6.11: Increase Security.

However, in order to perform the Security and QoS tradeoff we need to compare different alternatives. So, probably, the most interesting change in the CPRM-based system is when the administrator sets up a PC. In the PC, specific mechanisms are considered as instances of general parameters. For example, the instantiation can be done by providing a property. In our case, the main objective is to design the CPRM-based system in order to perform the Security and QoS tradeoff. Thus, the next step is to show how security parameters in the GC can be instantiated in order to evaluate the final impact that security mechanisms have on the system.

6.3. Use Case 2: 5th Generation Green

In this section, the steps needed to deploy SQT-RS to assess the Security and QoS tradeoff in 5G Green environemnts are described. The solution is tested with the specific use case of relay selection in 5G scenarios, introduced in Chapter 1, Section 1.2.3.1.4. Different types of goals are selected, and then SQT return recommendations to achieve the selected goals, given the facts extracted from the CPRM-based 5G Green system.

The complete flowchart that illustrates the analysis of this particular use case - from the selection of the parameters to the final recommendations - is shown in Figure 6.12. The selection of parameters is performed using the *SQT 5G Green module* (SQT-5G) with two objectives: (i) help for training users, and (ii) allow the creation of CPRM schemes (.m files) to be used in SQT, based on a set of properties chosen by the user. These files are handled by SQT, and, therefore, the user can store and modify the files, extracting the PCs or integrating new ones. To do this, the graphical user interface (GUI) for SQT has been improved with the SQT-5G module. The parameters and their value in 5G Green are set up using the 5G selector module, which includes the use of fuzzy logic. The complete map of actions available using the SQT-RS tool are shown in Figure 6.13.

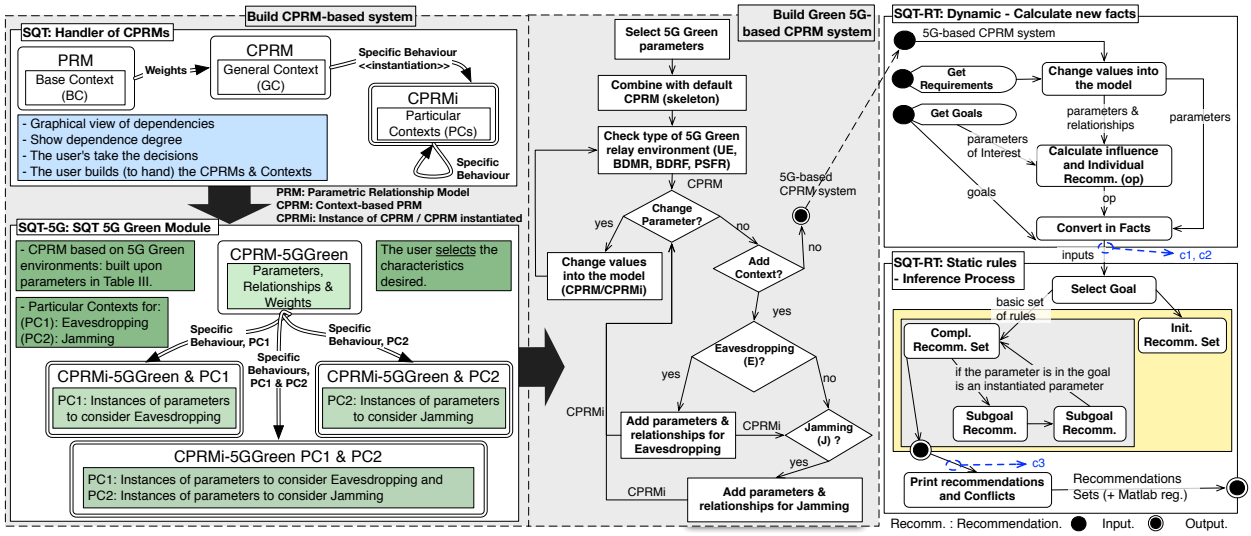


Figure 6.12: SQT-RG Green module.

The $CPRM_i$ to be evaluated is built, based on the value of parameters selected by the user, which determine the type of relay used at a given time, according to the parameters taken from [62]. Moreover, the instantiated model considers the behaviour when, in the environment, there are eavesdroppers or jammers. This information is taken from [60, 128, 129, 130].

Moreover, the recommendations for the different goals based on the contexts are discussed. Intuitively, the results change based on the final set of parameters and relationships, that were generated in the previous step. It is important to note that prior to the analysis of results, the set of parameters and their relationships has been thoroughly tested in order to model the behaviour of a 5G Green relay system. This consists of different tests for iden-

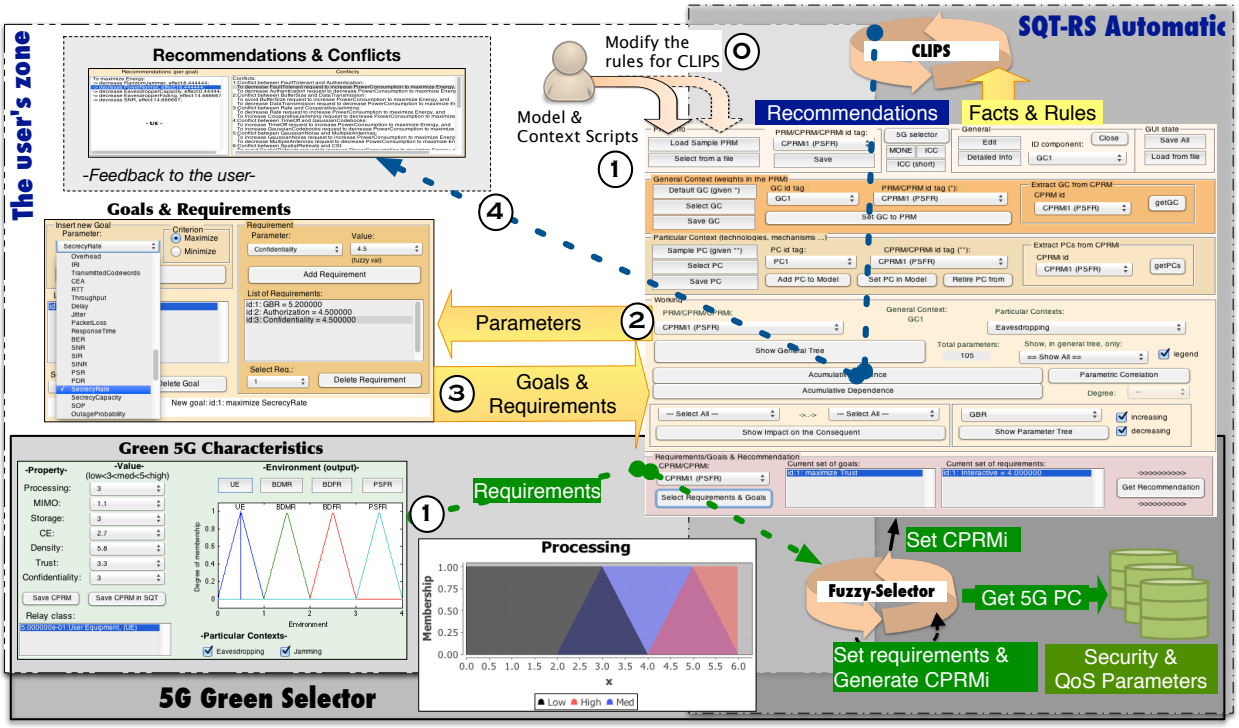


Figure 6.13: Deployment using 5G selector.

tifying general incoherences given by the propagation of the effects through the parametric tree. When the behaviour of the model with the parameters in the basic set is reasonable [2], the different $CPRM_i$s are built using the basic parameters set and adding the new specific parameters and relationships. Then, we are able to test the recommendations provided by SQT-RS.

Specifically, once the definition of the parametric relationship model taken as a base has been completed, SQT-RS operates using a known set of previously tested parameters and relationships. It therefore helps to provide better recommendations in the configuration using the specific parameters and the values included dynamically. The idea is that by using SQT-RS, the modelling of the behaviour of the complete system is simplified: SQT-RS provides the recommendations that an expert can derive from visual observation.

Some results given, based on the goals and the scenarios considered, are shown in Figure 6.19 and Figure 6.20. Moreover, Figure 6.14 shows a portion of the list of facts given in the UE case combined with Eavesdropping (E) and Jamming (J) contexts. The final user does not work with the facts shown in Figure 6.14. These facts are generated by SQT-RS, and processed by the recommendation system developed in Jess. However, this is an example of the list of facts that are internally handled by SQT-RS. In the following sections the requirements that most influence the decision process are detailed.

Note that the parametrisation performed here was done to analyse the impact of the type of relay on the rest of parameters, using SQT-RS. This is a combination of knowledge-based

[2]For example, it is assumed that the parameter Energy increases/decreases when the parameter PowerConsumption decreases/increases.

```
(goal (priority 1) (criterion maximize) (parameter 20))
(op (todo decrease) (on 21) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 21) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 24) (to maximize) (p 20) (val 0.444444))
(internal-op (parameter 24) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 31) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 31) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 32) (to maximize) (p 20) (val 2.666667))
(internal-op (parameter 32) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 34) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 34) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 35) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 35) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 40) (to maximize) (p 20) (val 5.333333))
(internal-op (parameter 40) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 41) (to maximize) (p 20) (val 0.444444))
(internal-op (parameter 41) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 42) (to maximize) (p 20) (val 0.444444))
(internal-op (parameter 42) (request increase) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 43) (to maximize) (p 20) (val 0.444444))
(internal-op (parameter 43) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 44) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 44) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 45) (to maximize) (p 20) (val 0.444444))
(internal-op (parameter 45) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 46) (to maximize) (p 20) (val 2.222222))
(internal-op (parameter 46) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 47) (to maximize) (p 20) (val -0.888889))
(avoid 47 maximize 20 -0.888889)
(internal-op (parameter 47) (request decrease) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 48) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 48) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 49) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 49) (request decrease) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 50) (to maximize) (p 20) (val 2.666667))
(internal-op (parameter 50) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 54) (to maximize) (p 20) (val -22.666667))
(avoid 54 maximize 20 -22.666667)
(internal-op (parameter 54) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 55) (to maximize) (p 20) (val -22.666667))
(avoid 55 maximize 20 -22.666667)
(internal-op (parameter 55) (request increase) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 56) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 56) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 57) (to maximize) (p 20) (val 4.000000))
(internal-op (parameter 57) (request increase) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 61) (to maximize) (p 20) (val 2.666667))
(internal-op (parameter 61) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 71) (to maximize) (p 20) (val 1.333333))
(internal-op (parameter 71) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 80) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 80) (request decrease) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 82) (to maximize) (p 20) (val 0.888889))
(internal-op (parameter 82) (request decrease) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 83) (to maximize) (p 20) (val 2.222222))
(internal-op (parameter 83) (request decrease) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 87) (to maximize) (p 20) (val 2.666667))
(internal-op (parameter 87) (request increase) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 92) (to maximize) (p 20) (val -22.666667))
(avoid 92 maximize 20 -22.666667)
(internal-op (parameter 92) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 93) (to maximize) (p 20) (val 1.333333))
(internal-op (parameter 93) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 94) (to maximize) (p 20) (val 1.333333))
(internal-op (parameter 94) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 95) (to maximize) (p 20) (val 0.444444))
(internal-op (parameter 95) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 96) (to maximize) (p 20) (val 0.444444))
(internal-op (parameter 96) (request increase) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 97) (to maximize) (p 20) (val 33.333333))
(internal-op (parameter 97) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 98) (to maximize) (p 20) (val -33.333333))
(avoid 98 maximize 20 -33.333333)
(internal-op (parameter 98) (request increase) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 99) (to maximize) (p 20) (val 6.666667))
(internal-op (parameter 99) (request increase) (dependent 24) (to maximize) (target 20))(op (todo increase) (on 100) (to maximize) (p 20) (val 17.777778))
(internal-op (parameter 100) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 101) (to maximize) (p 20) (val 2.222222))
(internal-op (parameter 101) (request increase) (dependent 24) (to maximize) (target 20))(op (todo decrease) (on 102) (to maximize) (p 20) (val 3.111111))
(internal-op (parameter 102) (request increase) (dependent 24) (to maximize) (target 20))(parameter (id 20) (name "Energy") (layerP "LocalProperties") (typeP
"Resource"))
(parameter (id 21) (name "Memory") (layerP "LocalProperties") (typeP "Resource"))
(parameter (id 24) (name "PowerConsumption") (layerP "LocalProperties") (typeP "Performance"))
(parameter (id 31) (name "Mobility") (layerP "LocalProperties") (typeP "Characteristic"))
(parameter (id 32) (name "RelayClass") (layerP "LocalProperties") (typeP "Characteristic"))
(parameter (id 34) (name "AvailableTimeSlots") (layerP "LocalProperties") (typeP "Resource"))
(parameter (id 35) (name "BufferSize") (layerP "LocalProperties") (typeP "Resource"))
(parameter-instantiated (id 40) (name "TransmissionPower") (layerP "Interface") (typeP "Performance"))
(parameter (id 41) (name "ReceptionPower") (layerP "Interface") (typeP "Performance"))
(parameter (id 42) (name "TimeOn") (layerP "Interface") (typeP "Performance"))
(parameter (id 43) (name "TimeOff") (layerP "Interface") (typeP "Performance"))
(parameter-instantiated (id 44) (name "TransmissionCapacity") (layerP "Interface") (typeP "Performance"))
(parameter (id 45) (name "Rate") (layerP "Interface") (typeP "Performance"))
(parameter (id 46) (name "CooperativeJamming") (layerP "Interface") (typeP "Security"))
(parameter (id 47) (name "GaussianCodebooks") (layerP "Interface") (typeP "Security"))
(parameter (id 48) (name "GaussianNoise") (layerP "Interface") (typeP "Security"))
```

Figure 6.14: A section of a file of facts used in SQT-RS for 5G Green environments.

solutions that enable us to derive information about a particular scenario. It is possible to combine the different scenarios (UE, BDMR, BDFR, PSFR) to obtain information about an environment with all these types of relays working together. However, this is not done here because we wish to focus on scenarios that help the reader to easily understand how SQT-RS works with a large number of parameters.

6.3.1. Automatic selection of CPRM-based 5G environment

As a final improvement, the recognition of a specific scenario for 5G Green networks based on the parameters selected by a user is provided (Figure 6.13). This last module defines the input variables/parameters *Processing, MIMO, Storage, CE, Density, Trust and Confidentiality*, which are capabilities for relay classes in 5G scenarios according to [62]. The possible values for the different parameters, expressed as requirements to be satisfied or facts in the system, take fuzzy values given the membership (MS) functions *Low*, *Med*, and *High*. In the final system, there is a fuzzy-variable for each parameter defined as a requirement, and, by default, each variable (parameter) uses the same MS functions to map their values. Figure 6.15 shows the values taken by MS functions for the parameter Processing.

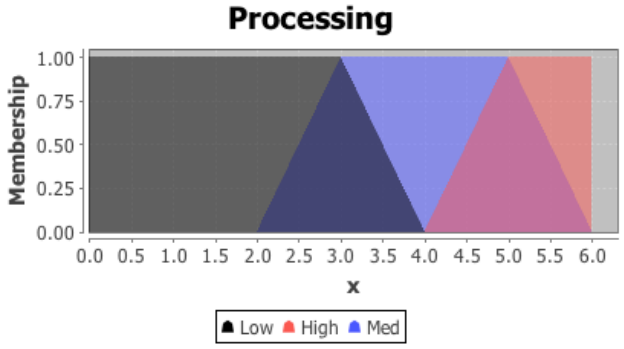


Figure 6.15: Fuzzy Membership Functions (ej. Processing)

These variables are inputs for a set of rules [3], it is possible to identify the 5G scenario that should be loaded in SQT (Figure 6.16). In particular, these scenarios are based on the relay class used: user equipment (UE), battery-dependent mobile relay (BDMR), battery-dependent fixed relay (BDFR) and power-supplied fixed relay (PWFR).

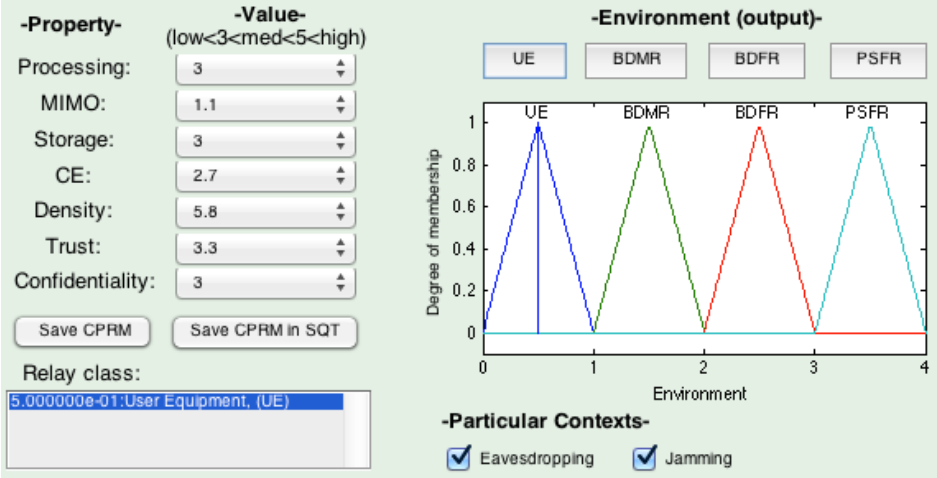


Figure 6.16: Automatic PC selection based on 5G capabilities.

The final aim of this component (Figure 6.16), is to simplify the use of SQT. Based on the result/output (UE, BDMR, BDFR, PWFR), the CPRM-based system is generated, and the PCs shown in Table 6.4 are integrated. The result is then included in SQT-RS.

6.3.2. Description of the CPRM-based 5G Green environment

SQT-RS provides recommendations based on a set of parameters and their relationships defined as part of a CPRM-based system. Hence, a general description of the parameters to be considered in a 5G Green environment and the assumptions for the analysis are provided below. This section is divided into three subsections, that consider, in this order: the general parameters and their relationships, specific contexts and the goals for the analysis.

[3] 65 rules are defined for inferring the results based on the values shown in Figure 6.16.

6.3.2.1. CPRM-based 5G Green model

The set of parameters considered in the analysis, classified in layers, is shown in Table 6.3. This is a cross-layer classification based on abstract relationships between parameters. Hence, the parameters are not classified based on physical layers, rather, we have based our approach on abstract layers[4]. For example, the *Secrecy Rate* can be considered as a physical security mechanism, however, in our classification, it is considered as part of the measurements layer because it is used as a measurement of the security at the physical layer in relay networks.

The parameters selected combine mobile platform parameters and relay network parameters. This selection is to provide a basic context with the definition of parameter candidates to be instantiated.

The parameters are grouped in layers according to Chapter 2. Moreover, the parameters of type resource are used based on the 3GPP Standarised QCI Attributes [131]. Specifically, the relationships between the parameters of performance type, and especially, those belonging to the measurements layer, can be extracted from widely known formulation. The relationships that cannot be extracted directly from mathematic formulation were defined, based on the results from the current related work, some of which are discussed in [62].

Finally, the composition of dependencies generates a graph of relationships between all the parameters in the CPRM-based system. Then, the operations defined in Table 5.1 can be applied to the set of parameters.

Moreover, the weights for some of the parameters in Table 6.3 are different, depending on the relay class analysed and the policy chosen. For example, the parameter UserExperience is more *relevant* in scenarios where UE are used. These factors are indicated in the information of the parameters given the environment, and this type of change in the weights is modelled through the GC script. The changes in the default values are indicated in Table 6.3, in parentheses just after the parameter involved. For example, UserExperience is targeted with a weight equal to 3 for UE relays.

6.3.2.2. Particular contexts

The specific contexts considered are shown in Table 6.4. In the following paragraphs, two specific contexts are added to the CPRM chosen in the previous step: Eavesdropping (E) and Jamming (J). Note that, in Table 6.3, Eavesdropping and Jamming are parameters of type *Threat* at layer *Environment*.

In the Eavesdropping context, we add new parameters to enhance the information in the models with information typically considered in eavesdropping scenarios [128, 129]. For example, the fading in the eavesdropper has to be considered for determining the secrecy rate. As this is a fading with additional relationships (given by the eavesdropper role), then, the parameter EavesdropperFading is added as an instance of Fading. Moreover, a new parameter (not shown) called NormalFading is added to inherit the default behaviour of the parameter Fading. In the same way, the eavesdropper's capacity produces the opposite effect

[4]SQT-RS allow the modification of layers, types and parameters. Therefore, the user can group the parameters in new layers or change their type.

Table 6.3: Parameters for a Base Context in 5G Green environments

	HIGH LEVEL REQUIREMENT
Resource type	Guaranteed bit rate (GBR), non GBR (N-GBR).
Security	Authentication, authorisation, accounting, confidentiality, integrity, non repudiation, trust, privacy.
QoE	Conversational, Interactive, Streaming, Background, User Experience (UE=6, BDMR=4, BDFR=0, PSFR=0).
Characteristic	Complexity, fault tolerant (BDFR=4, PSFR=6), availability (BDFR=4, PSFR=6), reliability.
	LOCAL PROPERTIES
Resource	Battery, Memory, Processing, Storage.
Performance	Node lifetime, power consumption.
Security	Anti-tampering, signature, encryption, Asymmetric Cryptography (AC), Symmetric Cryptography (SC), key generation, Reputation.
Characteristic	Mobility(UE=4, BDMR=6, BDFR=0, PSFR=0), relay class.
Threat	Misbehaviour (UE=3).
	COMMUNICATION
Resource	Available time-slots, buffer size.
Performance	Packet size, signal strength, data transmission, transmission time, transmission power (PSFR=4), reception power, time on, time off, transmission capacity (PSFR=4), rate.
Security	collaborative jamming, gaussian codebooks.
Characteristic	Multiple antennas, MIMO, successive relaying, half-duplex (HD), full-duplex (FD), decode and forward (DF), amplify-and-forward (AF), channel surfing, spatial retreats, Channel State Information (CSI) (BDFR=3, PSFR=6) availability, multimode.
Consequence	Retransmission, congestion, overhead, inter-relay interference (IRI).
	MEASUREMENTS
Performance	Max rate, min rate, transmitted codewords, Channel Estimation Accuracy (CE) , RTT, throughput, delay, jitter, packet loss, response time, bit-error rate (BER), Signal-to-noise ratio (SNR), Signal-to-interference ratio (SIR), Signal-to-interference-plus-noise ratio (SINR), packet sent ratio (PSR), packet delivery ratio (PDR).
Security	Secrecy rate, secrecy capability, secrecy outage probability (SOP).
Consequence	Outage probability.
	ENVIRONMENT
Characteristic	Density, participants, diversity, noise, channel symmetry, network lifetime, multi-path fading, eavesdropper fading, Handover.
Consequence	Error probability.
Threat	Denial of Service (DoS), Eavesdropping, Jamming.

on the secrecy capacity [129] regarding the normal behaviour defined for the Transmission Capacity.

Furthermore, the specific context for Jamming (J), is built based on the information given in [60], where a discussion on how to identify different types of jammers (constant, deceptive, random and reactive) based on the effect on the performance parameters *Packet Delivery Rate* (PDR) and *Packet Sent Ratio* (PSR) is provided. PDR is calculated as the number of packages received between the number of total packets sent. Therefore, it is a measure of the boundary of the channel. To the contrary, the PSR is a measure of packets sent by a legitime sender. The authors explain when, depending on the type of jamming, it

Table 6.4: Weights w_d for Relationships in the PCs

Context	Dependence			Weight
	Antecedent	Relationship	Consequent	w_d
Eavesdropping (E)	EavesdropperFading	c	SecrecyRate	1
	EavesdropperFading	nc	Eavesdropping	1
	EavesdropperCapacity	c	Eavesdropping	1
	EavesdropperCapacity	$\neg c$	SecrecyCapacity	1
Jamming (J)	ConstantJammer	+	DoS	4
	ConstantJammer	$\neg c$	PDR	3
	DeceptiveJammer	+	DoS	4
	DeceptiveJammer	+	TimeOn	4
	RandomJammer	+	DoS	2
	ReactiveJammer	+	DoS	3
	ReactiveJammer	+	TimeOn	3
	ReactiveJammer	i-	PSR	0
	ReactiveJammer	$\neg c$	PDR	5
	Jamming	c	PowerJamming	1
	PowerJamming	$\neg c$	SecrecyCapacity	1
	PowerJamming	$\neg c$	TransmissionCapacity	3

is better to use PDR or PSR to identify possible jammers.

In addition, according to [129], the channel capacity under the control of a jammer attacker is decreased in direct relation to the power transmission of the jammer. To consider this effect, the parameter PowerJamming is added in the second PC to inherit the general behaviour of TransmissionPower and add the behaviour relative to the jamming scenario. As in the eavesdropping context, an additional parameter was added to maintain the behaviour of TransmissionPower by default. However, whether Eavesdropping or Jamming, when new properties are added to a parameter defined in a CPRM, it is necessary to add the relationship between this parameter and the new one. This is done in the eavesdropping context, and in the jamming context for PowerJamming. Note that this is not required for instances of Eavesdropping or Jamming, because, in these cases, the new parameters are directly related to the previous parameters (inheritance).

Moreover, Figure 6.17 shows the effect of PowerJamming in TransmissionCapacity before the relationship $PowerJamming \xrightarrow{\neg c} TransmissionCapacity$ is added. This is because PowerJamming inherits the behaviour of Power, so, by default, PowerJamming impacts on the instances of TransmissionCapacity positively. To avoid this behaviour we introduce a negative complete ($\neg c$) relationship. Note that, when we do this, the model does not show the behaviour at the Jammer, but rather it concentrates on the effect of the Jammer in the model.

However, the Eavesdropper capacity is considered in the model because it is necessary to evaluate the effect of eavesdropping on the model. This requirement is related to the knowledge of global state information assumed in most related work.

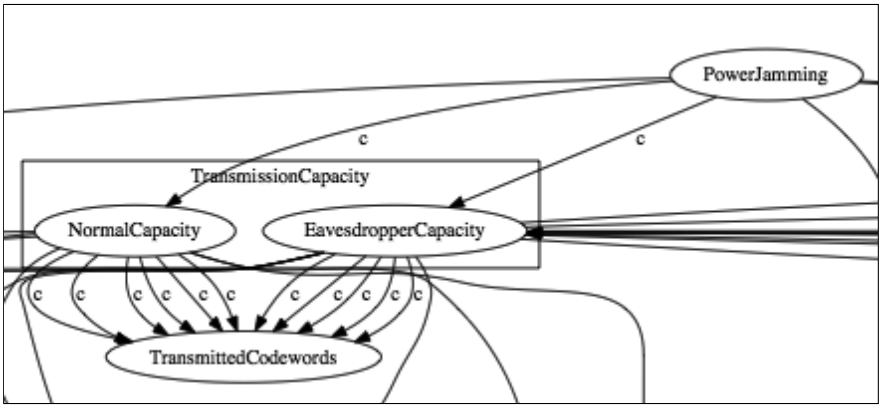


Figure 6.17: PowerJamming Influence (before $\neg c$).

6.3.2.3. Goals

The analysis is based on the selection of two goals. First, the parameter secrecy rate is selected to be maximised, given its relevance for this particular use case, according to the literature. In fact, other parameters such as secrecy capacity and secrecy outage probability (SOP), used as security metrics in several previous approaches, can be defined based on the secrecy rate [62]. So, as Figure 6.18 shows, the script used, when this parameter is increased, the probability for eavesdropping decreases, and then the confidentiality and privacy increase. Note that an alternative interpretation could be that when eavesdropping occurs, the secrecy rate will probably be poor. However, this depends on the type of eavesdropper and the properties defined. So, in this case, we have a protective outlook, in which the probability of misbehaviour decreases or is prevented when the security is improved.

Although the parametric trees for the secrecy rate (Figure 6.18) are similar in both cases (decreasing and increasing), the relationships from Confidentiality and Privacy to UserExperience are not considered in the case of increasing the parameters Confidentiality and Privacy. This is because it can be considered that a security failure in such terms may affect the UserExperience much more than the correct performance of the system, which, for many users, is taken to be normal. An alternative for modelling this difference is to use different weights for those cases in which an improvement in the security properties is not directly perceived by the user but the deterioration is.

In addition, the maximisation of energy is also considered. In this case, as energy is close to being a leaf node, this parameter will not affect a large number of parameters behind it. However, it does affect the node's lifetime, which is critical for the survivability of the node, and, therefore, given the ad-hoc nature of future networks, and moreover, relay networks, it is fundamental for the lifetime of relay nodes that are battery-dependent.

6.3.3. Recommendations and Conflicts

The parameters Secrecy Rate and Energy were chosen as goals as a proof of concept in the use of SQT-RS to assess Security and QoS tradeoff in 5G Green scenarios.

These parameters have different behaviours, and, therefore, the recommendations/ goals will also be different. Moreover, as different types of relays define different relevances for

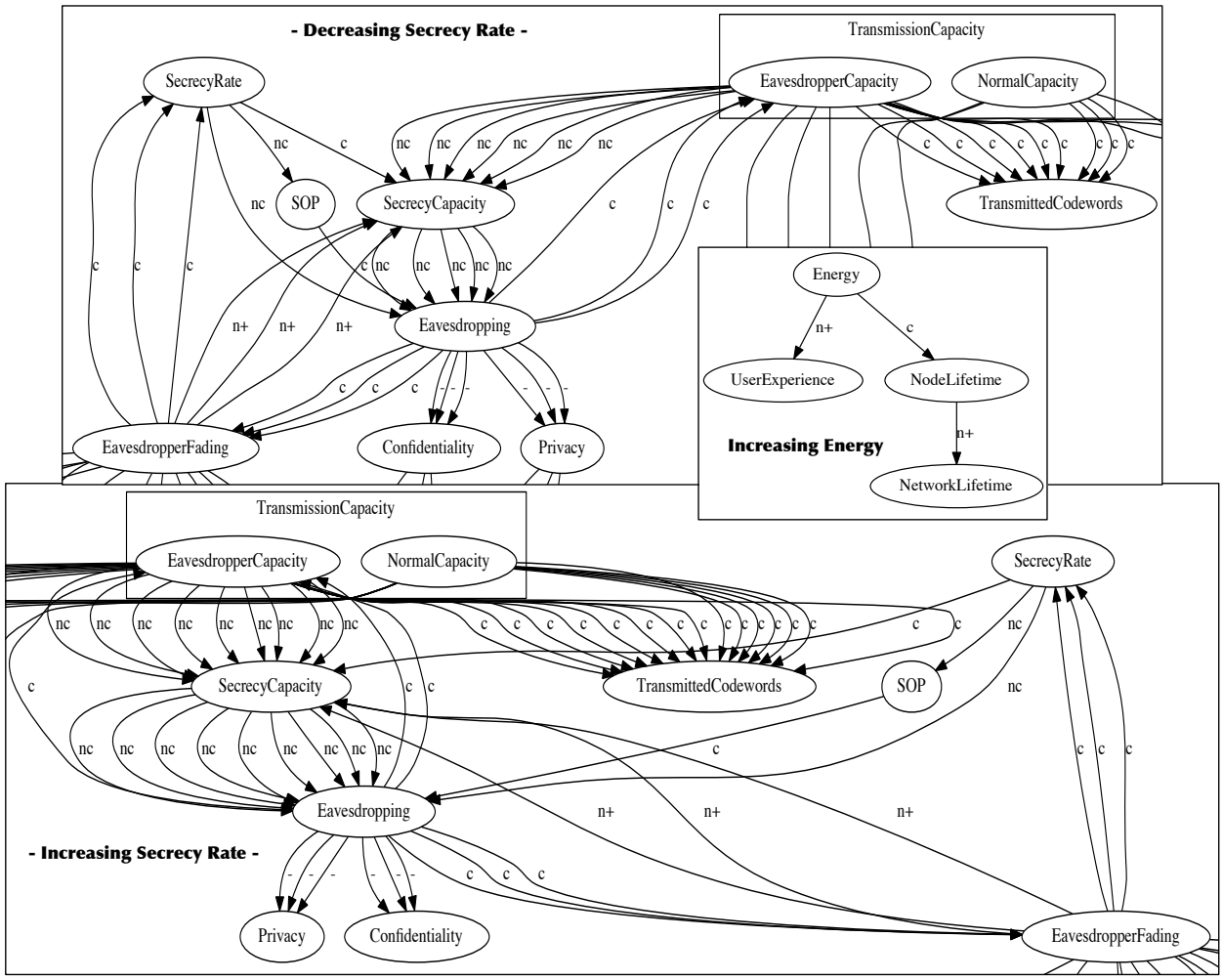


Figure 6.18: Sections of particular parametric trees.

the parameters, the results given for the different scenarios also vary with respect to the final effect that the recommendation produces. We now explain how SQT-RS handles these differences.

In this example, the environments are very similar with regard to the dependencies defined. When these environments are enhanced with information about the specific mechanisms for implementing the properties, the recommendations may vary considerably.

6.3.3.1. SecrecyRate

Figure 6.19 shows the relevant parameters for maximising the SecrecyRate. In this case, a reduced group of parameters are capable of performing changes in it, given our definition of the models. The list of parameters in the recommendation are without any particular order because in this case we do not order the subgoals.

As one of the main approaches to achieve the maximum secrecy rate is to gain an advantage over the eavesdropper by improving the quality of the channel, it is expected that the behaviour for CPRM-based non-limited relays will show improvements in the secrecy rate

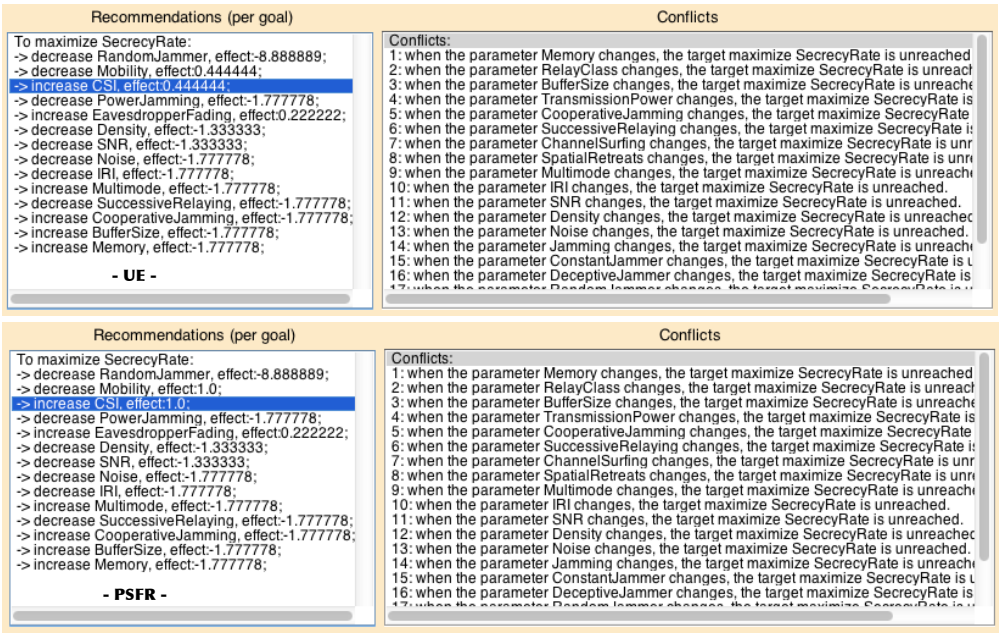


Figure 6.19: SecrecyRate - UE and PSFR.

at the expense of increasing the value of the parameters at communication layer to improve the quality in the measurements. This means that the best improvements in secrecy rate can be made by the PSFR relays. Moreover, the fixed relays (PSFR and BDFR) have the aditional advantage of preventing mobility. This is an advantage if we consider that global channel state information (CSI) is used to identify the quality of the channel when secrecy rate is calculated. So, this is the reason why the improvements for PSFR and BDFR are better than for mobile-based relays as UE and BDMR.

Moreover, as expected, the behaviour of BDMR and UE follows a similar approach. The main difference is to assume that in UE scenarios misbehaviour is likely, and that the user's experience is more relevant.

6.3.3.2. Energy

In Figure 6.20, the relevant parameters for maximising Energy are shown. In this case, only the results for UE are shown. This is because the value of the parameters relevant to Energy (left-hand side of Figure 6.20) are quite similar, so the recommendations for that are also similar. Furthermore, although PSFR is not battery-dependent, the relevance of this parameter is higher than 0, because in Green networks it is a valuable requirement.

Note that in our model the parameter that most affects Energy is modelled as PowerConsumption, and, in particular, in this example, the instance PowerNormal is the most relevant. The EavesdropperFading parameter has been related to Energy because in the recommendation it is the instance of Fading that best maximises the goal. Besides, there are a high number of conflicts defined (right side at Figure 6.20). This is because in the current version, all the conflicts between intermediate parameters are listed. For example, the conflict between FaultTolerant and Authentication. This conflict comes about when FaultTolerant is decreased, as Figure 6.21 shows. This behaviour is reasonable, because to make

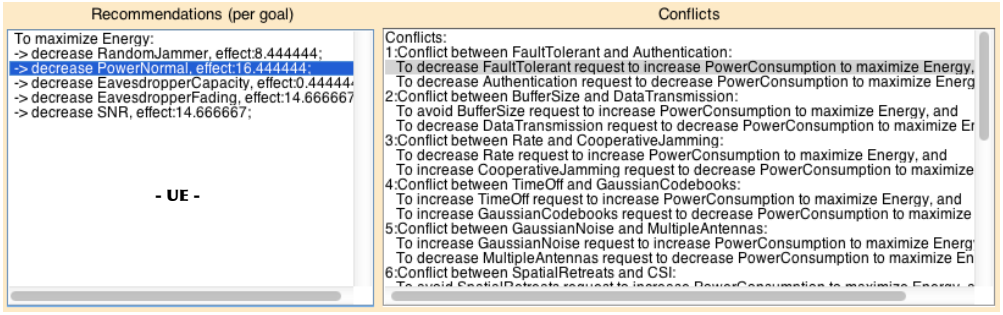


Figure 6.20: Energy - UE.

the system FaultTolerant requires the deployment of additional mechanisms which can take measurements and implement protocols to react to diverse events in real time. So, in the process of decreasing the FaultTolerant parameter, the Energy parameter is increased but only because there are additional mechanisms that will not be applied. This is not always desirable.

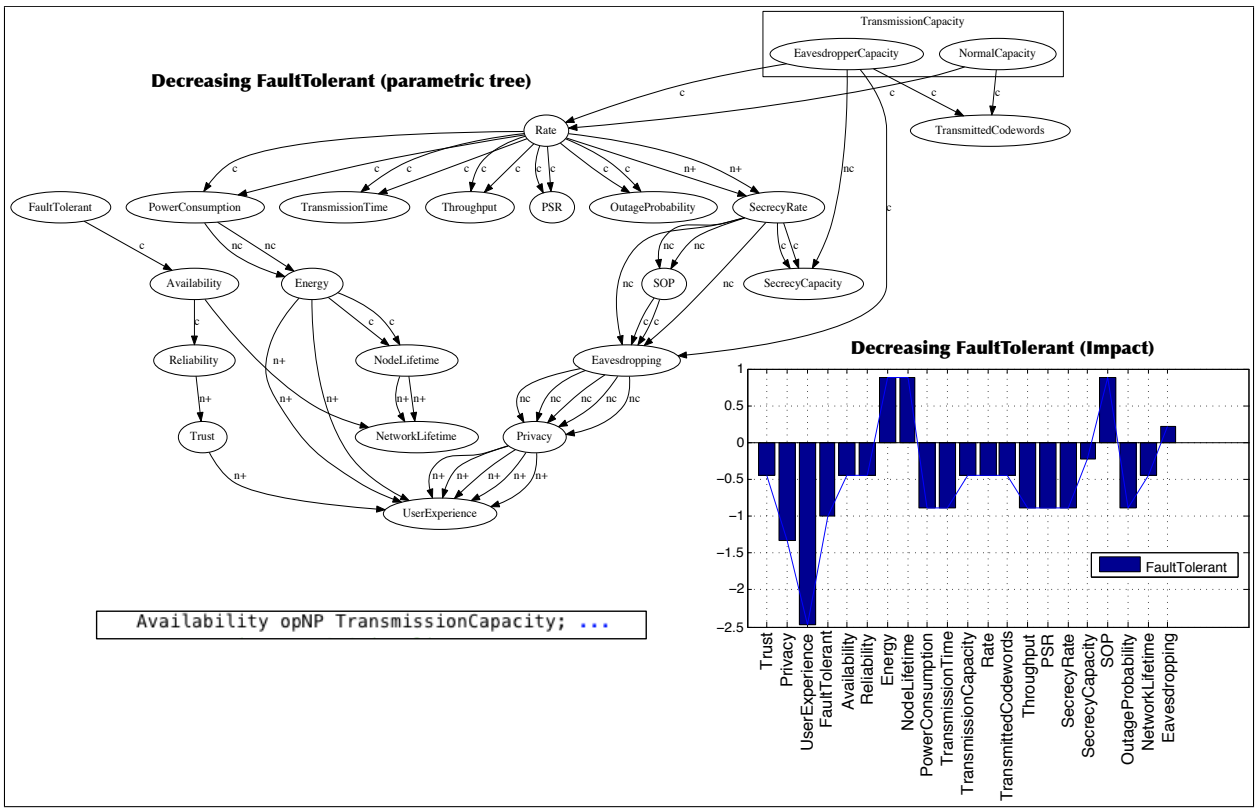


Figure 6.21: Decreasing FaultTolerant.

As can be observed in Figure 6.20, there is a long list of conflicts that affect PowerConsumption and Energy. In its current form, the list of conflicts is a log that shows information, in addition to the recommendations. Thus, it must be observed that although in the list of recommendations PowerNormal is considered as the most representative instance to maximise Energy, in the list of conflicts the parameter that appears is PowerConsumption,

because the behaviour is more general. If, in the list of conflicts only the instances are considered, then, the list of conflicts would be extremely long. Furthermore, when a parameter defines the same behaviour for several instances, the list of conflicts can be summarised using the instantiated parameters instead of the instances.

6.3.4. Additional Considerations

In what follows, additional considerations regarding the models used in the analysis for 5G Green networks are presented. The aim of this final section is to provide an overview of some of the issues that have to be considered in the design of CPRM-based systems. These are illustrated using some of the design issues that we found when defining the parametric sets used in the analysis.

6.3.4.1. User-oriented Approach.

In this clasification, the UE and BDMR are considered to have an important effect on the user's experience, and therefore the relevance of these parameters increases with respect to the rest of the cases.

One interesting issue found here, is that when mobility is influenced, the network lifetime always decreases. However, mobility is not directly related to this parameter, but rather it is related to the parameters of the network that affect the network lifetime and also with the reaction to attacks, such as moving the legitimate nodes to gain advantage over the malicious nodes (e.g. avoid jammers or eavesdroppers).

So, considering that the parameter - network lifetime - represents a conflict for any parameter which affects mobility as an intermediary parameter. Therefore, this behaviour particularly affects BDMR, where the parameter's mobility becomes much more relevant than others, because the devices have this capability.

6.3.4.2. Resource Independence.

The types of relays chosen have their own characteristics that, provided in the tool, give us information about the different behaviours depending on the relevance of the parameters chosen. For example, when PSFR is modelled, we do not worry about the user's experience, because we assume that the user at this level is the operator, and that the user's experience (in a traditional sense) in the UE should be more relevant than in the PSFR.

Therefore, the PSFR takes advantage because (i) has not to consider the user's perception and (ii) the resources for defense against the different attacks are not as limited as in the rest of the relays. However, network performance and availability are key at this point. When a fixed base station is a target, the mobility of the base station to take advantage of the signal, is not an option.

6.3.4.3. Threats

In this analysis, we have taken into account diverse parameters and relationships from external papers to include information on the impact of different threats in our configuration.

In this case, the parameters Eavesdropping, Jamming and MaliciousBehaviour have been considered as possible threats to the network.

Although both eavesdropping and jamming are considered as malicious behaviours, we differentiate these relationships, because, for example, in the case of Jamming, we consider cooperative jamming to prevent the malicious nodes from eavesdropping on the network. So, we differentiate between eavesdropping, jamming and other malicious behaviours that can affect the network.

6.3.4.4. Discussion about the Relationships

As mentioned, SQT-RS is a knowledge-based system. Specifically, SQT-RS depends on the relationships defined in the CPRM-based environment. Therefore, our choices of which relationships should be considered have a decisive impact on the final results. This should be carefully considered.

6.3.4.4.1. Eavesdropping. For example, one of the decisions that should be discussed is the decision about the relationships defined between SecrecyRate and Eavesdropping. As can be observed in Figure 6.18, we indicate that SecrecyRate influences Eavesdropping but the opposite relationship is not included.

In Figure 6.22 the parameter tree defined for an alternative interpretation between the relationships is considered. Given this parametric tree, when Eavesdropping is considered, either increasing or decreasing, the parameter EavesdroppingFading is affected. Therefore, two points should be considered here:

1. The relationship from Eavesdropping to EavesdropperFading. This implies that when there is eavesdropping activity then eavesdropper fading is present. However, this interpretation entails the risk of considering the performance in the channel of the eavesdropper as not good, and this is not necessarily true in all cases.
2. The interpretation of EavesdroppingFading and EavesdroppingCapacity. The changes depending on the placement in the relationships. In our analysis, it has been assumed that both parameters influence Eavesdropping, so both are in the antecedent. Although the existence of Eavesdropping implies that specific fading and capacity exist, the interpretation chosen allows the eavesdropping behaviour to be modelled given these parameters.

To easily understand the decisions taken for modelling the behaviour of 5G Green relay networks, Figure 6.22 is shown as an example of alternative behaviour for Eavesdropping. Specifically, Figure 6.22 shows the parametric tree of decreasing these parameters.

In more detail, related with (1), as the increasing of the EavesdroppingFading degradates the reception of the eavesdropper, this is considered as an advantage for the SecrecyRate. However, given the relationships, this means that decreasing the Eavesdropping activity, the EavesdroppingFading decreases and therefore the SecrecyRate also decreases, when really it should be increased, because the relevance of decreassing the eavesdropping activity should be higher than the eavesdropping fading, which is obviously irrelevant when eavesdropping is not present.

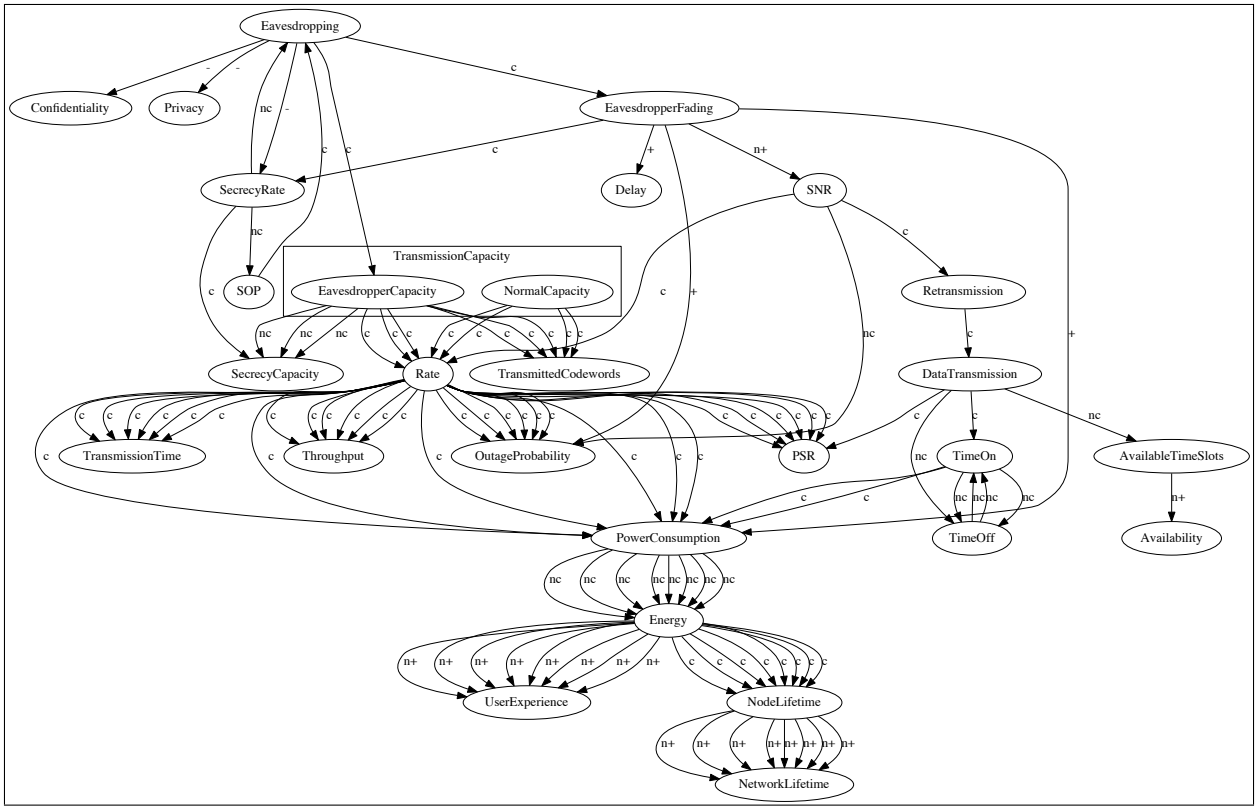


Figure 6.22: Decreasing Eavesdropping (example of an alternative case).

To solve these conflicts, the relationship between Eavesdropping and SecrecyRate should define higher weights than the relationship between EavesdropperFading and SecrecyRate.

Moreover, note that Figure 6.22 shows a double decrease in Eavesdropping. First, Eavesdropping is decreased to calculate the parametric tree, and once Eavesdropping has been decreased, SecrecyRate has increased, and, when SecrecyRate is increased the Eavesdropping is decreased. This is another way to handle the different effects between the parameters using the dependencies. However, given that the number of dependencies increases the complexity of the parametric tree, it is better to use the weights to define the behaviour of the environment.

Related to (2), this interpretation of EavesdroppingFading does not allow defining the quality or the probability of Eavesdropping given the quality of the channel of the eavesdropper. So, for this reason, in our approach we follow this approach.

6.3.4.4.2. Jamming. As detailed, PowerJamming redefines the relationship with TransmissionCapacity, not only does the weight, but also the instance redefine the operation with TransmissionCapacity, which by default is *c* because the relationship between TransmissionPower and TransmissionCapacity. However, this redefinition causes the following chain of dependencies:

1. Increasing PowerJamming triggers the decreasing on TransmissionCapacity. TransmissionCapacity is instantiated, so this effect is propagated to the instances Eavesdrop-

pingCapacity and NormalCapacity. This is logical, because the jamming affects both normal devices and eavesdroppers.

2. When EavesdroppingCapacity is decreased, Eavesdropping is also decreased (because these are related with a complete relationship *c*. When Eavesdropping is decreasing, SecrecyRate is increasing, because it is assumed that if there are not eavesdroppers, then SecrecyRate is maximum.

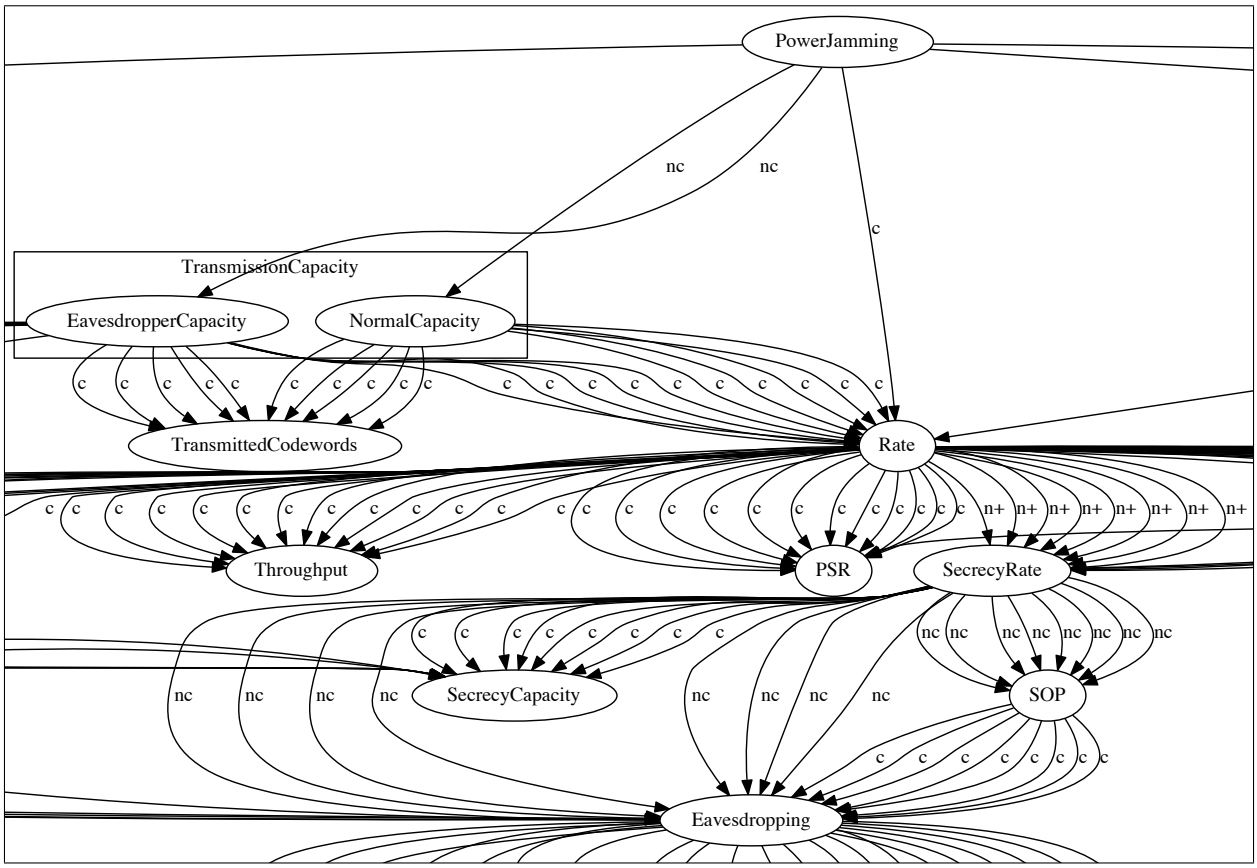


Figure 6.23: Increasing Power Jamming (subset of parametric tree).

However, SecrecyRate cannot be at maximum if it is impossible to send information because the network has collapsed due to Jamming. Then, environmental conditions have to be related to SecrecyRate and in this case decrease the positive impact on SecrecyRate. In our case, Rate is directly related to SecrecyRate as Figure 6.23 shows.

6.3.4.5. Discussion about the Weights

It is important to highlight that before testing the environments, the adjustment of the parameters and relationships was carried out in order to identify a reasonable behaviour between the different parameters.

During this phase, it is usual to identify some anomalies because the relationships defined have to be completed or adjusted by defining the adequate weights for the relationships.

For example, in preliminary results given for the goal *max. Energy*, the value of energy for the instances of Jamming *ConstantJammer* and *ReactiveJammer* increases when both threats are present in the network (increasing). This was considered to be an anomaly, because we understood that jamming should decrease Energy. A more detailed analysis showed that these results are positive because of the relationship between TimeOn (the time the antennas are required to be active) and PowerConsumption. These two parameters were related to the default value, and the difference was compensated because other values were positive.

In this case, the relationship between TimeOn and PowerConsumption was modified so as to increase the weight of the dependency. In general, these types of modifications are done, based on the parameters and their impact on the environment. When the analysis reveals that the compensation is justified, then, the result does not show an anomaly, rather it shows the real behaviour of the modelled system. However, if the parameters that are responsible for the behaviour (instead of the assumed normal behaviour) are not as relevant as those which determine the normal behaviour, then, the results may be considered as anomalies and will have to be properly handled before adding new parameters to the model.

As a result, an additional advantage of using the parameters defined in SQT-RS, is that the behaviour defined for the models has been proved and justified through several, meticulous analyses.

6.4. Summary

In this chapter, the usability of SQT has been tested, based on a use case, where the parameter Authentication was analysed in a Wireless Sensor Network (WSN), using a small set of coexisting mechanisms for its implementation. In this step, the different results for the system were taken into account using the tool, and analysed in order to show how the Security and QoS tradeoff process using SQT is carried out. Moreover, the steps for deploying SQT-RS to assess the Security and QoS tradeoff in 5G Green environemnts have been described. The solution has been tested with the specific use case of relay selection in 5G scenarios. Different types of goals are selected, and SQT returns recommendations for achieving the selected goals, given the facts extracted from the 5G Green system based on the Context-based Parametric Relationship Model (CPRM). The instances of the model to be evaluated are built based on the value of parameters selected by the user, which determine the type of relay used at any given moment, according to the parameters taken from previous results. Moreover, the instantiated model considers the behaviour, when there are eavesdroppers or jammers in the environment. The aim of this and the scenario chosen is to validate the usability of SQT-RS in FI. We have selected 5G Green relay networks because of their high diversity of parameters. In these scenarios, the focus is on the recommendations provided by the tool for the different contexts, decided by the type of relay/device and the objective criteria selected by the user. Note that SQT-RS was built considering those parts that may be enhanced as dynamic. It is not only the definition of the parameters and the concept of instantiation, but also the deployment architecture for SQT-RS, that enables the modification of the intermediary files used by the tool to deliver the decisions.

CHAPTER 7

Conclusions and Open Challenges

This chapter concludes this thesis. In the following sections, final remarks and future challenges to be addressed in the scope of Security and QoS tradeoff are discussed. These future directions and open challenges, although imposible to include in this thesis, we consider are sufficiently interesting to open new research lines.

7.1. Conclusions

The main objective of this thesis is to identify how to assess the security and QoS tradeoff, considering the requirements in FI environments. The diversity and heterogeneity in these environments suggest that the analysis of the security and QoS tradeoff has to be done at high-level, based on a set of characteristics and properties. So, in Chapter 1 we focused on a set of representative environments - candidate networks to be part of the FI - to identify the main topics and challenges to be addressed in the security and QoS tradeoff. In Chapter 2 we provided a classification of relevant parameters in these networks is provided, differentiating between specific and integration parameters. The chapter explicitly enumerates the main motivations for our study: the current approaches that analyze the security and QoS tradeoff do not consider (i) the diversity of parameters in FI environments, (ii) the subjectivity of the components, and (iii) are usually focused on a specific level of the analysis (e.g. at the service level in the composition of services, or at the physical level when physical security - secrecy rate - must be applied).

We advocate for assessing Security and QoS tradeoff based on the analysis of parametric relationships, considering different layers of abstraction, and separating the parameters based on their type, layer and any additional subjective values needed in the dynamic Future Internet. These parametric relationships define the dependencies between Security parameters and QoS parameters, but at different layers. Moreover, defining these relationships based on a context is essential in order to express their impact on the final configuration. The definition of the parameters, operations and the rest of the components and properties within a context is also a key issue in identifying their relevance in the final composition of elements in the environment. Our approach considers the composition of *things*. We are not only interested in the services, but also in the low-layer characteristics or technologies that can be used to increase coexistence and cooperation in the network between the Security and the QoS mechanisms.

Following this approach, in Chapter 3, two models for analysing the security and QoS tradeoff are provided: PRM (non-contextual) and CPRM (contextual, that includes PRM). Precisely, the Context-based Parametric Relationship Model (CPRM) address the problem of evaluating the tradeoff between security and QoS parameters, by defining the behaviour of the systems where the parameters play an important role. This model allows abstract parameters to be defined as the main properties that can be provided by dynamic mechanisms that can change at any given moment. The inclusion of dynamic behaviour in the model is one of the key issues that enables the model to be used for assessing security and QoS tradeoff in dynamically composed heterogeneous networks.

A proof of concept of this model was implemented in Chapter 4, were the *Security and QoS Tradeoff Tool* (SQT) was defined. This tool was defined and implemented in MATLAB for handling, dynamically, different types of parameters classified in abstract layers. SQT is able to handle CPRM-based components, that are scripts in which the behaviour of the parameters is described, and contexts, which contains the description of the new behaviour to be integrated inside the scripts.

In Chapter 5 a recommendation system is proposed to be integrated in SQT (SQT-RS). Such system provides recommendations in user-based language, and also helps the tool to

automatically interpret some results. In addition, to help to train the user, SQT-RS provides an additional functionality to generate CPRM-based description of environments based on a set of properties selected by the user, using a GUI.

Finally, in Chapter 6 two use cases were provided to validate our approach in representative scenarios of the FI.

The rest of this chapter is dedicated to highlight some of the points related with the objectives (**O1**–**O9**) addressed in this thesis, and to provide what we consider open challenges.

7.1.1. Security and QoS in the Dynamic and Heterogeneous FI

Our approach is based on the use of CPRM to represent the relationships between security and QoS parameters because it has been built, taking into account important issues identified in our analysis of Security and QoS tradeoffs in Future Internet (FI) scenarios:

Heterogeneous & Dynamic behaviour in FI environments. In order to assess the security and QoS tradeoff in heterogeneous environments, it must be considered that there are generic parameters (properties) and specific parameters (mechanisms) of security and QoS, and that these may change at any moment. Therefore, the mechanisms developed to assess the security and QoS tradeoff have to be able to handle different types of parameters and still work even when these have changed. So, the dynamic behaviour, that is, the multiple change of contexts have to be accepted and integrated as part of the mechanisms for evaluating the security and QoS tradeoff, because this behaviour is in the nature of FI environments, and because being dynamic is the best way to increase the lifetime of a solution.

Security and QoS parameters are related through different type of parameters. It is important to consider in the analysis, those intermediate parameters that are affected by security parameters, and whose modification affects the QoS parameters, and vice versa. Indeed, security and QoS parameters are not allways directly related in a point-to-point relationship. For example, as detailed in Chapter 6, the parameter Overhead may be activated by parameters related with Authentication mechanisms. To predict this type of cascade effect is very complex but expressing the environments as CPRM-based systems can help to better understand these scenarios, where multiple parameters are affected indirectly by others in different places of the dependence chain.

Security and QoS relationships are present at different abstract layers. So, security and QoS parameters have to be represented at different layers, and said layers may vary, depending on the requirements to be evaluated. The classification of parameters in different layers help in the analysis of parameters of different types taken together, because they belong to the same abstract layer. Moreover, as occurs in the previous cases, abstract layers may vary, for example whether is not useful to analyse the group of parameters together. Our schema considers the modification of these layers, and handles types, layers and even operations transparently.

Security and QoS are subjective. When the user is involved the value or relevance of a parameter may vary very quickly. Security loses its relevance when the lack of usability of the mechanisms makes the security unuseful. At this point, the user starts to find alternatives to make security *easy* and herein lies the problem. The problem itself does not lie with the user, but is in the way in which the tool is presented to the user, or prepared for him/her.

Furthermore, the relevance of the QoS may be circumstancial. For example, when a security problem occurs the user may forget the QoS for a brief period, depending on the problem. In general, a poor QoS is more detectable by the user than a poor security mechanism, which in many cases is only detectable when problems arise.

Considering the aforementioned requirements, identified after realising an *overview of the main topics and challenges to be addressed in the FI and the IoT related to security and QoS tradeoff* (**O1**), CPRM enables the security and QoS tradeoff of heterogeneous environments dynamically, at the abstract layer. In particular, the parameters in a CPRM are classified based on their type, layer and also differentiating between contexts (**O2**). Moreover, in the design the subjective value of parameters and the specific or objective values are considered (**O3**). Finally, CPRM is *a general model to describe scenarios in the FI based on their parameters and relationships, and provide a set of general parameters of security and QoS that can be adjusted to the needs of any user that makes use of the model* (**O4**).

7.1.2. Proof of Concept: Implementing CPRM-based Behaviour

As CPRM-based systems define the language for interpreting the composition of environments based on their parameters and relationships, the tool SQT implements the functions defined by CPRM and provides a proof of concept of this solution based on the knowledge (**O5**). This process has been carried out very carefully. Indeed, we consider that developing proofs of concept of the proposed models is crucial for identifying flaws in the model and for providing guarantees of their usability and utility.

The component-based architecture of SQT allows the validation of our proposal of incrementality. The final system is formed by a set of components that have been tested separately. These are MATLAB functions and, in general, these are grouped in different blocks depending on the functionality provided (non-contextual or contextual structures, etc.). The new modules, as SQT-RS use some of the functions defined for SQT, and SQT does not use the new behaviour defined in SQT-RS, which is handled by new GUIs. Moreover, as the key functions are defined as individual funcions in MATLAB, it is possible to modify these functions separatelly. Furthermore, the hardest part of implementing SQT is transparent to the user and is the implementation of the CPRM model's behaviour. This does not change, because is the definition of the model itself.

The dynamic behaviour of SQT is in the knowledge provided by the user. SQT has been defined and implemented to enable selected components to be dynamically modified. The dynamic behaviour of SQT lies in the set of parameters used. SQT provides the set of security and QoS parameters and relationships that we have tested. However, these can be modified and enhanced using the tool or directly throught the scripts defined according the model.

With the decision to separate the static part of SQT (the definition of the handler of components) and the dynamic part (the knowledge provided throught the parametric relationships), we provide a tool that implements the functionality of CPRM which is also able to increase in complexity when required to by the circumstances. So, it is expected that while the functionality of CPRM is the same, the definition of SQT does not requiere changing.

The functionality of SQT is given by the CPRM model. To modify the functionality provided by the tool, it is necessary to modify the sources of the code of SQT. This proof of concept has been implemented using MATLAB because it is a powerful tool for working with matrixes, and includes a component for defining fuzzy logic-based controllers, and makes definiting of GUIs very easy. Moreover, the use of MATLAB is widely extended and there is a lot of useful information provided by the community in this regard.

One of the key points to be highlighted is the definition of the behaviour of CPRM-based systems using known parameters, and, most importantly, parameters belonging to heterogeneous environemnts. Moreover, SQT not only provides a basic set of parameters for analysing the security and QoS tradeoff (**O4**), but also provides the tools for integrating/extracting contexts to be combined (**O6**), and supports the modification of the values and relationships between parameters in the model. Furthermore, SQT *provides a modular design of all the processes in order to allow the modification of any of the files, separately* (**O8**), so it is possible to enhance the tool by adding new modules written in MATLAB, whenever is required.

7.1.3. Defining basic Sets of Parameters and Relationships

The most complicated part of this thesis has been to properly define the relationships between the parameters (related to **O4**), that is, modelling the behaviour of the different systems used in our approach. Handling generic parameters and the instantiation process defined in CPRM-based systems has been very hard. However, once the behaviour of the parametric model is stable, a different test can be done and the current CPRM-based models can be downloaded, stored and modifed using SQT. Regarding this point, there are two main benefits in using SQT:

Security and QoS parameters widely tested. Future users can personalise their own environments using one existing set of parameters of security and QoS and diverse intermediate parameters related to these, whose behaviour has been previously tested. The basic set of parameters and relationships has been defined to establish the most stable part of the model, prove it and then provide the adequate mechanisms for modifying the information in a controlled environment.

The adaptation of new information is incremental and also simplifies the analysis of results. When all the parameters are handled together, the system becomes very complex. For this reason, we suggest that the basic set of parameters are highly controlled, and consider the most dynamic parameters to be instances of a parameter in the basic parametric set. Another possibility is to nullify the effect of the parameters that are not relevant at a given time, setting their weight of relevance to 0. We have adopted this solution for some parameters in the basic test during the experiments. However, when this becomes habitual for a parameter the parameter should be removed or even be considered as a dynamic parameter.

So, we provide a basic set of parameters because we assume that the networks are always changing and growing, and therefore mechanisms to provide the properties also change. Therefore, we think that it is unrealistic to assume that all these parameters can be grouped together and never change.

Instead, a basic set of parameters has been defined considering different trends. In our opinion, there is a set of steps to take when working with parametric relationships as we do:

1. Select a set of candidate networks and technologies.
2. Identify common sets of parameters and specific parameters, that are, the contexts.
3. Define the relationships between the parameters.
4. Test the environment and identify the anomalies.
5. Solve the anomalies: weights and other assumptions.

In the case of security and QoS tradeoffs, **the identification of anomalies is quite specific because of the use of subjective values**. The subjective weights are used for measuring the relevance of the parameters considered at a given time (e.g. the relevance of trust in the environment *home*). Therefore, for example, if in an environment the parameter timeOn[1] is set to 0, the modification (increasing or decreasing) of this parameter will does not affect the energy. Indeed, this, in a wired environment is not considered to be an anomaly. If this model is applied to a wireless environment the subjective value to timeOn must change. The same happens with the types of threats that are present in a network: some of them are more likely to affect a network than others. This is not only due to the network resources, but also because of the role of the user in these networks.

Some subjective values may vary and be set up as non-subjective values. This is understood to be a value taken from a measurable property in the network (e.g. the probability of packet loss in a network). Indeed, our definition of CPRM also allows the anomalies in the behaviour to be identified based on the known properties of the parameters.

Anomalies based on mathematical formulation. One invariant of our system can be set up on the definition of the behaviour between parameters that have a mathematical formulation available. For example, some parameters such as Throughput or Delay have been defined based on the mathematical formulation which describes their relationship with other parameters. So before setting up any subjective value in the environemnt these definitons have to be true.

Anomalies based on an expected behaviour. Not only can the mathematical formulation help, but also our own knowledge about the system to be modeled can identify anomalies. This may be complex though, depending on the experience of the user that is handling SQT and the number of parameters and dependencies to be handled.

Finally, what is considered an anomaly depends on the user and the final system to be modelled. The process of defining the set of parameters and relationships becomes more complex when the user is involved in the system to be modelled.

7.1.4. User-based Language & Interpretation of Results

A recommendation system for SQT has been defined and integrated into SQT, denoted as SQT-RS (**O7**). This module allows extracting information dynamically from CPRM-based environments loaded in SQT and inferring information about the modifications needed in the parameters of the system to achieve different objectives. Like SQT, SQT-RS provides a

[1]Time that the wireless interface still on.

modular design and can be modified to provide additional information not implemented in our prototype (**O8**).

Enhancing the usability of SQT. This final module for SQT has been proposed to simplify the use of SQT, because the visual representation of results provided are very difficult to understand for untrained users in security and QoS tradeoffs. Moreover, when SQT was developed it was assumed that the final aim would be that different developers could indicate their own mechanisms for specific parts of the schema, so the final user may end up working with parameters that do not understand, but an expert does.

5G Selector component. Moreover, an additional module has been developed to dynamically build CPRM-based systems based on the user's requirements. This module, which is used for 5G Green environments, can be exported to generic environments. The module has been built using fuzzy-logic, and, for this reason, the weights of our parameters are mapped to the values used by this component.

Multiple technologies. SQT-RS is implemented using MATLAB (to be integrated in SQT), Java (to build the .jar), and Jess for handling the file write in CLIPS from Java and to infer results. The file writen in CLIPS (.clp) can be modified and SQT-RS will still work without any problems, if the principles of design of SQT-RS are maintained. As a result, SQT-RS provides the final recommendations in a language that is understandable to the user, that can be improved by modifying the text in the file .clp without modifying SQT-RS.

7.1.5. Applicability to Future Heterogeneous Environements

In this thesis the SQT-RS has been proposed for assessing security and QoS tradeoffs. With the aim of analysing the suitability of our approach for the FI, two use-cases are considered (**O9**).

Security and QoS Tradeoff in Wireless Sensor Networks. In this use case the composition of final scripts from different contexts is analysed. To do this, the initial set of parameters considered is listed and a PC for defining authentication mechanisms in WSNs is provided. The results illustrate that the process of instantiation is able to modify the context of the basic set of parameters and provide valuable information about the security and QoS tradeoff. Moreover, in this use case it is demonstrated that by aggregating information in the model from different sources the final result is richer.

Security and QoS Tradeoff in 5G Green Relay Networks. This use case is proposed for testing the SQT-RS component. This scenario is selected for the great diversity of parameters in this type of network and because it suits FI environments very well. The role of the user is key in these environments. The results show that the recommendation system provides information understandable to the user, but that it is also possible to provide valuable feedback to SQT by analysing the results provided by the recommendations system.

In our opinion, the two use-cases considered are highly suitable for our approach. Both cases cover parameters at different abstract layers, and the presence of resource-restricted devices in a solution that is considering the security and QoS tradeoff together with parameters at a high-layer of abstraction. Furthermore, the use-cases have been chosen because they cover the candidate networks to be part of the FI according to our initial analysis in Chapter 1. Taking this into account and the definition of context in CPRM, and also the

dynamic nature of our solution, we consider that this approach is useful for assessing the security and QoS tradeoff in future heterogeneous environments.

Moreover, as mentioned in Chapter 3, our approach is extendable to various areas of study, because of the dynamic definition and integration of contexts. However our main objective is to use it to assess the security and QoS tradeoff, and for this reason our tool SQT-RS only provides models where these types of parameters and the intermediate parameters that affect them are described and no others. These others have to be integrated by the person using the tool.

7.1.6. Dissemination and Collaboration

It is our intention that our approach could be useful in the future. In this thesis the code of the tool is not provided because it is very extensive, but of course it will be properly shared by suitable means as soon as possible.

Moreover, part of the work presented in this paper has been published in diverse international conferences and in ranked international journals not only to justify the theoretical content, but also to share our approach and results, which we consider to be very interesting, and so get valuable feedback from the research community.

This step is very important because collaboration between different entities to provide valuable information for security and QoS tradeoff is only possible through shared information and identifying common needs to be addressed.

7.2. Open Challenges

In the following sections open challenges are discussed. On the one hand, in our opinion, it is essential to invest effort in developing mechanisms for handling information automatically, since they are the first step in adquisition of heterogeneous data for storing in big data bases. On the other hand, carrying out the security and QoS tradeoffs in resource-constrained devices is very interesting, but very difficult to accomplish in heterogeneous environments. In this thesis, we have provided a tool for assessing the Security and QoS tradeoff but this tool will be handled in a non limited device. Further steps should be taken to build these types of solutions into these devices. Security and QoS tradeoff embedded inside the resource-constrained devices is a clear future challenge.

7.2.1. Automatic Data Adquisition and Classification

The convergence of heterogeneous networks generates large amounts of data, from which information about the user's preferences, network performance, and QoS can be inferred.

These data are not discarded, instead, there are multiple companies that take advantage of this information to enhance the services offered to the user, or to identify possible threats or misbehaviour in the network. So, all this information is stored in data bases or in the cloud to be analysed by expert systems or system administrators, who use diverse technologies to deduce information, like for example, pattern matching techniques. Depending on the technologies involved and the ability to present information, human observation is also

considered, particularly in those systems where authomatic responses are not applicable, for example, in certain critical systems.

For example, our second use case is based on 5G Green relay systems because there are large amounts of data that could be useful to identify the effect that different technologies and configurations have on security and QoS. These effects, between parameters at different layers can be identified as dependencies, and all the dependencies and parameters at any given moment can be understood as the context of a system. Particularly, as 5G Green relay networks can involve, from users with personal devices, to computer infrastructures with powerful devices such as base stations or different kinds of relays, assessing the security and QoS tradeoff is very complex.

We have used parameters taken from current publications in this field. However, the ideal scenario would be the automatic extraction of useful information - parameters and their relationships, and varying weights - from the current systems where the information is stored. Although there are some initiatives for extracting information for marketing and similar aspects, for assessing security and QoS tradeoff a collaborative analysis on the behaviour of different networks is required, as is the permanent training of intelligent systems to learn about the most suitable configurations.

7.2.2. Built-in Security and QoS Tradeoff

CPRM-based systems are dependent on the information provided to the model to work. Therefore, as it is a knowledge-based model, the results are more specific and concise when there is more information (parameters and relationships) defined in the script for the evaluation. One problem with this, is that this also increases the number of facts to be processed by SQT-RS and increases the computation and the memory, thereby, limiting their applicability in real-time systems to assess the Security and QoS tradeoff. Additional issues are in the adaptation of this approach or similar approaches to be built in resource-constrained nodes to enable these devices to make their own decisions. We think that the scheme proposed for SQT-RS could be adapted to work taking into account the aforementioned issues, however this is beyond the scope of this thesis.

Some steps in this direction are addressed in Chapter 2, through the definition of the components in a node (Figure 2.3). However, the cost of implementing this solution and the lifetime of the node with such diverse components have not yet been evaluated.

Moreover, this could pave the way to an interesting discussion about how to prolong the usability of optimised hardware-based solutions and combine them with software updating capacity without increasing the cost of the final device. One of the problems of security components is that they severely limit the lifetime of the devices, and that security for applications requires frequent updates. How to provide up-to-date security without affecting the availability of sensors in industrial environments is crucial to wether or not this kind of solution is adopted.

7.2.3. Grain Fine Recommendation Systems

In Chapter 5 a recommendation system for our tool SQT, named SQT-RS, is provided. The basic instructions for deploying this solution for CPRM-based systems are given, and the

usability of this solution has been tested with a specific use case. However, we are convinced that the definition of SQT-RS has greater potential than demonstrated here. Once more information about the use of CPRM-based systems becomes available, the rules used by SQT-RS can be improved so as to allow grain fine recommendations, even dynamically, with respect to the characteristics of the CPRM-based environment.

These future developments, will be only possible if the first challenge proposed (Section 7.2.1) is properly addressed. Moreover, and most importantly, mechanisms need to be provided for sharing all the information handled by the diverse systems which is essential if these solutions are to be adopted. Security mechanisms are vital in this process, as where to define how the diverse data is shared and under what circumstances and after what modifications for preserving anonimity, privacy, confidentiality and other critial properties have to be defined.

APPENDIX A

MATLAB Scripts

In the following the scripts used to the use cases are included. The programming language is MATLAB. These scripts generate CPRM-based systems and PC components than can be loaded from SQT. Therefore, note that the scripts can be modified to be adapted to any other problem.

A.1. Use case 1: WSN

This use case is defined using the PRM, the default context, and the PC described in the following. In the case of the PC, the lines with comments (%) represents additional information that finally was not considerd in the analysis.

A.1.1. PRM

```
%============ GENERAL
PROPERTIES = 5; %minimum 5
NUM_PARAM = 48; %can be avoided (see example for 5G Green)
PRM = cell(5+NUM_PARAM+1, PROPERTIES);
%============
%---- Definition of Layers
NUM_LAYERS = 5;
PRM(1, 1:2) = {NUM_LAYERS ...
        {1 'HighLayerRequirements' 'red'; ...
         2 'LocalProperties' 'orange'; ...
         3 'Communication' 'blue'; ...
         4 'Measurements' 'green'; ...
```

```
          5 'Environment' 'pink'}};
hlrID = PRM{1,2}{1}; lpID = PRM{1,2}{2}; iID = PRM{1,2}{3};
mID = PRM{1,2}{4}; eID = PRM{1,2}{5};

% ---- Definition of Types
NUM_TYPES = 6;

% id, name, color_dot, shape_dot
PRM(2, 1:2) = {NUM_TYPES ...
        {1 'Class' 'coral' 'egg' [1 1 0]; ...
         2 'Performance' 'coral' 'trapezium' [1 0 1]; ...
         3 'Characteristics' 'cadetblue2' 'oval' [0 1 1]; ...
         4 'Security' 'brown1' 'box' [1 0 0]; ...
         5 'Attacks' 'brown3' 'septagon' [0 0 1]; ...
         %6 'QoE' 'cadetblue3' 'septagon' [0 1 0]; ...
         6 'Consequences' 'burlywood1' 'hexagon' [0.5 0.5 0.5]}};
qosID = PRM{2,2}{1};
perID = PRM{2,2}{2}; charID = PRM{2,2}{3};
secID = PRM{2,2}{4}; attID = PRM{2,2}{5}; %qoeID = PRM{2,2}{6};
consID = PRM{2,2}{6};

% ---- Definition of Operations (rel. A -> B)
positive = 1; % B increasing because A
negative = -1; % B decreasing because A
ntd = 0;%nothing to do: there is not effect on B

NUM_OP = 9;
PRM(3, 1:2) = {NUM_OP ...
        {1 '+' 0.3 positive ntd; ...
         2 '-' 0.6 negative ntd; ...
         3 't' 0.9 positive negative; ...
         4 'n+' 1.2 ntd negative; ...
         5 'n-' 1.5 ntd positive; ...
         6 'i+' 1.8 positive positive; ...
         7 'i-' 2.1 negative negative; ...
         8 'c' 2.4 positive negative; ...
         9 'nc' 3 negative positive}};

opP = PRM{3,2}{1}; opM = PRM{3,2}{2}; opT = PRM{3,2}{3};
opNP = PRM{3,2}{4}; opNM = PRM{3,2}{5}; opiP = PRM{3,2}{6};
opiN = PRM{3,2}{7}; opC = PRM{3,2}{8}; opNC = PRM{3,2}{9};

%Others, for example, the default directory to save the images generated
PRM(4) = {'prICC'}; PRM{4,2}='ICC''14';
%Number of parameters and properties
PRM(5, 1:2) = {NUM_PARAM PROPERTIES};

%---- Definition of Parameters:
PRM(6:(5+NUM_PARAM),1:PROPERTIES) = {
    %******* HIGH LAYER *******
    %Performance parameters
    hlrID perID 5 'Reliability' [];...
    hlrID perID 6 'Availability' [];...
    hlrID perID 7 'FaultTolerant' [];...
```

```
67      %Security
68      hlrID secID 8 'Authentication' [];...
69      hlrID secID 9 'Authorization' [];...
70      hlrID secID 10 'Confidentiality' [];...
71      hlrID secID 11 'Integrity' [];...
72      hlrID secID 13 'Trust' [];...
73      hlrID secID 14 'Privacy' [];...
74      %Consequences
75      %******* LOCAL PROPERTIES *******
76      lpID perID 19 'PowerConsumption' [];...
77      lpID perID 20 'Memory' [];...
78      lpID perID 20 'Rayleigh' []; ...
79      lpID perID 20 'Energy' [];...
80      lpID perID 20 'ComputationTime' [];...
81      %Security
82      lpID secID 28 'AntiTampering' [];...
83      lpID secID 29 'Encryption' [];...
84      lpID secID 30 'PublicKeyCryptography' [];...
85      lpID secID 31 'SymmetricCryptography' [];...
86      lpID secID 32 'SecureKeyExchange' [];...
87      lpID secID 33 'SecureKeyredistribution' [];...
88      lpID secID 34 'KeyGeneration' []; ...
89      lpID secID 34 'SignatureScheme' [];...
90      lpID secID 34 'Certificate' [];...
91      %******* COMMUNICATION *******
92      %Performance parameters
93      iID perID 35 'DataRate' []; ...
94      iID perID 36 'PacketSize' []; ...
95      iID perID 37 'SignalStrength' []; ...
96      iID perID 38 'DataTransmission' []; ...
97      iID perID 39 'TransmissionTime' []; ...
98      iID perID 40 'TransmissionPower' []; ...
99      %Characteristics
100     iID charID 41 'TimeSleeping' []; ...
101     iID charID 42 'RequiredTimeOn' []; ...
102     iID charID 42 'RoutingProtocol' []; ...
103     %Consequences
104     iID consID 45 'Retransmission' []; ...
105     %******* MEASUREMENTS *******
106     %Performance parameters
107     mID perID 47 'Throughput' [];...
108     mID perID 48 'Delay' [];...
109     mID perID 49 'Jitter' [];...
110     mID perID 50 'PacketLoss' [];...
111     mID perID 51 'ResponseTime' [];...
112     mID perID 52 'BER' [];...  %Bit error rate
113     %******* ENVIRONMENT *******
114     %Performance parameters
115     eID perID 54 'AllowableBandwidth' [];...
116     eID perID 55 'ErrorProbability' [];...
117     %Attacks
118     eID attID 58 'DoS' [];...
119     eID attID 60 'MaliciousDevices' [];...
120     %Consequences
```

```
121     eID consID 61 'Interference' [];...
122     eID consID 62 'Congestion' [];...
123     eID consID 63 'Overhead' [];...
124     eID consID 64 'Fading' [];...
125     eID consID 65 'Shadowing' [];...
126     eID consID 66 'Noise' [];...
127
128 };
129
130 %Assigning identifiers to parameters
131 for i=1:NUM_PARAM
132     eval(sprintf('%s_num=%d;', PRM{i+5,4},i)); %assigning ID
133     PRM{i+5,3} = i; %coherence
134     evalin('base', sprintf('%s_num=%d;', PRM{i+5,4},i)); % workspace
135 end
136
137 %----  Definition of Relationships
138
139 PRM{6+NUM_PARAM} = {
140     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
141     % HIGH LAYER REQUIREMENTS
142     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
143         Reliability_num opNP Trust_num; ...
144         Integrity_num opC Trust_num; ...
145         Reliability_num opNP Availability_num; ...
146         FaultTolerant_num opP Availability_num; ...
147         Authorization_num opP Confidentiality_num; ...
148         Authentication_num opC DataTransmission_num; ...
149         Authentication_num opC ResponseTime_num; ...
150         Authentication_num opNC Memory_num; ...
151         Authentication_num opP PacketSize_num; ...
152         Authentication_num opC SignatureScheme_num; ...
153         Privacy_num opC Trust_num; ...
154
155     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
156     % LOCAL PROPERTIES
157     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
158         SymmetricCryptography_num opP SecureKeyredistribution_num; ...
159         SymmetricCryptography_num opP SecureKeyExchange_num; ...
160         KeyGeneration_num opP PowerConsumption_num; ...
161         PublicKeyCryptography_num opP Encryption_num; ...
162         SecureKeyExchange_num opC PublicKeyCryptography_num; ...
163         SecureKeyredistribution_num opP SecureKeyExchange_num; ...
164         AntiTampering_num opP Trust_num; ...
165         Encryption_num opP Confidentiality_num; ...
166         SecureKeyExchange_num opP DataTransmission_num; ...
167         Encryption_num opC ResponseTime_num; ...
168         PowerConsumption_num opNC Energy_num; ...
169         SignatureScheme_num opC PowerConsumption_num; ...
170         SignatureScheme_num opC ComputationTime_num; ...
171         Rayleigh_num opNC Energy_num; ...
172         Rayleigh_num opNC AllowableBandwidth_num; ...
173         Memory_num opNC Overhead_num; %esta linea hace que .dot falle
174
```

```
175       %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
176       % INTERFACE
177       %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
178           RequiredTimeOn_num opNC TimeSleeping_num; ...
179           TimeSleeping_num opNC RequiredTimeOn_num; ...
180           DataRate_num opC TransmissionTime_num; ...
181           SignalStrength_num opNC TransmissionPower_num; ...
182           Retransmission_num opC DataTransmission_num; ...
183           DataTransmission_num opC RequiredTimeOn_num; ...
184           PacketSize_num opC TransmissionTime_num; ...
185           TransmissionTime_num opC RequiredTimeOn_num;...
186           RequiredTimeOn_num opP PowerConsumption_num; ...
187           TimeSleeping_num opNC PowerConsumption_num; ...
188           TransmissionPower_num opC PowerConsumption_num; ...
189           TimeSleeping_num opC ResponseTime_num; ...
190           DataRate_num opC Throughput_num; ...
191           DataRate_num opC Delay_num; ...
192           PacketSize_num opC Delay_num;...
193           DataTransmission_num opC Overhead_num; ...
194
195       %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
196       % MEASUREMENTS
197       %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
198           Delay_num opiP Jitter_num; ...
199           Delay_num opP PacketLoss_num; ...
200           PacketLoss_num opC Retransmission_num; ...
201           Throughput_num opNC AllowableBandwidth_num; ...
202           Throughput_num opC Overhead_num; ...
203
204
205       %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
206       % ENVIRONMENT
207       %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
208           Fading_num opP  ErrorProbability_num; ...
209           Congestion_num opM AllowableBandwidth_num; ...
210           Congestion_num opP ErrorProbability_num; ...
211           Shadowing_num opP Fading_num; ...
212           Noise_num  opC Interference_num; ...
213           Overhead_num opC Interference_num; ...
214           Interference_num opT ErrorProbability_num; ...
215           Noise_num opT ErrorProbability_num; ...
216           MaliciousDevices_num  opM Trust_num; ...
217           DoS_num opC Availability_num; ...
218           Congestion_num  opC PacketLoss_num; ...
219           ErrorProbability_num opP PacketLoss_num; ...
220           Interference_num opC  Delay_num; ...
221           ErrorProbability_num opC BER_num; ...
222           Overhead_num  opP ResponseTime_num; ...
223           Congestion_num opP ResponseTime_num; ...
224
225   };
```

A.1.2. PC: Authentication mechanisms in WSN

```
% Asumes the use of the script for PRM in Use Case 1.
CPRM = contextual(PRM);

%We use the identifiers in CPRM to be clearer
for i=1:CPRM{5,1}
    eval(sprintf('%s_num=%d;', CPRM{i+5,4},i));
end

%Get the next identifier, although this is irrelevant, these
%may be changed by the action rules.
IDnext = getNextID(CPRM);

PARAMS = {
%Authentication, in Yasmin et al.
[Authentication_num] 0 'CAS' 1;...
[Authentication_num] 0 'DAS' 1;...
[Authentication_num] 0 'IDS' 1;...
[Authentication_num] 0 'IMBAS' 1;...

%Signature scheme
[SignatureScheme_num] 0 'ECDSA' 1;...
[SignatureScheme_num] 0 'PairingBased' 1;...
[SignatureScheme_num] 0 'BNN' 1;...
};

%==========PC
PROPERTIES = 4; % Properties to parameters
NUM_PARAM = size(PARAMS,1);
PC = cell(1+NUM_PARAM+1, PROPERTIES);%+1:INICIAL,+1 DEPENDENCIAS

%---- INSTANCES DEFINITION:
PC(2:(1+NUM_PARAM), 1:PROPERTIES) = PARAMS;

%Use the identifiers:
for i=1:NUM_PARAM
    eval(sprintf('%s_num=%d;', PC{i+1,3},IDnext+i)); %asignar el valor id
    PC{i+1,2} = IDnext+i; %coherence
end

%Operations according CPRM:
opP = CPRM{3,2}{1}; opM = CPRM{3,2}{2}; opT = CPRM{3,2}{3};
opNP = CPRM{3,2}{4}; opNM = CPRM{3,2}{5}; opiP = CPRM{3,2}{6};
opiN = CPRM{3,2}{7}; opC = CPRM{3,2}{8}; opNC = CPRM{3,2}{9};


%---- Relationships:
RELATIONSHIPS ={
CAS_num opP ECDSA_num 1; ...
DAS_num opP ECDSA_num 1; ...
IDS_num opP PairingBased_num 1; ...
```

```
IMBAS_num opP BNN_num 1; ...

CAS_num opNC Memory_num 0; ...
DAS_num opNC Memory_num 5; ...
IDS_num opNC Memory_num 0; ...
IMBAS_num opNC Memory_num 0; ...

CAS_num opP PacketSize_num 5; ...
DAS_num opP PacketSize_num 1; ...
IDS_num opP PacketSize_num 3; ...
IMBAS_num opP PacketSize_num 4; ...
CAS_num opC Certificate_num 2; ...

ECDSA_num opNP Energy_num 1; ...
PairingBased_num opNP Energy_num 5; ...
BNN_num opNP Energy_num 4; ...

ECDSA_num opP ComputationTime_num 1; ...
PairingBased_num opP ComputationTime_num 5; ...
BNN_num opP ComputationTime_num 4; ...
};

PC{2+NUM_PARAM} = RELATIONSHIPS;

%Update dependencies:
NUM_DEPS = size(RELATIONSHIPS, 1); %size(PC{2+NUM_PARAM},1);

%PC identifier:
PC(1, 1:4) = {NUM_PARAM PROPERTIES NUM_DEPS {1 'Authe.SigSch.'}};
```

A.2. Use case 2: 5G Green

This use case is defined using the PRM, the default context, modifications performed from the 5G Selector defined in Chapter 6 and the two PCs detailed in the following sections.

A.2.1. PRM

```
%======================= BUILDING THE DATA
% This SCRIPT builds a Parametric Relationship Model (PRM), containing:
% 1.- Basic set of layers.
% 2.- Basic set of types.
% 3.- Basic set of operations.
% 4.- Default directory & Short description.
% 5.- Information about the number of parameters and properties.
% 6.- List of parameters and properties for each parameter.
% 7.- List of dependences between parameters.
%
% The final structure, denoted as PRM, is a cell structure.

```

```
%=========================================================================
%   INITIAL VALUES (NUMBER OF PARAMETERS & PROPERTIES OF PARAMETERS)
%=========================================================================
%PROPERTIES = 5; %not change

%=========================================================================
%   LAYERS
%=========================================================================
NUM_LAYERS = 5;
LAYERS = {NUM_LAYERS ...
        {1 'HighLayerRequirements' 'red'; ...
         2 'LocalProperties' 'orange'; ...
         3 'Communication' 'blUserExperience'; ...
         4 'Measurements' 'green'; ...
         5 'Environment' 'pink'}};

%--- LAYERS ID
higID = LAYERS{1,2}{1}; locID = LAYERS{1,2}{2}; intID = LAYERS{1,2}{3};
meaID = LAYERS{1,2}{4}; envID = LAYERS{1,2}{5}; comID = LAYERS{1,2}{6};

%=========================================================================
%   TYPES
%=========================================================================
NUM_TYPES = 8;
% id, name, color_dot, shape_dot
TYPES = {NUM_TYPES ...
        {1 'SLA' 'coral' 'egg' [1 1 0]; ...
         2 'Performance' 'coral' 'trapezium' [1 0 1]; ...
         3 'Characteristic' 'cadetblue2' 'oval' [0 1 1]; ...
         4 'Security' 'brown1' 'box' [1 0 0]; ...
         5 'Threat' 'brown3' 'septagon' [0 0 1]; ...
         6 'QoE' 'cadetblue3' 'septagon' [0 1 0]; ...
         7 'Consequence' 'burlywood1' 'hexagon' [0.5 0.5 0.5];...
         8 'Resource' 'coral' 'septagon' [0.9 0 1]}};

%--- TYPES ID
qosID = TYPES{1,2}{1}; perID = TYPES{1,2}{2}; chaID = TYPES{1,2}{3};
secID = TYPES{1,2}{4}; attID = TYPES{1,2}{5}; qoeID = TYPES{1,2}{6};
conID = TYPES{1,2}{7}; resID = TYPES{1,2}{8};

%=========================================================================
%   OPERATIONS
%=========================================================================
positive = 1; % Operation on A (increasing/decreasing) does B increasing
negative = -1; % Operation on A does B decreasing
ntd = 0;%nothing to do (operation on A does not influence B)

NUM_OP = 9;
OPERATIONS = {NUM_OP ...
        {1 '+' 0.3 positive ntd; ...
         2 '-' 0.6 negative ntd; ...
         3 't' 0.9 positive negative; ...
         4 'n+' 1.2 ntd negative; ...
         5 'n-' 1.5 ntd positive; ...
```

```
67          6 'i+' 1.8 positive positive; ...
68          7 'i-' 2.1 negative negative; ...
69          8 'c' 2.4 positive negative; ...
70          9 'nc' 3 negative positive}};
71
72 %--- OPERATIONS ID
73 opP = OPERATIONS{1,2}{1}; opM = OPERATIONS{1,2}{2}; opT = ...
       OPERATIONS{1,2}{3};
74 opNP = OPERATIONS{1,2}{4}; opNM = OPERATIONS{1,2}{5}; opiP = ...
       OPERATIONS{1,2}{6};
75 opiN = OPERATIONS{1,2}{7}; opC = OPERATIONS{1,2}{8}; opNC = ...
       OPERATIONS{1,2}{9};
76
77 %=========================================================================
78 %   PARAMETERS
79 %
80 %   layerID typeID id name [];
81 %=========================================================================
82 %IRI: Inter-relay Interference
83 %GBR: Guaranteed bit rate (GBR), Non GBR (NGBR).
84 %UserExperience: User Experience
85 %AC: Asymmetric Cryptography
86 %SC: Symmetric Cryptography
87 %SKD: Secure Key Distribuition
88 %HD: Half duplex
89 %FD: Full duplex
90 %AF: Amplify-and-forward
91 %CSI: Channel State Information
92 %CE: Channel Estimation Accuracy
93 %BER: Bit-error-rate
94 %SNR: Signal-to-noise ratio
95 %SIR: Signal-to-interference ratio
96 %SINR: Signal-to-interference-plus-noise ratio
97 %PSR: Packet sent ratio
98 %PDR: Packet delivery ratio
99 %SOP: Secrecy Outage Probability
100 %DoS: Denial of service
101
102 PARAM_INIT = {
103     %********** HIGH LAYER ************
104     %qos
105     higID qosID 1 'GBR' [];...
106     higID qosID 1 'NGBR' [];...
107     %Security
108     higID secID 1 'Authentication' []; ...
109     higID secID 1 'Authorization' []; ...
110     higID secID 2 'Accounting' []; ...
111     higID secID 2 'Confidentiality' []; ...
112     higID secID 2 'Integrity' []; ...
113     higID secID 2 'NonRepudiation' []; ...
114     higID secID 1 'Trust' []; ...
115     higID secID 2 'Privacy' []; ...
116     %qoe
117     %higID qoeID 3 'QoS' [];...
```

```
118     higID qoeID 3 'Streaming' [];...
119     higID qoeID 3 'Conversational' [];...
120     higID qoeID 3 'Background' [];...
121     higID qoeID 3 'Interactive' [];...
122     higID qoeID 3 'UserExperience' [];...
123     %characteristic
124     higID chaID 3 'Complexity' [];  ...
125     higID chaID 3 'FaultTolerant' [];  ...
126     higID chaID 3 'Availability' [];  ...
127     higID chaID 4 'Reliability' [];  ...
128     %********** LOCAL PROPERTIES ************
129     %Resource
130     locID resID 6 'Energy' [];...
131     locID resID 6 'Memory' [];...
132     locID resID 6 'Storage' [];...
133     locID resID 7 'Processing' [];...
134     %performance
135     locID perID 8 'NodeLifetime' [];...
136     locID perID 26 'PowerConsumption' [];...
137     %security
138     locID secID 14 'AntiTampering' [];  ...
139     locID secID 14 'Signature' [];  ...
140     locID secID 14 'Cyphering' [];  ...
141     locID secID 14 'AC' [];  ...
142     locID secID 14 'SC' [];  ...
143     locID secID 14 'KeyGeneration' [];  ...
144     locID secID 14 'Reputation' [];  ...
145     %characteristics
146     locID chaID 9 'Mobility' [];...
147     locID chaID 9 'RelayClass' [];  ...
148     %threats
149     locID attID 10 'Misbehaviour' [];...
150     %********** INTERFACE *************
151     %resource
152     locID resID 8 'AvailableTimeSlots' [];...
153     locID resID 8 'BufferSize' [];...
154     %performance
155     intID perID 15 'PacketSize' [];...
156     intID perID 15 'SignalStrength' [];...
157     intID perID 15 'DataTransmission' [];  ...
158     intID perID 15 'TransmissionTime' [];...
159     intID perID 16 'TransmissionPower' [];...
160     intID perID 15 'ReceptionPower' [];...
161     intID perID 15 'TimeOn' [];...
162     intID perID 15 'TimeOff' [];...
163     intID perID 18 'TransmissionCapacity' [];...
164     intID perID 19 'Rate' [];...
165     %security
166     intID secID 17 'CooperativeJamming' [];...
167     intID secID 17 'GaussianCodebooks' [];...
168     intID secID 18 'GaussianNoise' [];  ...
169     %characteristics
170     intID chaID 13 'MultipleAntennas' [];...
171     intID chaID 13 'MIMO' [];...
```

```
172     intID chaID 10 'SuccessiveRelaying' [];...
173     intID chaID 11 'HD' [];...
174     intID chaID 12 'FD' [];...
175     intID chaID 13 'AF' [];...
176     intID chaID 13 'ChannelSurfing' [];...
177     intID chaID 14 'SpatialRetreats' [];...
178     intID chaID 24 'CSI' [];...
179     intID chaID 25 'Multimode' [];...
180     %consequences
181     intID conID 15 'Retransmission' [];...
182     intID conID 15 'Congestion' [];...
183     intID conID 16 'Overhead' [];...
184     intID conID 17 'IRI' [];...
185     %********** MEASUREMENTS ************
186     meaID perID 31 'TransmittedCodewords' [];...
187     meaID perID 23 'CE' []; ...
188     meaID perID 23 'RTT' []; ...
189     meaID perID 23 'Throughput' []; ...
190     meaID perID 23 'Delay' []; ...
191     meaID perID 23 'Jitter' []; ...
192     meaID perID 23 'PacketLoss' []; ...
193     meaID perID 23 'ResponseTime' []; ...
194     meaID perID 23 'BER' []; ...
195     meaID perID 23 'SNR' []; ...
196     meaID perID 23 'SIR' []; ...
197     meaID perID 23 'SINR' []; ...
198     meaID perID 23 'PSR' []; ...
199     meaID perID 23 'PDR' []; ...
200     %security
201     meaID secID 5 'SecrecyRate' []; ...
202     meaID secID 4 'SecrecyCapacity' []; ...
203     meaID secID 4 'SOP' []; ...
204     %consequence
205     meaID conID 28 'OutageProbability' [];...
206     %********** ENVIRONMENT ************
207     %characteristic
208     envID chaID 34 'Density' [];...
209     envID chaID 34 'Participants' [];...
210     envID chaID 35 'Diversity' [];...
211     envID chaID 36 'Noise' [];...
212     envID chaID 37 'ChannelSymmetry' [];...
213     envID chaID 38 'Handover' [];...
214     envID chaID 32 'NetworkLifetime' [];...
215     envID chaID 32 'Fading' [];... %multipath fading
216     %security
217     envID secID 37 'SKD' []; ...
218     %consequence
219     envID conID 37 'ErrorProbability' [];...
220     %threat
221     envID attID 33 'DoS' [];...
222     envID attID 32 'Eavesdropping' [];...
223     envID attID 34 'Jamming' [];...
224 };
225
```

```
[NUM_PARAM, PROPERTIES] = size(PARAM_INIT);
if(PROPERTIES≠5)
    fprintf('[ERROR]>> This is not a PRM\n');
    exit(-1)
end

%============================
%  PRM INITIALIZATION
%============================
PRM = cell(5+NUM_PARAM+1, PROPERTIES);
PRM(2, 1:2) = TYPES;
PRM(1, 1:2) = LAYERS;
PRM(3, 1:2) = OPERATIONS;
PRM(5, 1:2) = {NUM_PARAM PROPERTIES}; %Parameters and properties
PRM(6:(5+NUM_PARAM),1:PROPERTIES) = PARAM_INIT;
%Others (for example, the default directory and a basic description)
PRM(4) = {'prRELAY'};PRM{4,2}='5G Relay';

%============================
%  IDENTIFIERS
%============================
%--- LAYERS ID
%higID = PRM{1,2}{1}; locID = PRM{1,2}{2}; intID = PRM{1,2}{3};
%meaID = PRM{1,2}{4}; envID = PRM{1,2}{5}; comID = PRM{1,2}{6};
%--- TYPES ID
%qosID = PRM{2,2}{1}; perID = PRM{2,2}{2}; chaID = PRM{2,2}{3};
%secID = PRM{2,2}{4}; attID = PRM{2,2}{5}; qoeID = PRM{2,2}{6};
%conID = PRM{2,2}{7};
%--- OPERATIONS ID
%opP = PRM{3,2}{1}; opM = PRM{3,2}{2}; opT = PRM{3,2}{3};
%opNP = PRM{3,2}{4}; opNM = PRM{3,2}{5}; opiP = PRM{3,2}{6};
%opiN = PRM{3,2}{7}; opC = PRM{3,2}{8}; opNC = PRM{3,2}{9};

%To use understandable IDs
for i=1:NUM_PARAM
    eval(sprintf('%s=%d;', PRM{i+5,4},i)); %asignar el valor id
    PRM{i+5,3} = i; %por si acaso, que haya coherencia
    evalin('base', sprintf('%s=%d;', PRM{i+5,4},i)); % workspace
end

%=========================================================================
% DEPENDENCES
%=========================================================================
%idA op idB; ...
PRM{6+NUM_PARAM} = {
    %=============
    %   HIGH LAYER
    %=============
    Reliability opNP Trust; ...
    Privacy opNP UserExperience; ...
    Trust opNP UserExperience; ...
    Reputation opC Trust;...
    Accounting opC NonRepudiation;...
    Streaming opC UserExperience; ...
```

```
280     Interactive opC UserExperience; ...
281     Conversational opC UserExperience; ...
282     Background opC UserExperience; ...
283     Conversational opC GBR;...
284     Interactive opC GBR;...
285     Streaming opC GBR;...
286     Streaming opC NGBR;...
287     Background opC NGBR;...
288     NonRepudiation opP Trust; ...
289     Integrity opC Trust; ...
290     Availability opC Reliability; ...
291     FaultTolerant opC Availability; ...
292     Confidentiality opNP UserExperience; ...
293     %--- interlayer
294     Authentication opC DataTransmission; ...
295     Authentication opC Handover;...
296     Authentication opC ResponseTime; ...
297     Authorization opC ResponseTime; ...
298     Authorization opC Confidentiality; ...
299     Streaming opP BufferSize; ...
300     Background opP PacketSize; ...
301     Availability opNP NetworkLifetime; ...
302     Availability opNP TransmissionCapacity; ...
303
304     %=============
305     %   LOCAL PROPERTIES
306     %=============
307     Processing opP Delay; ...
308     RelayClass opC MultipleAntennas; ...
309     RelayClass opC BufferSize; ...
310     PowerConsumption opNC Energy; ...
311     Energy opC NodeLifetime;...
312     KeyGeneration opP PowerConsumption; ...
313     Processing opP KeyGeneration; ...
314     AC opP Signature; ...
315     AC opP Cyphering; ...
316     SC opC Cyphering; ...
317     SC opP SKD; ...
318     %SKD opC KeyGeneration; ...
319
320     %--- interlayer...
321     Misbehaviour opNC Reputation; ...
322     Mobility opP ErrorProbability; ...
323     Mobility opP Handover; ...
324     Mobility opNC  CSI; ...
325     %Mobility opNC Jamming; ...
326     Mobility opP Misbehaviour; ...
327     Mobility opNC CSI;...
328     KeyGeneration opP Complexity; ...
329     KeyGeneration opC Integrity; ...
330     SC opP SKD; ...
331     SKD opC ResponseTime; ...
332     Cyphering opC ResponseTime; ...
333     Signature opC ResponseTime; ...
```

```
334     AntiTampering opP Trust; ...
335     Cyphering opC Confidentiality; ...
336     RelayClass opC Processing; ...
337     RelayClass opC CE; ...
338     RelayClass opC Density; ...
339     RelayClass opC Trust; ...
340     RelayClass opC Confidentiality; ...
341     NodeLifetime opNP NetworkLifetime; ...
342     AntiTampering opP Trust; ...
343     Cyphering opP Confidentiality;...
344     Energy opNP UserExperience; ...
345     BufferSize opNC Memory; ...
346
347     %=============
348     %   INTERFACE
349     %=============
350     ChannelSurfing opNC Jamming; ...
351     SpatialRetreats opNC Jamming; ...
352     Rate opC PowerConsumption; ...
353     Rate opC TransmissionTime; ...
354     Rate opC OutageProbability; ...
355     %Formulation data rate -- throughput
356     Rate opC Throughput; ...
357     Rate  opNP SecrecyRate; ...
358     %Formulation packetsize--transmission time
359     PacketSize opC TransmissionTime; ...
360     %Formulation packetsize-delay
361     PacketSize opC Delay; ...
362     BufferSize opC Diversity; ...
363     BufferSize opNC OutageProbability; ...
364     BufferSize opC Delay; ...
365     BufferSize opNM PacketLoss; ...
366     TransmissionPower opC PowerConsumption; ...
367     TransmissionPower opC SNR; ...
368     TransmissionPower opC TransmissionCapacity; ...%Zhu11
369     TransmissionPower opC Rate; ...
370     GaussianNoise opNC TransmissionCapacity; ... Zhu11
371     ReceptionPower opC PowerConsumption; ...
372     Retransmission opC DataTransmission; ...
373     DataTransmission opC TimeOn; ...
374     DataTransmission opNC TimeOff; ...
375     DataTransmission opNC AvailableTimeSlots; ...
376     DataTransmission opC PSR; ...
377     TimeOn opNC TimeOff; ...
378     TimeOn opC PowerConsumption; ...
379     TimeOff opNC TimeOn; ...
380     Rate opC PSR;...
381     AvailableTimeSlots opC TransmissionCapacity; ...
382     AvailableTimeSlots opNP Availability; ...
383     TransmissionCapacity opC TransmittedCodewords; ...
384     IRI opP Retransmission; ...
385     SuccessiveRelaying opP FD; ...
386     SuccessiveRelaying opC IRI; ...
387     TransmissionCapacity opC Rate; ...
```

```
388     CSI opC Complexity; ...
389     CSI opC SecrecyRate; ...
390     CSI opNC Memory; ...
391     CSI opC Storage; ...
392     CSI opC DataTransmission; ...
393     Multimode opC Complexity; ...
394     Multimode opNC IRI; ...
395     Multimode opNC OutageProbability; ...
396     Multimode opC TransmissionCapacity; ...
397     Multimode opC Rate; ...
398     MIMO opC MultipleAntennas;...
399     MIMO opC Rate; ...
400     MultipleAntennas opNC OutageProbability; ...
401     MultipleAntennas opC Diversity; ...
402     Congestion opP ErrorProbability; ...
403     ErrorProbability opC PacketLoss; ...
404     ErrorProbability opNC Reliability; ...
405     GaussianCodebooks opP CooperativeJamming; ...
406     GaussianCodebooks opC ResponseTime; ...
407     GaussianCodebooks opC GaussianNoise; ...
408     CooperativeJamming opNC Eavesdropping; ...
409     CooperativeJamming opNC DataTransmission; ...
410     CooperativeJamming opNC Noise; ...
411
412     %=============
413     %   MEASUREMENTS
414     %=============
415     ResponseTime opC Delay; ...
416     SNR opC Retransmission; ...
417     IRI opNC SNR; ...
418     IRI opNC TransmissionCapacity; ...
419     IRI opNC Rate; ...
420     SNR opC TransmissionCapacity; ...
421     SNR opC Rate; ...
422     SNR opNC OutageProbability; ...
423     Delay opiP Jitter; ...
424     Delay opC RTT; ...
425     RTT opC ResponseTime; ...
426     Delay opP PacketLoss; ...
427     %interlayer
428     Jitter opM Conversational; ...
429     Delay opM Conversational; ...
430     PacketLoss opM Streaming; ...
431     PacketLoss opNC PDR; ...
432     Jitter opM Streaming; ...
433     BER opM Interactive; ...
434     BER opM Background; ...
435     ResponseTime opNC UserExperience; ...
436     PacketLoss opC Retransmission;...
437     PacketLoss opC ErrorProbability;...
438     %Throughput opC Overhead; ...
439     SecrecyRate opC SecrecyCapacity; ...
440     SecrecyRate opNC Eavesdropping; ...
441     SecrecyRate opNC SOP; ...
```

```
442      SOP opC Eavesdropping; ...
443      %SecrecyCapacity opNC Eavesdropping; ...
444
445      %=============
446      %   ENVIRONMENT
447      %=============
448      %New relationships 11052014
449      Diversity opNC OutageProbability; ...
450      Diversity opNC PowerConsumption; ...
451      Diversity opC TransmissionCapacity; ...
452      Diversity opC Rate; ...
453      Fading opNP SNR; ...
454      Fading opP OutageProbability; ...
455      Fading opP PowerConsumption; ...
456      Fading opP Delay; ...
457      Fading opNP SecrecyRate; ...
458      Noise opC OutageProbability; ...
459      Noise opNC TransmissionCapacity; ...
460      Noise opNC Rate; ...
461      Noise opNC SNR; ...
462      Noise opC PowerConsumption; ...
463      ChannelSymmetry opC NetworkLifetime; ...
464      Participants opNC OutageProbability; ...
465      Participants opC Complexity; ...
466      Participants opC Delay; ...
467      %Formulation number of users, throughput
468      Participants opC Throughput; ...
469      Participants opC Overhead; ...
470      Density opC Diversity; ...
471      Density opC Participants; ...
472      Density opNC OutageProbability; ...
473      Density opC Eavesdropping;...
474      Density opC Misbehaviour; ...
475      Noise opC ErrorProbability; ...
476      Handover opC ResponseTime; ...
477      Handover opC Authentication; ...
478      %interlayer
479      DoS opM Availability;...
480      %Eavesdropping opNC Confidentiality; ...
481      Eavesdropping opNC Privacy; ...
482      Jamming opP DataTransmission;...
483      Jamming opP ErrorProbability; ...
484      ErrorProbability opNC PSR; ...
485      Jamming opNC PDR; ...
486      Jamming opNC SignalStrength; ...
487      Jamming opC Noise; ...
488      Jamming opC DoS; ...
489      };
490
491  %==== BUILD THE GENERAL TREE
492  %call to graphViz ---> install graphViz
493  %createFunctionDependencyDotFile(PRM,1);
```

A.2.2. PC: Eavesdropping

```
%Particular Context (PC) for eavesdropping.
%================================================================================
%This identifier will be updated by SQT.
IDnext = getNextID(CPRM);

%[parentID] id 'name' weight.
PARAMS = {
    [Fading] 0 'EavesdropperFading' 1;...
    [Fading] 0 'NormalFading' 1; ...
    [TransmissionCapacity] 0 'EavesdropperCapacity' 1; ...
    [TransmissionCapacity] 0 'NormalCapacity' 1; ...
};

%===========DEFINICION DE LA MATRIZ PC
% 0.- INITIAL ROWS
% ------------------
PROPERTIES = 4; %las proiedades de los parametros
NUM_PARAM = size(PARAMS,1);
PC = cell(1+NUM_PARAM+1, PROPERTIES);%+1:INICIAL,+1 DEPENDENCIAS

%
% 1.- SPECIFIC PARAMETERS
% ------------------
PC(2:(1+NUM_PARAM), 1:PROPERTIES) = PARAMS;

%Work with the identifiers defined in this script.
for i=1:NUM_PARAM
    eval(sprintf('%s=%d;', PC{i+1,3},IDnext+i)); %asignar el valor id
    PC{i+1,2} = IDnext+i; %por si acaso, que haya coherencia
    %evalin('base', sprintf('%s_num=%d;', CPRM{i+5,4},i)); % workspace
end

%Get the value defiened to the operations.
opP = CPRM{3,2}{1}; opM = CPRM{3,2}{2}; opT = CPRM{3,2}{3};
opNP = CPRM{3,2}{4}; opNM = CPRM{3,2}{5}; opiP = CPRM{3,2}{6};
opiN = CPRM{3,2}{7}; opC = CPRM{3,2}{8}; opNC = CPRM{3,2}{9};

%paramID op paramID weight;
RELATIONSHIPS ={
    %NormalFading opNC SecrecyRate 1;...
    EavesdropperFading opC SecrecyRate 1;...
    EavesdropperFading opNC Eavesdropping 1; ...
    EavesdropperCapacity opC Eavesdropping 1; ...
    EavesdropperCapacity opNC SecrecyCapacity 1; ... %Zhou11
};

PC{2+NUM_PARAM} = RELATIONSHIPS;

%update dependencies
NUM_DEPS = size(RELATIONSHIPS, 1); %size(PC{2+NUM_PARAM},1);
```

```
51
52 %the identifier (id) is in possition (1, 4)
53 PC(1, 1:4) = {NUM_PARAM PROPERTIES NUM_DEPS {1 'Eavesdropping'}};
```

A.2.3. PC: Jamming

```
1  %Particular Context (PC) for jamming, based on the paper "Jamming
2  %Sensor Networks: Attack and Defense Strategies", Wenyuan Xu et al.
3  %=========================================================================
4  %This identifier will be updated by SQT.
5  IDnext = getNextID(CPRM);
6
7  %[parentID] id 'name' weight.
8  PARAMS = {
9      [Jamming] 0 'ConstantJammer' 1;...
10     [Jamming] 0 'DeceptiveJammer' 1;...
11     [Jamming] 0 'RandomJammer' 1;...
12     [Jamming] 0 'ReactiveJammer' 1;...
13     [TransmissionPower] 0 'PowerJamming' 1; ...
14     [TransmissionPower] 0 'PowerNormal' 1; ...
15 };
16
17 %===========DEFINICION DE LA MATRIZ PC
18 % 0.- INITIAL ROWS
19 % ------------------
20 PROPERTIES = 4; %las proiedades de los parametros
21 NUM_PARAM = size(PARAMS,1);
22 PC = cell(1+NUM_PARAM+1, PROPERTIES);%+1:INICIAL,+1 DEPENDENCIAS
23
24 %
25 % 1.- SPECIFIC PARAMETERS
26 % ------------------
27 PC(2:(1+NUM_PARAM), 1:PROPERTIES) = PARAMS;
28
29 %Working with the identifiers:
30 for i=1:NUM_PARAM
31     eval(sprintf('%s=%d;', PC{i+1,3},IDnext+i)); %asignar el valor id
32     PC{i+1,2} = IDnext+i; %por si acaso, que haya coherencia
33     %evalin('base', sprintf('%s_num=%d;', CPRM{i+5,4},i)); % workspace
34 end
35
36 %Working with the value of the operations:
37 opP = CPRM{3,2}{1}; opM = CPRM{3,2}{2}; opT = CPRM{3,2}{3};
38 opNP = CPRM{3,2}{4}; opNM = CPRM{3,2}{5}; opiP = CPRM{3,2}{6};
39 opiN = CPRM{3,2}{7}; opC = CPRM{3,2}{8}; opNC = CPRM{3,2}{9};
40
41 % paramID op paramID weight;
42 RELATIONSHIPS ={
43     ConstantJammer opP DoS 4; ...
44     ConstantJammer opNC PDR 3; ...
45     DeceptiveJammer opP DoS 4; ...
```

```
46      DeceptiveJammer opP TimeOn 4; ...
47      RandomJammer opP DoS 2; ...
48      ReactiveJammer opP DoS 3; ... %response time for reactive jammer.
49      ReactiveJammer opP TimeOn 3; ...
50      ReactiveJammer opiN PSR 0; ...%Reactive jammer does not affect the PSR.
51      ReactiveJammer opNC PDR 5; ...%Reactive jammer makes 0 PDR (affects ...
            so much).
52      Jamming opC PowerJamming 1; ...
53      PowerJamming opNC SecrecyCapacity 1; %Zhu11
54      PowerJamming opNC TransmissionCapacity 1;
55  };
56
57  PC{2+NUM_PARAM} = RELATIONSHIPS;
58
59  %Update dependencies
60  NUM_DEPS = size(RELATIONSHIPS, 1); %size(PC{2+NUM_PARAM},1);
61
62  %Identifier in possition (1, 4)
63  PC(1, 1:4) = {NUM_PARAM PROPERTIES NUM_DEPS {1 'Jamming'}};
```

APPENDIX B

CLIPS code

The following source code writen in CLIPS is used in Chapter 6. This is a simplification of the schema proposed in said chapter. The simplification consists on avoid the analysis of the subgoals in rules r31-B and r32-B.

```
1
2  ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3  ;TEMPLATES
4  ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
5  ;Parameter:
6  ;All the parameters (1) are described based on their id and name,
7  ;(2) belongs to a layer and are of a particular type.
8  (deftemplate parameter "Parameter (not instantiated and not instance)"
9      (slot id (type INTEGER))
10     (slot name (type STRING))
11     (slot layerP (type STRING))
12     (slot typeP (type STRING))
13     (slot value (default very-low) (allowed-values very-low low low-med ...
           med med-high high)))
14
15 ;Instance:
16 ;In a CPRMi, the parameters can be instantiated or instances of
17 ;parameters. An instance is a parameter that implements to another
18 ;parameter.
19 (deftemplate parameter-instance "Instance for a parameter (mechanism)"
20     (slot id (type INTEGER))
21     (slot name (type STRING))
22     (slot layerP (type STRING))
23     (slot typeP (type STRING))
24     (slot id-parent (type INTEGER))
```

```
25      (slot value (default very-low) (allowed-values very-low low low-med ...
            med med-high high)))
26
27  ;Instantiated:
28  ;If exists an instance defined for a parameter, the parameter is
29  ;Instantiated.
30  (deftemplate parameter-instantiated "Parameter instantiated"
31      (slot id (type INTEGER))
32      (slot name (type STRING))
33      (slot layerP (type STRING))
34      (slot typeP (type STRING))
35      (slot value (default very-low) (allowed-values very-low low low-med ...
            med med-high high)))
36
37  ;Goal:
38  ;To define a goal: (priority == id of the goal)
39  (deftemplate goal
40      (slot priority (type INTEGER))
41      (slot criterion (allowed-values maximize minimize))
42      (slot parameter (type INTEGER)))
43
44  ;Goal related with an instantiated-parameter
45  (deftemplate subgoal
46      (slot priority (type INTEGER))
47      (slot criterion (allowed-values maximize minimize))
48      (slot parameter (type INTEGER)))
49
50  ;Individual recommendation (operations):
51  ;Recommendations are given as simple facts:
52  ; (op incre 2 (max 3 val 5))
53  ; (op (todo increase) (on 2) (to maximize) (p 3) (val 5))
54  (deftemplate op "Individual recommendation"
55      (slot todo (allowed-values increase decrease change))
56      (slot on (type INTEGER))
57      (slot to (allowed-values maximize minimize))
58      (slot p (type INTEGER))
59      (slot val (type FLOAT)))
60
61  ;Recommendation:
62  ;The inference process given by the rules will provide complex
63  ;recommendations based on the goals. The next template is for the
64  ;composed recommendation list
65  (deftemplate recommendation-set "Shows recommendations for a goal"
66      (slot goal (type INTEGER))
67      (slot toprint (type STRING))
68      (slot toMATLAB (type STRING)))
69
70  ;Recommendation for a subgoal:
71  ;Stores max and minimum values
72  (deftemplate subgoal-recomm "Shows recommendtions for a subgoal"
73      (slot parameter (type INTEGER))
74      (slot priority (type INTEGER))
75      (slot toprint (type STRING))
76      (slot toMATLAB (type STRING))
```

```
77      (slot min (type FLOAT))
78      (slot max (type FLOAT)))
79
80  ;Define the intermediary operations in the tree for a
81  ;parameter to identify the conflicts.
82  (deftemplate internal-op "Internal operation to achieve a goal"
83      (slot todo)
84      (slot parameter)
85      (slot request (allowed-values increase decrease))
86      (slot dependent)
87      (slot to)
88      (slot target))
89
90  ;Conflicts
91  (deftemplate conflicts "Conflicts between operations in the chain"
92      (slot toprint (type STRING))
93      (slot idn (type INTEGER))
94      (slot toMATLAB (type STRING)))
95
96  ;=========================================================
97  ;FUNCTIONS
98  ;=========================================================
99  ; no functions
100
101 ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
102 ;RULES
103 ;A: Normal (non-instantiated, non-instance)
104 ;B: Instantiated
105 ;C: Instance
106 ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
107 ;first assumptions: (current null) initially.
108 ;=========================================================
109 ;1.- Rules that can be done in the first step
110 ;=========================================================
111 ;:::::: Search for conflicts
112 (defrule init-conflict
113     (phase conflict)
114     (not (exists (conflicts)))
115     =>
116     (assert (conflicts (toprint "
117     Conflicts:
118     ") (idn 1) (toMATLAB "{"))))
119
120 (defrule add-conflict-1
121     (phase conflict)
122     ?io1 <- (internal-op (todo ?opp1) (parameter ?p1) (request ?op) ...
            (dependent ?pd) (to ?tar) (target ?idp))
123     ?io2 <- (internal-op (todo ?opp2) (parameter ?p2) (request ?op2) ...
            (dependent ?pd) (to ?tar) (target ?idp))
124     (goal (criterion ?c) (parameter ?idp) (priority ?g))
125     (oposite ?op ?op2)
126     (or (parameter (id ?p1) (name ?n1))
127         (parameter-instance (id ?p1) (name ?n1))
128         (parameter-instantiated (id ?p1) (name ?n1)))
```

```
129      (or (parameter (id ?p2) (name ?n2))
130          (parameter-instance (id ?p2) (name ?n2))
131          (parameter-instantiated (id ?p2) (name ?n2)))
132      (or (parameter (id ?idp) (name ?nidp))
133          (parameter-instance (id ?idp) (name ?nidp))
134          (parameter-instantiated (id ?idp) (name ?nidp)))
135      (or (parameter (id ?pd) (name ?npd))
136          (parameter-instance (id ?pd) (name ?npd))
137          (parameter-instantiated (id ?pd) (name ?npd)))
138      ?conf <- (conflicts (toprint ?tp) (toMATLAB ?tm) (idn ?k))
139      =>
140      (bind ?tp (str-cat ?tp ?k ":Conflict between " ?n1 " and " ?n2 ":
141                  To " ?opp1 " " ?n1 " request to " ?op " " ?npd " to " ...
                        ?tar " " ?nidp ", and
142                  To " ?opp2 " " ?n2 " request to " ?op2 " " ?npd " to " ...
                        ?tar " " ?nidp ".
143      "))
144      (bind ?tm (str-cat ?tm "{" ?g ",oposite," ?pd "," ?idp "};
145      "))
146      (modify ?conf (toprint ?tp) (toMATLAB ?tm) (idn (+ 1 ?k)))
147      (retract ?io1)
148      (retract ?io2)
149      )
150
151  (defrule add-conflict-2  "Avoid clausule"
152      (phase conflict)
153      ?a <- (avoid ?c ?t ?p ?val)
154      ?conf <- (conflicts (toprint ?tp) (toMATLAB ?tm) (idn ?k))
155      (or (parameter (id ?c) (name ?n1))
156          (parameter-instance (id ?c) (name ?n1))
157          (parameter-instantiated (id ?c) (name ?n1)))
158      (or (parameter (id ?p) (name ?n2))
159          (parameter-instance (id ?p) (name ?n2))
160          (parameter-instantiated (id ?p) (name ?n2)))
161      (goal (criterion ?t) (parameter ?p) (priority ?g))
162      =>
163      (bind ?tp (str-cat ?tp ?k ": when the parameter " ?n1 " changes, ...
            the target " ?t " " ?n2 " is unreached.
164      "))
165      (bind ?tm (str-cat ?tm "{" ?g ",avoid," ?c "," ?val "};
166      "))
167      (modify ?conf (toprint ?tp) (toMATLAB ?tm) (idn (+ 1 ?k)))
168      (retract ?a)
169  )
170
171  ;end phase of conflics. modify.
172  (defrule end-conflicts "End phase of conflicts"
173      ?c <- (phase conflict)
174      (not (and (internal-op (parameter ?p1) (request ?op) (dependent ...
            ?pd) (to ?tar) (target ?idp))
175               (internal-op (parameter ?p2) (request ?op2) (dependent ?pd) ...
                     (to ?tar) (target ?idp))
176               (oposite ?op ?op2)))
177      (not (avoid))
```

```
178      ?con <- (conflicts (toMATLAB ?tm))
179      =>
180      (retract ?c)
181      (bind ?tm (str-cat ?tm "}"))
182      (modify ?con (toMATLAB ?tm))
183      (assert (phase select-goal)))
184
185  ;::::::: Selection of next goal to be processed:
186  ;1.- consider the next goal to be processed
187  (defrule r11-nextgoal "Select next goal to be processed"
188      ?ph<-(phase select-goal)
189      ?cur <- (current null)
190      (goal (priority ?p))
191      ?nx <- (next ?p)
192      =>
193      (retract ?ph)
194      (retract ?cur)
195      (retract ?nx)
196      (assert (current ?p) (next (+ ?p 1)) (phase init-recommendation)))
197
198  ;2.-if there are not more goals, we have been finishied
199  ;we have finished when (next ?g)
200  ;and there are not goals with id==?g.
201  (defrule r12-goToPrint "There are not goals to be processed."
202      ?ph <- (phase select-goal)
203      ?cur <- (current null)
204      ?n <- (next ?g)
205      (not (exists (goal (priority ?g))))
206      =>
207      (retract ?ph)
208      (retract ?cur)
209      (retract ?n)
210      (assert (phase print) (print 1))
211      (printout t crlf " " crlf))
212
213  ;=========================================================
214  ;2.- Rules that can be done in second step:
215  ;GENERATE ONE RECOMMENDATION SET FOR GOAL
216  ;=========================================================
217  ;2.1.-Initialize the recommendations if there is a goal without
218  ;recommendation and there are individual recommendations (op)
219  ;that should be processed:
220  ;NOTE: one rule per type of parameter: normal, instantiated, instance.
221  (defrule r21-initrecomm-A "Init recommendation for a goal max/min ...
       normal parameter"
222      ?ph <- (phase init-recommendation)
223      (current ?p)
224      ?g <- (goal (criterion ?c) (parameter ?idp) (priority ?p))
225      (parameter (id ?idp) (name ?n))
226      (not (exists (recommendation-set (goal ?p))))
227      =>
228      (bind ?tp (str-cat "
229      To " ?c " " ?n ":
230      "))
```

```
231      (bind ?mc (str-cat "g" ?p "={" ?p "," ?idp "," ?c ",{...
232      "))
233      (assert (recommendation-set (goal ?p) (toprint ?tp) (toMATLAB ?mc)))
234      (assert (phase built-recommendation))
235      (retract ?ph)) ;we whant that the goal be actived again
236
237  ;Special case, if is a parameter instantiated, then the satisfaction of
238  ;the criteria is achieved through its instances
239  (defrule r21-initrecomm-B "Init recommendation for a goal max/min ...
        parameter instantiated"
240      (current ?p)
241      ?ph <- (phase init-recommendation)
242      ?g <- (goal (criterion ?c) (parameter ?idp) (priority ?p))
243      (parameter-instantiated (id ?idp) (name ?n))
244      (not (exists (recommendation-set (goal ?p))))
245      =>
246      (bind ?tp (str-cat "
247      To " ?c " " ?n ", the mechanisms (instances) should be " ?c"d:
248      "))
249      (bind ?mc (str-cat "g" ?p "={" ?p "," ?idp "," ?c ",{...
250      "))
251      (assert (recommendation-set (goal ?p) (toprint ?tp) (toMATLAB ?mc)))
252      (assert (phase built-recommendation))
253      (retract ?ph)) ;we whant that the goal be actived again
254
255  (defrule r23-initrecomm-C "Init recommendation for a goal max/min ...
        parameter instance"
256      (current ?p)
257      ?ph <- (phase init-recommendation)
258      ?g <- (goal (criterion ?c) (parameter ?idp) (priority ?p))
259      (parameter-instance (id ?idp) (name ?n))
260      (not (exists (recommendation-set (goal ?p))))
261      =>
262      (bind ?tp (str-cat "
263      To " ?c " " ?n ":
264      "))
265      (bind ?mc (str-cat "g" ?p "={" ?p "," ?idp "," ?c ",{...
266      {"))
267      (assert (recommendation-set (goal ?p) (toprint ?tp) (toMATLAB ?mc)))
268      (assert (phase built-recommendation))
269      (retract ?ph)) ;we whant that the goal be actived again
270
271  ;=========================================================
272  ;3.- Rules that can be done in third step:
273  ;COMPLETE THE RECOMMENDATION SET PER GOAL
274  ;=========================================================
275  ;=========
276  ;3.1.-Normal parameters in antecedent, provide recommendation.
277  ;(?p2 non instantiated and non instance)
278  ;=========
279  (defrule r31-Builtrecomm-A "operate normal parameter to max/min normal ...
        parameter"
280      (current ?p)
281      (phase built-recommendation)
```

```
282      (goal (criterion ?c) (parameter ?idp) (priority ?p))
283      ?op <- (op (todo ?f) (on ?p2) (to ?c) (p ?idp) (val ?val))
284      (not (exists (op (todo ?fun2) (on ?par2) (to ?c) (p ?idp) (val ...
             ?x&:(> ?x ?val)))))
285      (parameter (id ?p2) (name ?nameX))
286      (parameter (id ?idp))
287      ?rs <- (recommendation-set (goal ?p) (toprint ?tp) (toMATLAB ?tm))
288      =>
289      (bind ?tp (str-cat ?tp "-> " ?f " " ?nameX ", effect:" ?val ";
290      "))
291      (bind ?tm (str-cat ?tm "{" ?f "," ?p2 "," ?p "};...
292      "))
293      (retract ?op)
294      (modify ?rs (toprint ?tp) (toMATLAB ?tm)))
295
296  (defrule r31-Builtrecomm-B "operate normal parameter to max/min ...
        parameter instantiated"
297      (current ?p)
298      (phase built-recommendation)
299      (goal (criterion ?c) (parameter ?idp) (priority ?p))
300      (parameter-instantiated (id ?idp) (name ?idpname))
301      ?rs <- (recommendation-set (goal ?p) (toprint ?tp) (toMATLAB ?tm))
302      (subgoal (priority ?p) (parameter ?pinstance))
303      (parameter-instance (id ?pinstance) (id-parent ?idp) (name ?nisn))
304      ?op <- (op (todo ?f) (on ?p2) (to ?c) (p ?pinstance) (val ?val))
305      (parameter (id ?p2) (name ?p2name))
306  ;    (not (and (exists (op (todo ?fk) (on ?p2) (to ?c) (p ?pinst2) (val ...
        ?x&:(> ?x ?val))))
307  ;              (parameter-instance (id ?pinst2) (id-parent ?idp))))
308      =>
309      (retract ?op)
310      (bind ?tp (str-cat ?tp "->" ?f " " ?p2name " to " ?c " " ?idpname " ...
             throught the instance " ?nisn ", effect:" ?val ";
311      "))
312      (bind ?tm (str-cat ?tm "{" ?f "," ?p2 "," ?val "};...
313      "))
314      (modify ?rs (toprint ?tp) (toMATLAB ?tm)))
315
316  ;instance in the antecedent
317  (defrule r31-Builtrecomm-C  "operate normal parameter to max/min ...
        parameter instance"
318      (current ?p)
319      (phase built-recommendation)
320      (goal (criterion ?c) (parameter ?idp) (priority ?p))
321      ?op <-  (op (todo ?f) (on ?p2) (to ?c) (p ?idp) (val ?val))
322      (not (exists (op (todo ?fun2) (on ?par2) (to ?c) (p ?idp) (val ...
             ?x&:(> ?x ?val)))))
323      (parameter (id ?p2) (name ?nameX))
324      (parameter-instance (id ?idp))
325      ?rs <- (recommendation-set (goal ?p) (toprint ?tp) (toMATLAB ?tm))
326      =>
327      (bind ?tp (str-cat ?tp "-> " ?f " " ?nameX ", effect:" ?val ";
328      "))
329      (bind ?tm (str-cat ?tm "{" ?f "," ?p2 "," ?p "};...
```

```
330      "))
331      (retract ?op)
332      (modify ?rs (toprint ?tp) (toMATLAB ?tm))
333      )
334
335  ;=========================================================
336  ;3.1.1- Rules in the third step - construction of subgoals
337  ;COMPLETE THE RECOMMENDATION SET PER SUBGOAL AND GET THE
338  ;BEST RECOMMENDATION.
339  ;=========================================================
340  ;search the best recommendation for subgoals. if the parameter to
341  ;achieve the subgoal is instantiated, then, for sure exists
342  ;an instance that satisfies the goal.
343  ;=========
344  ;3.2.-Instantiated parameters in the antecedent
345  ;(?p2 instantiated) --> exist ?pk tq (?pk instance).
346  ;=========
347  ;In case of instances, it is required a more specific
348  ;recommendation. This can be done by sorting the parameters
349  ;regarding the operations that are prefarable, based on the
350  ;value provided in the goal.
351  ;In this case, we can use this property of the model:
352  ;PROPERTY OF THE MODEL:
353  ;If there is a recommendation for a instantiated parameter,
354  ;then, there is a recommendation for each instance of said
355  ;parameter. This is due to that:
356  ;If x belongs to Ip, then for all y | x \in P(y), that is, x is parent
357  ;of y, then, y belongs to Ip too.
358  (defrule r32-Builtrecomm-A "operate parameter instantiated to max/min ...
       normal parameter"
359      (current ?p)
360      (phase built-recommendation)
361      ?g <- (goal (criterion ?c) (parameter ?idp) (priority ?p))
362      ?op <- (op (todo ?f) (on ?p2) (to ?c) (p ?idp) (val ?val))
363      ?op2 <- (op (todo ?f3) (on ?p3) (to ?c) (p ?idp) (val ?val3))
364      (parameter-instantiated (id ?p2))
365      (parameter (id ?idp))
366      (parameter-instance (id ?p3) (name ?nameX) (id-parent ?p2))
367      (not (exists (and (parameter-instance (id ?p4) (id-parent ?p2))
368                        (op (on ?p4) (to ?c) (p ?idp) (val ?val4&:(> ?val4 ...
                             ?val3))))))
369      ?rs <- (recommendation-set (goal ?p) (toprint ?tp) (toMATLAB ?tm))
370      =>
371      (retract ?op2)
372      (retract ?op)
373      (bind ?tp (str-cat ?tp "-> " ?f3 " " ?nameX ", effect:" ?val3 ";
374      "))
375      (bind ?tm (str-cat ?tm "{" ?f3 "," ?p3 "," ?p "};...
376      "))
377      (modify ?rs (toprint ?tp) (toMATLAB ?tm)))
378
379  ;when the objective is a parameter instantiated (idp)
380  ;==== was adapted (ass r31-builtrecomm-B)
381  ;if the objective is a parameter instantiated, then the problem ...
```

```
      consists on identify
;the instance that best satisfies the objective criterion.
(defrule r32-Builtrecomm-B "operate parameter instantiated to max/min ...
   parameter instantiated"
    (current ?p)
    (phase built-recommendation)
    (goal (criterion ?c) (parameter ?idp) (priority ?p))
    (parameter-instantiated (id ?idp) (name ?idpname))
    ?rs <- (recommendation-set (goal ?p) (toprint ?tp) (toMATLAB ?tm))
    (op (todo ?f) (on ?p2) (to ?c) (p ?pinstance) (val ?val))
    ?op <- (op (todo ?f2) (on ?p21) (to ?c) (p ?pinstance) (val ?val2))
    (parameter-instance (id ?pinstance) (id-parent ?idp) (name ?pinsn))
    (parameter-instance (id ?p21) (id-parent ?p2) (name ?p21name))
;   (not (and (exists (op (todo ?fk) (on ?pk) (to ?c) (p ?pinst2) (val ...
   ?valk&:(> ?valk ?val2))))
;            (exists (parameter-instance (id ?pk) (id-parent ?p2)))
;            (exists (parameter-instance (id ?pinst2) (id-parent ?idp)))))
    =>
    (retract ?op)
    (bind ?tp (str-cat ?tp "->" ?f2 " " ?p21name " to " ?c " " ?idpname ...
       " throught " ?c " " ?pinsn ", effect:" ?val2 ";
    "))
    (bind ?tm (str-cat ?tm "{" ?f2 "," ?p21 "," ?val2 "};...
    "))
    (modify ?rs (toprint ?tp) (toMATLAB ?tm)))

;when the objective is a parameter instance (idp)
(defrule r32-Builtrecomm-C "operate parameter instantiated to max/min ...
   parameter instance"
    (current ?p)
    (phase built-recommendation)
    ?g <- (goal (criterion ?c) (parameter ?idp) (priority ?p))
    ?op <- (op (todo ?f) (on ?p2) (to ?c) (p ?idp) (val ?val))
    ?op2 <- (op (todo ?f3) (on ?p3) (to ?c) (p ?idp) (val ?val3))
    (parameter-instantiated (id ?p2))
    (parameter-instance (id ?idp))
    (parameter-instance (id ?p3) (name ?nameX) (id-parent ?p2))
    ?rs <- (recommendation-set (goal ?p) (toprint ?tp) (toMATLAB ?tm))
    =>
    (bind ?tp (str-cat ?tp "-> " ?f3 " " ?nameX ", effect:" ?val3 ";
    "))
    (bind ?tm (str-cat ?tm "{" ?f3 "," ?p3 "," ?p "};...
    "))
    (retract ?op2)
    (modify ?rs (toprint ?tp) (toMATLAB ?tm)))

;=========================================================
;4.- Check if we have finish with this goal, and CLEAN
;When finish, assert (current null) to go back to 1.
;Or (phase print) (print 1) to jump to the last phase and
;print the results, starting for the first goal.
;=========================================================
;we have finished a goal when there are not more options
;(individual recomm.) to be applied for the goal.
```

```
(defrule r41-endGoal
    ?ph <- (phase built-recommendation)
    (goal (priority ?g) (criterion ?c) (parameter ?idp))
    ?rs <- (recommendation-set (goal ?g) (toprint ?tp) (toMATLAB ?tm))
    ?cg <- (current ?g)
    (or (not (exists (op (to ?c) (p ?idp))))
        (forall (op (todo ?f) (on ?idp))
            (parameter-instantiated (id ?idp))))
    (conflicts (toMATLAB ?tmc))
    =>
    (retract ?ph)
    (retract ?cg)
    (bind ?tm (str-cat ?tm "}," ?tmc  "};
"))
    (modify ?rs (toMATLAB ?tm))
    (assert (current null))
    (assert (phase select-goal))
    (assert (next (+ 1 ?g))))

;=============================================================
;5.- PRINT RESULTS
;(last phase)
;=============================================================
(defrule r5-print-recomm "Print recommendation set for goals"
    (phase print)
    ?pr <- (print ?g)
    (recommendation-set (goal ?g) (toprint ?s) (toMATLAB ?tm))
    =>
    (printout t ?s crlf)
    (printout t ?tm crlf)
    (retract ?pr)
    (assert (print (+ ?g 1))))

;Conflics and requirements may be printed independently
(defrule    r5-print-conf "Print conflicts for goals"
    ?pp <- (phase print)
    (print ?g)
    (not (exists (recommendation-set (goal ?g2&:(≥ ?g2 ?g)))))
    (conflicts (toprint ?tp) (toMATLAB ?tm))
    =>
    (printout t ?tp)
    (printout t ?tm)
    (retract ?pp)
    (assert (phase clean)))

(defrule r6-clean
    (phase clean)
    ?tp <- (print ?p)
    =>
    (retract ?tp))
;=============================================================
;6.- DEFINITION OF FACTS (THIS IS MADE DYNAMICALLY BY JESS)
;=============================================================
```

APPENDIX C

Summary in Spanish / Resumen en español

C.1. Introducción y motivación

La Internet del Futuro (IF) plantea la interconexión de redes heterogéneas, compuestas por diversos tipos de dispositivos que pueden cooperar y aprovechar las mejoras fruto de la convergencia, como por ejemplo disponer de mayores rangos de conectividad y aplicar mecanismos de seguridad nativos en determinadas redes. La participación del usuario en las redes del futuro incentiva la convergencia entre diferentes tecnologías, a fin de aumentar aún más la funcionalidad ofrecida por los dispositivos de usuario.

De hecho, los mecanismos de seguridad son esenciales y de obligada aplicación en redes donde los dispositivos personales tienen cabida. Sin embargo, su adecuación a los escenarios de la IF tiene un coste en términos de calidad de servicio (QoS) que es necesario evaluar previo a su despliegue, más si cabe teniendo en cuenta que no todos los dispositivos desplegados en dichas redes cuentan con suficientes recursos para implementar o resistir el efecto de cualquier medida de seguridad. El problema se agrava si tenemos en cuenta que los propios mecanismos de QoS requieren recursos para funcionar, con el fin de garantizar en la medida de lo posible los diferentes tipos de tráfico que generan las aplicaciones y servicios requeridos por el usuario o los nodos que ofrecen sus servicios a la red.

Algunos enfoques para la configuración de mecanismos en base a criterios de seguridad y QoS se encuentran en la configuración de servicios, ya sea por medio del uso de ontologías [2], o bien en base a criterios de seguridad dependientes [132]. Aunque estos sistemas permiten un nivel de abstracción adecuado para el análsis de servicios, no permiten agregar componentes u *objetos* a distinto nivel, por ejemplo, la existencia de elementos anti-tampering en dispositivos de la red.

Por lo tanto, proponemos un enfoque paramétrico basado en la composición de *elementos* u *objetos*, dirigido a la evaluación de la composición de soluciones de seguridad y QoS, en base

a la descripción de los entornos en base a sus parámetros y dependencias. Con este fin, un estudio detallado de características de las redes candidatas a formar parte de la IF es preciso, para identificar parámetros de seguridad y QoS clave, así como otros parámetros intermedios por los cuales ambos tipos de mecanismos se ven relacionados. Esta información debe ser analizada y presentada a un usuario final de forma adecuada para su entendimiento, así como ser fácilmente extensible o modificable por cualquier usuario para su posterior adaptación a estudios futuros.

C.1.1. Redes en convergencia

De las redes candidatas a formar parte de la Internet del Futuro, las redes restringidas en recursos tienen un papel fundamental. De hecho, en este trabajo consideramos tres tipos de redes que pensamos cubren un espectro adecuado para la extracción de parámetros y posterior análisis. Estas redes son: las redes inalámbricas de sensores, las redes móviles ad-hoc y las redes celulares. Mientras que las redes de sensores permiten un acercamiento físico al entorno, las redes ad-hoc permiten ampliar el alcance de los dispositivos y las redes celulares aportan, por sus características, opciones de conectividad y seguridad mayores. A raíz del estudio de estas tres redes, la conclusión a la que llegamos es que jugarán un papel fundamental en las redes del futuro, y nuestro conjunto de parámetros comunes, del que hablaremos posteriormente, es extraído por tanto del análisis focalizado en estas redes.

C.1.1.1. Redes de Sensores

Una red inalámbrica de sensores, conocida por sus siglas en inglés *Wireless Sensor Network* (WSN), es una red inalámbrica compuesta de dispositivos capaces de tomar mediciones de su entorno, por lo general autónomos, desarrollados para resolver un problema específico, con limitada capacidad computacional (la necesaria para resolver el problema) y restringidos en recursos (ej. batería limitada). Las WSNs son usadas para monitorizar condiciones físicas o del entorno dentro de un área. Existe una amplia gama de sensores, capaces de medir la luminosidad, temperatura, aceleración y magnetismo entre otros. Por lo tanto, son muy usados en sistemas de monitorización, o localización. Los sensores para temperatura, humedad y radiación suelen ser usados en infraestructuras críticas, con lo que la protección de éstos cuando se encuentran expuestos en áreas de uso público es fundamental para evitar su uso mal intencionado.

Precisamente, la popularidad de los sensores se debe a su amplia variedad, su autonomía, su capacidad para soportar duras condiciones medioambientales y la facilidad de sustituir unas unidades por otras que realicen la misma función (ej. en caso de fallo). Su mayor limitación en cuanto a su integración en la Internet del Futuro se debe a sus restricciones físicas y funcionales. Sin embargo, su integración traería consigo numerosas oportunidades de negocio y funcionales, dado que los sensores son los elementos físicos más cercanos al mundo real.

La integración de las WSNs en la IF pesenta diversos desafíos, entre los que citamos algunos a continuación.

C.1.1.1.1. Despliegue de mecanismos de Seguridad. La dificultad en dotar estas redes con mecanismos de seguridad se encuentra reflejada en varios trabajos [15, 37]. Aunque soluciones centalizadas pudieran darse, en las que los nodos con más energía monitoizasen la red para detectar amenazas, el flujo de tráfico adicional acaba consumiendo el recurso más preciado de la red, que es la energía, y los propios nodos. Más aún, las soluciones que no contemplan la auto-defensa de un nodo, dejan el nodo expuesto al atacante con acceso físico o dentro de la cobertura de los nodos. Técnicas para incrementar la seguridad a nivel local de los nodos pueden basarse en emplear paquetes falsos para evitar que el atacante conozca la posición de los nodos, pero estas técnicas también acaban empleando recursos de red y energéticos más allá de lo que algunos nodos pueden tolerar [38, 39].

C.1.1.1.2. Seguridad como factor clave para el rendimiento. Pese al efecto negativo de los mecanismos de seguridad en los recursos, la carencia de éstos puede traer consecuencias muy nefastas. Por ejemplo, en [18] se analiza el efecto de no proporcionar confidencialidad, integridad, autenticidad y disponiblilidad en una red de sensores. El estudio abarca diversas tecnologías: WISA, WirelessHart, ISA 100.11a, ZigBee y 802.15.4 MAC. Los resultados muestran cómo la pérdida de paquetes y el empeoramiento del *throughput* se ven incrementados en escenarios de WSN donde se dan las carencias de las propiedades de seguridad citadas. Sin embargo, no ponen de manifiesto el coste en recursos que puede causar proporcionar los mecanismos que implementen dichas propiedades.

C.1.1.1.3. Redundancia de datos y jerarquía. La relación entre redundancia de datos, fiabilidad, consumo energético, fusión de datos, y retrasos puede consultarse en [17]. A mayor redundancia, la información resulta más fiable, aunque el inconveniente que presenta es el incremento de tráfico en la red, lo que conlleva a retrasos y pérdida de paquetes. La agregación en una jerarquía de nodos permite resumir la información y a la vez tomar diferentes mediciones de los nodos agrupados en clusters. El inconveniente de los clusters es que suponen puntos comunes de recogida de información y, aunque estos nodos dispongan de más recursos, los nodos vecinos ven reducida su energía por emplearla en labores de routing. Además, si un atacante logra suplantar un cluster (o engañar a éste), puede tomar gran relevancia en el comportamiento del sistema, que actúa según los datos recogidos de la red.

C.1.1.1.4. Despliegue de mecanismos de QoS. Los mecnanismos de QoS para una red de sensores tienen que contemplar el consumo energético. Sin embargo, esto no siempre es posible. Los mecanismsos de QoS, tienen que garantizar ciertas propiedades a las aplicaciones que requieren de éstos, y originalmente no están ideados para tener en cuenta otras prioridades de nivel físico, como la energía en los nodos. La reserva de recursos, en general, requiere tráfico adicional para reservar slots de tiempo o espacio en los buffers de comunicaciones. Por sí misma, una mala gestión de QoS en una WSN puede ser nefasta. Pero, además, los ataques a estos tipos de mecanismos de reserva de recursos pueden ser muy efectivos para desencadenar un ataque de denegación de servicio, por ejemplo, cuando peticiones de reserva son distribuidas pero nunca usadas, o de aislamiento de sectores de la red [48]. Los propios protocolos de reserva pueden liberar los recursos transcurido un tiempo, pero aún así, este hecho consume energía.

C.1.1.2. Redes Ad-Hoc Móviles

En general, una red ad-hoc móvil, conocida por sus siglas en inglés, *Mobile Ad-Hoc Network* (MANET), está, al igual que una red de sensores, compuesta por dispositivos (o nodos) conectados inalámbricamente, que tienen capacidades auto-configurables y auto-organizativas. El uso que se le da a cada tipo de red es lo que acentúa las diferencias entre ambas. Así, mientras que las redes de sensores emplean dispositivos que captan información de su entorno, una red ad-hoc está compuesta de dispositivos que pueden comunicarse entre sí, pero no por ello tienen que tener, necesariamente, mecanismos para obtener información física del entorno.

Además, las MANETs son redes más cercanas al usuario, utilizadas principalmente para efecturar labores de comunicación con aplicaciones que pueden resultar mucho más dinámicas y menos específicas que las llevadas a cabo sobre una WSN. Las WSN pueden configurarse como MANETs, si además de captar información son móviles y se comunican en uno o varios saltos de forma dinámica. Pero un dispositivo sin sensores y con capacidades ad-hoc no podría actuar como sensor.

A nivel de protocolos de comunicaciones, un sensor se rige por una arquitectura cross-layer para optimizar los recursos disponibles, mientras que un dispositivo MANET tiene la misma pila que la empleada en las redes TCP/IP. Esto debe ser así, precisamente porque los dispositivos MANET no están tan restringidos en uso como lo pudiera estar un sensor, que por norma general tiene una función específica dentro de una red. La arquitectura cross-layer permite que una funcionalidad común a varias capas pueda ser accedida directamente. Este planteamiento, que permite optimizar el acceso a tales funciones generales, puede volverse muy dificil de gestionar conforme aumenta la complejidad del nodo y la incertidumbre sobre el uso del dispositivo.

La integración de las MANETs en la IF pesenta diversos desafíos, entre los que citamos algunos a continuación.

C.1.1.2.1. Despliegue de mecanismos de QoS y Seguridad. Se espera que las redes MANET sean algo menos restringidas en recursos que una WSN, pero aún así el despliegue de los mecanismos de seguridad y QoS no es trivial. Aún así, existen diversas iniciativas al respecto, como los protocolos INSIGNIA y SWAN, basados en IntServ y DiffServ [50, 51], o el protocolo DRQoS, para intentar paliar los ataques contra la reserva de recursos [53]. Además, el uso de IPsec en MANET se estudió en [50], pero con el consiguiente uso de claves pre-compartidas y certificados. No todos los dispositivos pueden adecuarse a este modelo, y el coste en redes móviles puede resultar demasiado costoso en términos energéticos. Limita, además, la heterogeneidad de dispositivos que puedan coexistir en la MANET.

C.1.1.2.2. Naturaleza dinámica y auto-organizativa. Las MANETs permiten mayor flexibilidad por tanto en la definición de tales mecanismsos de seguridad y QoS, pero se exponen a un problema fundamental, y es que la movilidad permite retener menos información de la conducta impropia de los dispositivos, si éstos no se identifican en la red. Por tanto, un dispositivo móvil usado para ataques, puede desaparecer de la red por un tiempo y, posteriormente, volver a ésta y continuar con su ataque. La red puede mantener información sobre los dispositivos, pero el coste de mantener dicha información en numerosos entornos

es inviable. La captación de datos, su almacenamiento y posterior uso en tiempo real sin afectar a las comunicaciones plantea grandes desafíos.

C.1.1.3. Redes Celulares

La adaptación de las redes celulares al paso del tiempo y las crecientes demandas de los usuarios supone por sí mismo un gran desafío. A diferencia de las redes anteriores, estas redes se construyen sobre dos pilares: el despliegue de una infraestructura de comunicaciones cableada y su conexión con la infraestructura inalámbrica que se compone de estaciones base, menos restringidas en recursos, y dispositivos móviles de usuario, los cuales no llegan a ser independientes de la infraestructura y están sometidos a cierto control de seguridad.

A diferencia de las redes anteriores, una red celular sólo puede ser desplegada por grandes compañías que se encarguen del mantenimiento de las líneas, bases de datos y las antenas, así como del resto de elementos necesarios para la comunicación, que componen la infraestructura de red. Dada su complejidad, nos centraremos en la parte que permite la convergencia con el resto de redes y dispositivos.

Destacamos los siguientes desafíos, por sus diferencias con los casos anteriores.

C.1.1.3.1. Consideraciones de Negocio. Prácticamente la totalidad de los trabajos basados en las arquitecturas de cuarta generación (4G) de dispositivos móviles destacan el enfoque All-IP sobre el que están siendo ideadas, así como sus problemas de seguridad y establecimiento de garantías de QoS. Son redes muy ligadas al aprovisionamiento de servicio y a responder a las demandas del usuario, por lo que ambos aspectos de seguridad y QoS son primordiales. Por ejemplo, los ataques efectivos sobre estas infraestructuras pueden estar ligados a la denegación de servicio o a empeorar el rendimiento de la red [58]. Las técnicas criptográficas podrían ser aplicables en los dispositivos móviles de usuario, pero a coste de incrementar su precio y complejidad [59].

C.1.1.3.2. Alternativas a la Criptografía en escenarios 5G. Los escenarios de quinta generación (5G) tienen como objetivo primordial la mejora de la red de comunicaciones para proveer de mayor cobertura y ancho de banda a los usuarios. La amplia demanda de servicios por parte de éstos y el desafío añadido de la movilidad y el uso de dispositivos personales, estrechamente vinculados al usuario, hace que se planteen mecanismos de seguridad que puedan ser desplegados aprovechando las características ad-hoc de los dispositivos móviles con los relays o dispositivos intermedios entre el usuario y la estación base. Así, la seguridad a nivel físico plantea evitar ataques poco sofisticados por medio de la elección de los parámetros adecuados para permitir la comunicación entre los nodos legítimos evitando los nodos no autorizados. Estos mecanismos, aunque no evitan ataques complejos, y presuponen un conocimiento global de la red, permiten solventar algunos casos de amenazas sin usar primitivas criptográficas que no todos los nodos pueden emplear. Además, la elección de dichos parámetros es nativa en estos dispositivos, por lo que no supone un coste adicional superior, más allá del almacenamiento de la información global de red empleada para el cálculo.

C.1.1.3.3. Perspectiva 5G Green. Los nuevos dispositivos y mecanismos de comunicación en redes 5G se plantean como dispositivos *Green*. Esto quiere decir, que los materiales con los que serán fabricados se espera que puedan ser en su mayor parte reciclables, y que también se espera disminuir el consumo enerjético, todo ello dirigido a disminuir la huella de carbono por el impacto tecnológico. Esto supone un reto, ya que el consumo energético es un factor clave para posibilitar la implantación de numerosos mecanismos de seguridad que se basan en la comunicación entre los dispositivos. Además, la mejora de los dispositivos por medio de elementos hardware queda muy limitada, ya no sólo por la inversión económica por incrementar el coste del dispositivo, sino también por su vertiente no-green, o por el coste adicional (sobre el coste normal) que conlleva el proporcionar la vertiente Green de diversas tecnologías.

C.1.1.4. Colaboración

Para la convergencia entre las diferentes redes deben producirse cambios que permitan su adecuación a protocolos ampliamente empleados por redes tradicionales, como por ejemplo los esfuerzos para adaptar el uso de IP por parte de dispositivos restringidos en recursos [73], o bien migrar hacia nuevas formas de comunicación más eficientes, dirigida a conectar los dispositivos considerando los últimos requisitos de comunicación para las redes del futuro [88].

C.1.1.4.1. Mobilidad. Como parte fundamental en las redes de futuro se encuentran las adaptaciones para la movilidad de los dispositivos. Las soluciones que consideran múltiples dominios donde los usuarios son autenticados cuentan con el desafío permanente de la frescura de los datos y los tiempos de retardo. Además, la transparencia de cara al usuario es vital, por lo que el servicio no puede ser interrumpido [8]. Estos desafíos están presentes en redes con soporte de *Mobilidad IP* (MIP), donde además el tráfico es redirigido entre los diferentes dominios que gestionan la movilidad del dispositivo. Los servicios de movilidad independientes del medio (MIH) consideran además la migración entre diferentes protocolos de comunicación (ej. de WiMAX a 3G WWAN) sin pérdida de conexión para el usuario [75].

C.1.1.4.2. Protocolo de Seguridad para IP. El protocolo de seguridad para IP presenta varios inconvenientes para su uso en redes heterogéneas. En particular, el establecer las asociacione se seguridad que el protocolo necesita puede llegar a ser muy restrictivo y limitar la conectividad del dispositivo. El uso de IPsec no garantiza que el dispositivo sea confiable, a no ser que se establezcan opciones más restrictivas en IPsec. El uso de una comunicación así, dependería del número de dispositivos que decidan que quieren usar este modelo de comunicación. Siendo una solución poco flexible que consume picos de energía en el establecimiento de la comunicación, esto en una red móvil puede ser muy costoso energéticamente, ya que los nodos tienen que perder y recuperar la conexión un número ilimitado de veces, y sin que se perciba un retardo por ello.

C.1.1.4.3. Dispositivos restringidos en recursos. Sin embargo, no todos los dispositivos disponen de los recursos computacionales y energéticos para implementar mecanismos

de movilidad basados en IP. De hecho, el soporte de IP no es trivial, y el uso de protocolos traductores es necesario. Por ejemplo, 6lowPAN [88] permite la comunicación con dispositivos restringidos en recursos como sensores, proveniente de redes IP. Para ello, 6lowPAN encapsula las tramas IPv6 en paquetes de datos que pueden manejar los sensores. El coste de esta adaptación es la sobrecarga adicional por la fragmentación de los datos, y la compresión de las cabeceras de los paquetes IPv6.

C.1.1.4.4. Conexión con Internet. Alternativas a 6LoWPAN son ZigBee y el protocolo Máquina-a-Máquina (M2M). Mientras que ZigBee es dirigido a la comunicación entre sensores y no está ideado para su conexión con Internet, M2M es más genérico y considera dispositivos heterogéneos, pero esta heterogeneidad hace que el papel de conectar los dispositivos con Internet recaiga en redes de soporte (backbone) que cuentan con equipos más potentes. Sin embargo, la ventaja de 6LoWPAN es que la adpatación a IPv6 es nativa. La conexión con Internet es un punto clave en el desarrollo de la IF, ya que abre la puerta a la conectividad global, en cualquier momento y lugar.

C.1.1.4.5. Redes Definidas por Software y Virtualización. Las redes definidas por software, o SDN por sus siglas en inglés *Software Defined Networks*, proponen la defínción de componentes hardware como software, de manera que la gestión de dichos componentes pueda realizarse independientemente de la localización del hardware donde el compoente se ejecuta. El objetivo final es hacer que los componentes software que emulan el hardware sean indistinguibles de un componente hardware real, y que además estos componentes software puedan ser migrados bajo demanda. Se evitan así las limitaciones de espacio físico y costes de despliegue, entre otros. Este concepto es diferente de la virtualización (NFV) de funciones de red, que sí distinguiría los servicios locales de los recursos físicos [4].

C.1.1.4.6. Rendimiento, QoS y Seguridad. Rendimiento, QoS y Seguridad se encuentran intrínsecamente relacionados en los estudios de convergencia de redes. Indudablemente, los mecanismos de seguridad pueden afectar al rendimiento, y también el rendimiento puede empeorar debido a ataques en la red o un mal uso de los recursos. La QoS está ligada a las medidas de rendimiento, ya que una mala gestión de los recursos puede empobrecer mucho la calidad en las comunicaciones, y por tanto los parámetros de red no cumplen los requisitos necesarios para garantizar el buen funcionamiento de los servicios. Sin embargo, los mecanismos de QoS también pueden afectar negativamente a una red que no tiene soporte o recursos suficientes para alojarlos y gestionarlos. El caso más directo es el de las redes de sensores, donde los protocolos de reserva de recursos tradicionales pueden ser muy dañinos y acabar en denegación de servicio, o aislamiento de sectores críticos de red.

C.1.2. Objetivos de la tesis

A pesar de la manifiesta dependencia entre seguridad y QoS en las redes del futuro, los estudios actuales centran sus esfuerzos en pruebas aisladas sobre efectos puntuales en parámetros concretos, y por tanto no permiten un análisis conjunto de dependencias que permita evaluar el impacto combinado de mecanismos a distinto nivel de aplicación. Por ejemplo, la

mayoría de las soluciones dirigidas a la composición de alternativas están destinadas a la composición de servicios web o similares. Pero en la IF la heterogeneidad de dispositivos permite que estos mecanismos se den tanto a nivel físico (ej. mecanismos anti-tampering, tarjetas con identificadores de dispositivo), como a nivel de servicios (mecanismos de autenticación, capacidades multimedia).

Por lo que por nuestra parte sugerimos un análisis paramétrico que permita: 1) descomponer los sistemas en diferentes sub-partes que puedan ser integradas en diferentes ámbitos o contextos. 2) Definir cómo los parámetros son integrados en base al contexto y las propiedades y funciones que podrían aplicarse a tal fin. Este análisis estaría encaminado a cubrir el análisis de dependencias entre aspectos de QoS y Seguridad en las redes del FI, objetivo primordial de la tesis.

C.1.2.1. Contribución Directa

La contribución directa de esta tesis al análisis conjunto de seguridad y QoS puede dividirse en los siguientes subobjetivos alcanzados:

O1. Proporcionamos una visión general de los desafíos principales a ser abarcados en la Internet del Futuro y la Internet de las Cosas de cara a la coexistencia entre mecanismos de seguridad y QoS.

O2. Proporcionamos una clasificación de parámetros de seguridad y QoS relevantes para el análisis de la compensación paramétrica, diferenciando entre comportamiento basado en el contexto general, donde se describen los puntos de relevancia comunes para las diversas redes, y contextos específicos, donde los requisitos específicos relevantes para la supervivencia de cada red son tenidos en cuenta.

O3. Incluimos la percepción subjetiva de parámetros en el contexto, para identificar los parámetros que son prioritarios en base al contexto. Con este fin, identificamos que un enfoque paramétrico es el más adecuado para analizar la compensación de seguridad y QoS en la IF.

O4. Proporcionamos un modelo general para describir escenarios de la IF en base a sus parámetros y relaciones, y proporcionamos un conjunto de parámetros general de seguridad y QoS, ajustables a las necesidades de cualquier usuario que use el modelo.

O5. Proporcionamos los pasos para desplegar una herrmaienta que implemente el modelo, y finalmente implementamos un prototipo de la herramienta, denotado SQT, para gestionar el conjunto de parámetros de seguridad y QoS proporcionados como parte del modelo.

O6. Definimos la integración/extracción de contextos como parte del prototipo, y definimos la arquitectura del modelo para ser fácilmente modificada y mejorada por otros usuarios, según sus necesidades.

O7. Proporcionamos un sistema de recomendaciones para la herramienta que emplea técnicas de inteligencia artificial para aportar indicaciones sobre la información dinámica generada por el modelo, en base a unos criterios objetivo y restricciones seleccionables por el usuario.

O8. Proporcionamos un diseño modular de todo el proceso para permitir la modificación de cualquiera de los ficheros separadamente. Incluso el conjunto de reglas empleadas por el sistema de recomendaciones puede ser modificado para obtener información adicional no considerada en esta tesis.

O9. Validamos la aplicabilidad de las soluciones propuestas en entornos heterogéneos, proporcionando casos de uso relevantes al problema.

C.1.2.2. Distribución de la Tesis

La tesis se divide en seis capítulos, atendiendo a la identificación de los escenarios a evaluar o considerar dentro del FI, el análisis de los parámetros de seguridad y QoS a medir, la definición de un modelo para satisfacer el objetivo de evaluación y asesoramiento de la compensación, la implementación de una herramienta acorde al modelo y, por último, su despliegue para la evaluación en dos casos de uso que contemplen las redes candidatas seleccionadas en el primer paso. La Figura C.1 muestra una visión general de esta estructuración que pasamos a comentar con más detalle a continuación.

Los primeros dos capítulos de la tesis se centran en el análisis exhaustivo de mecanismos de seguridad y QoS en redes, que, por sus características, son candidatas a formar parte del FI (**O1**, **O2**), como lo son las redes de sensores (WSN), redes móviles Ad-Hoc (MANET) y las redes celulares. Con este fin, se estudió la interoperabilidad y dependencias entre parámetros de QoS y Seguridad **[Ci1]**. Como resultado de este estudio se obtuvo una taxonomía que muestra los principales requisitos de interconexión para cada tipo de red, identificándose además un conjunto de dependencias básicas entre parámetros de QoS y seguridad **[R1]**. Los resultados obtenidos ponían de manifiesto: 1) La carencia de mecanismos de balanceo de seguridad y QoS para redes de heterogéneas de composición dinámica. 2) La complejidad en el análisis conjunto de parámetros definidos a distinto nivel de abstracción, y 3) la necesidad de separar la información contextual en dos tipos de contextos: generales o aspectos comunes de las redes y particulares o contextos específicos de cada tecnología o escenario **[A1]**. El hecho de que las soluciones actuales no contemplan estos tres puntos nos conduce a la necesidad de definir de un modelo que permita solventar 1-2 e integre 3.

El análisis de la compensación entre seguridad y QoS se realizó por tanto siguiendo un enfoque paramétrico para dar cabida a los diferentes niveles de abstracción que las diversas soluciones de seguridad y QoS pueden requerir en la FI. Como fruto de este análisis se definió el Modelo de Relaciones Paramétricas (PRM) para la definición de parámetros en base a su nivel, tipo y el conjunto de dependencias que definen el comportamiento de un sistema definido en base al modelo, así como las funciones y las propiedades por las que el modelo se rige (**O4**). El modelo se definió tomando como caso base un escenario de plataformas móviles **[Ci2]**, en el que los parámetros de seguridad y QoS generales, así como

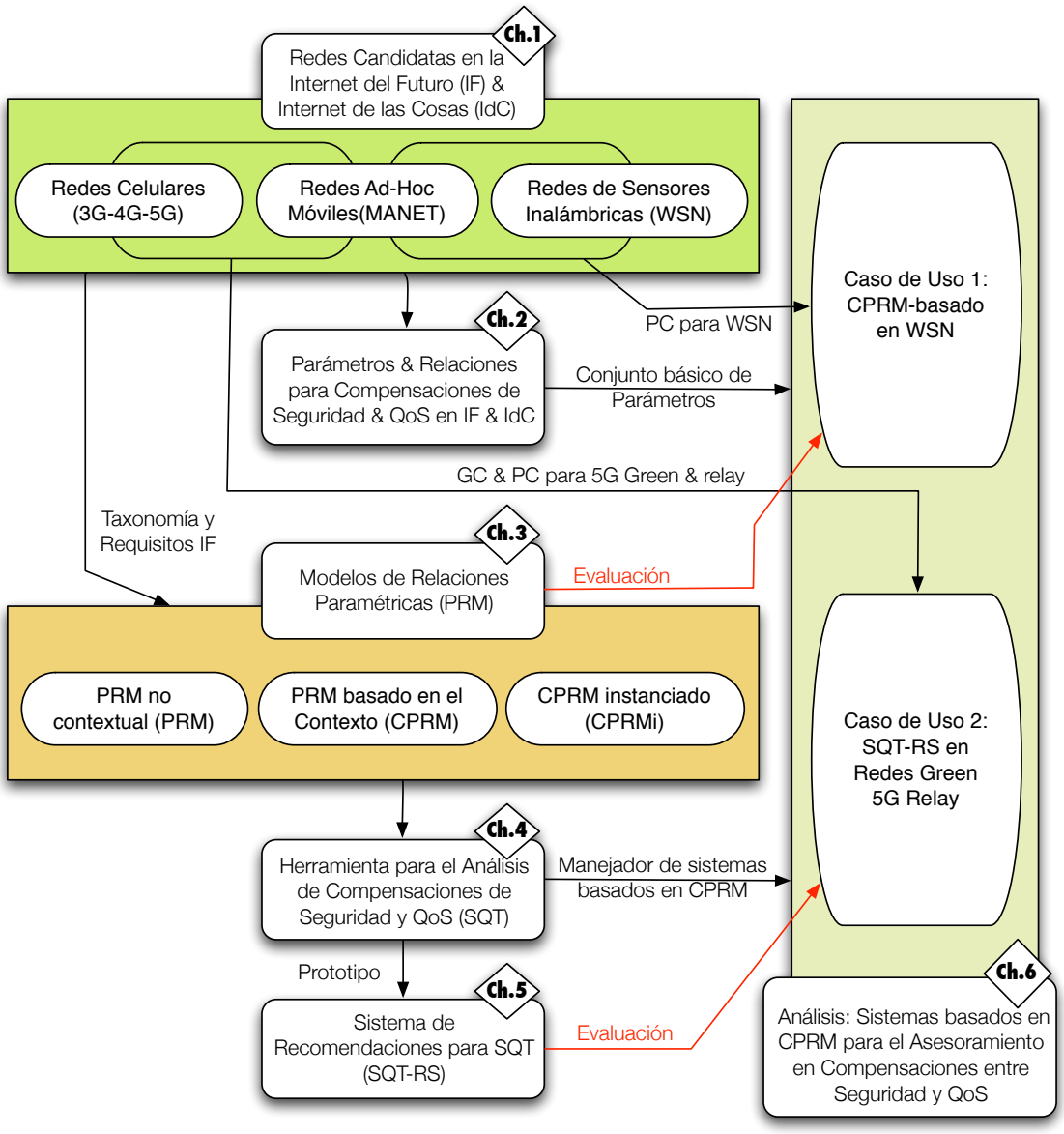


Figura C.1: Mapa de Capítulos.

la estructuración en niveles lógicos y la formulación matemática fueron descritos con detalle **[R2]**. Así, un escenario representado en base al modelo PRM proporciona un script que identifica los componentes en base a la definición del modelo.

La implementación del primer prototipo que demostraba la funcionalidad del modelo se realizó en MATLAB. El modelo PRM, aunque permite definir relaciones complejas entre parámetros, no definía la integración de contextos. Esto significa, que cuando nueva información paramétrica debía ser integrada o retirada del modelo, los scripts debían ser actualizados, y, en consecuencia, sistemas con gran volumen de parámetros generaban scripts muy complejos. Por ello, definimos una mejora sobre el modelo PRM, definiendo el Modelo de Relaciones Paramétricas basado en el Contexto (CPRM) **[Ci3]**, e implementando un manejador de elementos para estos modelos, la herramienta SQT, por sus siglas en inglés Security and QoS

Tradeoff Tool **[Cn1]**. El modelo CPRM define el contexto para los parámetros, niveles, tipos y operaciones definidos en un PRM y adapta los esquemas para la integración dinámica de nuevos parámetros, en base al contexto (**O6**). Además, contempla la subjetividad del valor de los parámetros (**O3**). Es decir, permite diferenciar la relevancia de un parámetro en un momento dado (ej. relevancia de los mecanismos de confianza dado que estamos en un entorno controlado como el hogar).

SQT fue diseñada siguiendo un esquema modular, y su implementación por medio de funciones e interfaces MATLAB hace que sea muy sencillo modificar funcionalidad de la herramienta de forma separada sin afectar a toda la estructura del código (**O5**, **O8**).

La integración dinámica de información o instanciación, tiene un coste computacional que es evaluado en **[Ci4]**, donde se proponen alternativas para mitigar el coste en la integración de información contextual. Sin embargo, al margen de la mitigación, el inconveniente de nuestro modelo es precisamente su mayor virtud: el uso de gran cantidad de información para la toma de decisiones. Mientras más información contiene el modelo, mejor es posible medir el impacto de unos mecanismos de seguridad frente a otros que implementen las mismas propiedades. Sin embargo, como SQT emplea gráficas para la visualización de los resultados, a mayor número de parámetros más compleja se torna el análisis visual de los datos. En especial, el proceso de entrenamiento de usuarios para el uso de SQT se vuelve muy complejo.

Para solventar este hecho, SQT es mejorado con un sistema de recomendaciones (SQT-RS) definido en base a las propiedades de los modelos CPRM **[Ci5]** (**O7**, **O8**). El sistema de recomendaciones toma como entrada información generada dinámicamente, hechos que se calculan en base a la información del modelo (script CPRM) seleccionado y en las propiedades de construcción de un CPRM, y a los objetivos y requisitos seleccionados por el usuario por medio de interfaces diseñadas a tal efecto. Las reglas del sistema de recomendaciones están definidas para CLIPS, implementadas en Jess y el programa que procesa los hechos fue compilado generando el .jar que SQT-RS invoca desde MATLAB para obtener las recomendaciones basadas en los objetivos y los requisitos.

Por último, para el análisis de la solución propuesta definimos dos casos de uso dentro de los escenarios candidatos a formar parte de IF (**O9**). En particular, el primer caso de uso se define sobre WSNs, donde se aplican pesos para cambiar la relevancia de los parámetros y se adaptan los resultados de un trabajo sobre autenticación en WSN al esquema CPRM **[R3]**. Entre los resultados obtenidos, apreciamos la visualización de información adicional a la aportada por el trabajo individual, y cómo el análisis de las propiedades genéricas de seguridad y QoS es posible.

El segundo caso de uso, está dirigido a demostrar la usabilidad de SQT-RS. Con este fin, seleccionamos un escenario con gran número de parámetros, como son las redes celulares de quinta generación 5G, y en particular el marco de trabajo es 5G Green relay. Este escenario cubre los casos de estudio MANET y redes celulares por las características de los dispositivos que componen el entorno. En este segundo caso de uso, nos centramos en las recomendaciones proporcionadas por la herramienta para los distintos contextos, dados en función del tipo de relay/dispositivo y los criterios objetivos. Al basarse en un sistema de expertos, la colaboración con expertos en la materia, llevada a cabo durante el periodo de estancia de tres meses del doctorando en el grupo receptor, ha sido muy provechosa.

Las conclusiones de esta tesis se desglosan en el último capítulo, junto con las observa-

ciones finales y los desafíos abiertos para la compensación entre seguridad y QoS.

C.1.3. Publicaciones y financiación

El contenido de esta tesis ha sido publicado en diversas revistas internacionales de impacto, así como en conferencias internacionales en el área de seguridad y QoS.

Revistas JCR:

R1. A. Nieto, and J. Lopez, *Analysis and Taxonomy of Security/QoS tradeoff solutions for the Future Internet*, In Security and Communication Networks (SCN) Journal, no. 1939-0122, Wiley-Blackwell, pp.2778-2803, 2014.

R2. A. Nieto, and J. Lopez, *A Model for the Analysis of QoS and Security Tradeoff in Mobile Platforms*, In Mobile Networks and Applications (MONET) Journal, vol. 19, issue 1, Springer US, pp. 64-78, 2014.

R3. A. Nieto and J. Lopez, *Contextualizing Heterogeneous Information in Unified Communications with Security Restrictions*, In Computer Communications Journal, Accepted for publication.

Conferencias internacionales:

Ci1. A. Nieto, and J. Lopez, *Security and QoS tradeoffs: towards a FI perspective*, In Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on, IEEE, pp. 745-750, 2012.

Ci2. A. Nieto, and J. Lopez, *Security and QoS relationships in Mobile Platforms*, In The 4th FTRA International Conference on Computer Science and its Applications (CSA 2012), Lecture Notes in Electrical Engineering 203, Springer Netherlands, pp. 13-21, 2012.

Ci3. A. Nieto, and J. Lopez, *A Context-based Parametric Relationship Model (CPRM) to Measure the Security and QoS tradeoff in Configurable Environments*, In IEEE International Conference on Communications (ICC'14), IEEE Communications Society, pp. 755-760, 2014.

Ci4. A. Nieto, *Evaluation of Dynamic Instantiation in CPRM-based Systems*, In 9th International Conference on Risk and Security of Internet and Systems (CRiSIS'14), vol. 8924, Springer, pp. 52-66, 2014.

Ci5. A. Nieto, and J. Lopez, *Security and QoS Tradeoff Recommendation System (SQT-RS) for Dynamic Assessing CPRM-based Systems*, In 10th ACM International Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet'14), ACM, pp. 25-32, 2014.

Conferencias nacionales (españolas):

Cn1. A. Nieto, and J. Lopez, *Herramienta para la Compensación de Parámetros de QoS y Seguridad*, In XIII Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2014), pp. 303-308, 2014.

Otras publicaciones:

A1. A. Nieto, and J. Lopez, *Traffic Classifier for Heterogeneous and Cooperative Routing through Wireless Sensor Networks*, In Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on, IEEE, pp. 607-612, 2012.

Esta tesis ha sido financiada por el Ministerio de Economía y Competitividad, bajo el marco del "Programa Nacional de Formación de Personal Investigador" (FPI). Algunas partes de este trabajo fueron realizadas durante la estancia pre-doctoral del autor en el Departamento de Información e Ingeniería de Sistemas de Comunicaciones, en la Universidad del Aegean, en la Isla de Samos (Grecia).

C.2. Clasificación de parámetros de seguridad y QoS

Pese a que el anásis de clasisificación de parámetros llevado a cabo en el Capítulo 2 es bastante extenso, en este resumen nos centramos en la consecuencia directa de este estudio, que es la clasificación paramétrica que finalmente usaremos para nuestro análisis. Clasificamos los parámetros siguiendo una estructuración por niveles abstractos, que pueden ser definidos por el usuario. En nuestro caso, dados los parámetros evaluados, consideramos cinco niveles de abstracción donde clasificar los parámetros comunes de las redes candidatas. Estos niveles se ajustan adecuadamente a los casos de uso propuestos, sin embargo, no son restrictivos. Es decir, las soluciones propuestas son lo suficientemente flexibles para que nuevos niveles sean definidos o bien los existentes sean reemplazados por otros niveles más adecuados según el escenario concreto de aplicación. Estos niveles se definen en los *scripts* que definen la instanciación de los modelos propuestos en esta tesis, por lo que son fácilmente modificables (ejemplo en Anexo A).

A continuación, detallamos las particularidades de cada nivel para la clasificación que hemos considerado en esta tesis. Estos niveles son lo suficientemente generales como para adaptarse a nuestros casos de uso.

C.2.1. Requisitos de alto nivel

El primer nivel agrupa los requisitos de alto nivel, o HLR por sus siglas en inglés *High-Level Requirements*. Es decir, conceptos comúnmente entendibles por el usuario, o más cercanos a éste. Los requisitos de seguridad (ej. autenticación, privacidad, autorización, confidencialidad) y tipos de QoS o tipos de tráfico (ej. multimedia, interactivo, etc.) se encontrarían definidos a este nivel. Además, a este nivel pueden ser considerados requisitos o mediciones de calidad de la experiencia (QoE), con el fin de evaluar el impacto de los requisitos en mejorar la experiencia del usuario. De hecho, la percepción del usuario es un parámetro muy subjetivo, porque depende de la opinión del usuario, que está basada en experiencias personales.

C.2.2. Propiedades locales

Las propiedades locales definen la composición del dispositivo. Por ejemplo, mejoras hardware como el uso de múltiples antenas, NFC o cualquier otra característica en el dispositivo serían contempladas a este nivel. Para mantener mayor simplicidad, aquellos elementos usados para interactuar con la red se situarán en el nivel de comunicaciones (siguiente). De esta manera, este nivel está fuertemente ligado a las características que, independientemente de la conexión del dispositivo a una red, describen las propiedades del dispositivo.

Por ejemplo, el consumo enerjético es muy dependiente de la transmisión de red. Sin embargo, este parámetro es afectado también independientemente de su conexión. Lo mismo sucede con la memoria disponible.

Por lo tanto, los parámetros a este nivel dependen de la implementación de los mecanismos y las aplicaciones desplegados y del equipamiento empotrado en el dsipositivo desde fábrica.

C.2.3. Comunicación

El conjunto de parámetros a este nivel están relacionados con la comunicación del nodo con la red. Por ejemplo, el parámetro *data rate* define el número de bits enviados por unidad de tiempo. Además, este parámetro se encuentra estrechamente vinculado con la transmisión de información y el tamañao de paquete. Por lo tanto, la transmisión de datos es considerada a este nivel. La máxima tasa de transmisión de datos, por ejemplo, puede ser considerada como un parámetro usado para las mediciones, y encontrarse a otro nivel.

En cuanto a los mecanismos de seguridad, se supone que deberían ser aplicados en niveles posteriores, a no ser que se implementen a nivel de protocolo de comunicaciones en el envío de datos. Por ejemplo, a este nivel se asume que la encriptación de los datos estaría ya realizada, que depende de si el dispositivo, a nivel local, por sus caracteristicas, puede realizarla.

Este nivel se compondría de recursos para la gestión de la información local de los interfaces de red (ej. fuerza de la señal), características y elementos que pueden ser usados a este nivel (ej. el tiempo que el interfaz está a la escucha), ataques relacionados con el interfaz que no se encuentren contemplados a nivel de entorno (ej. si el nodo tiene comportamiento malicioso y se encuentra en modo promíscuo), y consecuencias debidas a la baja calidad en la comunicación (ej. retransmisión de datos).

C.2.4. Mediciones

A este nivel se encuentran aquellos parámetros relacionados con las mediciones de los parámetros de red. Parámetros típicos para medir el rendimiento que pueden expresarse en función de formulación matemática son considerados a este nivel. Algunos se encuentran estrechamente ligados con el nivel de comunicaciones y en algunos casos puede ser interesante contemplarlos a nivel de comunicaciones, por ejemplo, para un análsis conjunto de parámetros particular.

C.2.5. Entorno

Finalmente, en a nivel de entorno lo parámetros relacionados con las condiciones de red y las características del entorno son considerados. Por ejemplo, el nivel de ruido, el número de dispositivos o la probabilidad de escuchas en la red pueden ser considerados a este nivel.

En efecto, el entorno define el escenario como tal y el contexto actual donde el dispositivo se encuentra. Por lo tanto, los parámetros a este nivel pueden variar considerablemente durante el tiempo de vida de un dispositivo móvil, o bien si el entorno está sometido a cambios permanentes, independientemente de la movilidad.

C.3. Extracción de información para el modelo

Los niveles anteriores se definen para agrupar parámetros que posteriormente se relacionarán entre sí, en base a una serie de características por las que se rigen los modelos propuestos para el análisis. Sin embargo, aún no hemos detallado cómo extraemos información para incluirla en el estudio paramétrico. En nuestro caso, consideramos dos fuentes potenciales de información:

(A). Relaciones basadas en la formulación (matemática). Esta es la información relacional más fiable, ya que las relaciones se extraen directamente de expresiones matemáticas.
(B). Relaciones basadas en la literatura. Esta es información que puede depender fuertemente del contexto. También permite definir relaciones subjetivas. Identificamos relaciones a partir de expresiones en lenguaje natural que encontramos en trabajos relacionados, o bien que un usuario pueda expresar en base a su conocimiento de una materia ó un entorno concreto. Una vez que identificamos estas expresiones, trasladamos la interpretación de estas expresiones al modelo.

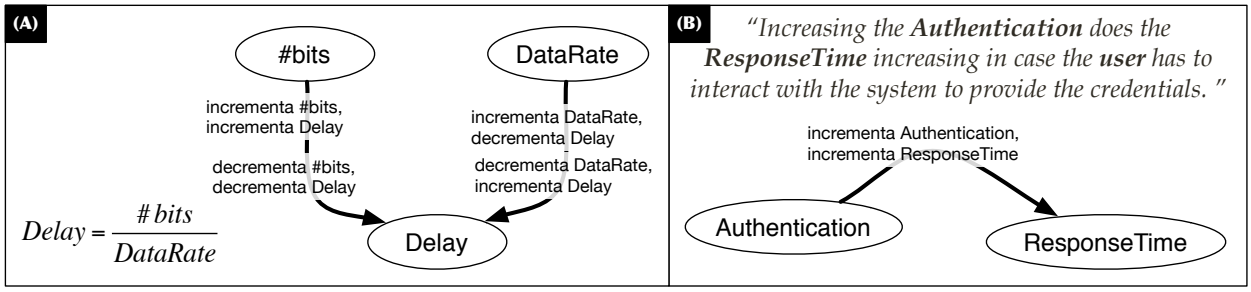


Figura C.2: (A)–Relaciones basadas en la formulación, (B)–Relaciones basadas en la literatura.

La Figura C.2 muestra un ejemplo de ambos casos. Nuestro modelo, presentado en el Capítulo 3, y resumido en los siguientes apartados de este anexo, define cómo expresar las relaciones paramétricas usando un lenguaje y el conjunto de fórmulas matemáticas requeridas para interpretar las relaciones automáticamente.

C.4. Modelos para el análisis

Diversos modelos para el análisis conjunto de aspectos de seguridad y calidad de servicio (QoS) emergen como consecuencia directa de la amplia diversidad de dispositivos que componen las redes heterogéneas. En particular, los modelos genéricos para el análisis del balanceo o compensación de requisitos de seguridad y QoS son, desde el punto de vista práctico, los más relevantes para las redes heterogéneas de composición dinámica, en las que no se puede prever con gran exactitud los dispositivos que formarán la red.

Definimos un modelo genérico para el análisis de la compensación de seguridad y QoS como aquel que se abstrae de detalles específicos de una tecnología y que ofrece la posibilidad de integrar en el estudio cualquier tipo de tecnología y dispositivo a distinto nivel. De hecho, podemos encontrar algunos ejemplos de modelos genéricos en la literatura que se ajustan en mayor o menor medida a esta definición [30, 111], enfoques más específicos sobre seguridad o QoS [14, 133, 134, 135], y otros, que emplean técnicas paramétricas para mejorar la configuración de servicios [136]. Por ejemplo, en [111] se emplean técnicas de *model checking* para verificar las equivalencias entre especificaciones de seguridad y QoS, con el objetivo de controlar los flujos de información ilegítimos en el sistema. No obstante, obliga a definir un modelo de comunicación entre las aplicaciones del sistema, restringiendo por tanto su ámbito de uso. Alternativamente, en [30] se define un modelo basado en el contexto, que proporciona una función de utilidad para tener en consideración las preferencias del usuario. Sin embargo, no permite medir el impacto que unos parámetros del sistema tienen sobre otros, y el conjunto de contextos es limitado.

No obstante, el análisis conjunto de los mecanismos de seguridad y QoS debería basarse en el estudio de relaciones paramétricas, es decir, relaciones de dependencia entre los parámetros que definen la composición de los mecanismos de seguridad y los de QoS. Sólo así es posible tener un control de grano fino sobre el nivel de abstracción del análisis, y combinar parámetros heterogéneos dentro de un mismo estudio. Además, definir estas relaciones en base a un contexto es básico para expresar la relevancia de los parámetros, relaciones, operaciones y otros componentes y propiedades, que tienen cabida en el sistema de información.

En concreto, el análisis de la seguridad y la QoS debería realizarse considerando los siguientes reguisitos:

i. Permitir diferentes niveles para el análisis. La mayoría de los análisis de tradeoff entre seguridad y QoS se centran en entornos específicos y adaptar estas soluciones a otros entornos no es trivial (p.ej. a nivel de servicios, composición de servicios).
ii. Considerar información parcial del entorno. En entornos dinámicos, las características de los dispositivos que forman el entorno pueden cambiar a lo largo del tiempo. De hecho, puede darse el caso de que al comenzar al desplegar los dispositivos no conozcamos cómo se comportarán hasta transcurrido un tiempo. Por ello, las soluciones para el análisis de la seguridad y la QoS deben considerar información parcial (comportamiento conocido a grandes rasgos) que pueda completarse con nueva información cuando se conozca.
iii. Integrar conceptos subjetivos como parte de los valores a considerar para la compensación. El criterio subjetivo es muy importante para la compensación. Un sistema es inviable si no satisface las demandas que son más relevantes para sus usuarios.

iv. Permitir conceptos heterogéneos que den cabida a diferentes tipos de redes/dispositivos. Esto es muy importante para integrar información procedente de diversos estudios que pueda enriquecer el análisis.

Dado que en la literatura actual no encontramos modelos con las características citadas, a continuación definimos un modelo para la definición de parámetros y relaciones dentro de un marco común y genérico que permita la posterior evaluación de las dependencias entre parámetros de seguridad y de QoS.

C.4.1. Modelo paramétrico general

El modelo para definir relaciones paramétricas, denotado por sus siglas en inglés, *Parametric Relationship Model* (PRM), tiene como objetivo permitir definir un escenario en base a un conjunto de parámetros y las relaciones entre los parámetros, dadas en función del impacto que la modificación de los parámetros tiene en el resto de parámetros por medio de sus relaciones directas.

El modelo define cómo el efecto de las modificaciones es propagado por el resto de parámetros, que finalmente forman un grafo de dependencias. El análisis conjunto de las modificaciones permite extraer información detallada sobre el grado de dependencia o influencia de parámetros o conjuntos de éstos. Los parámetros son agrupados en niveles genéricos, y definidos en base a un tipo (ej. seguridad, QoS, rendimiento).

La Tabla C.1 recoge un resumen de la formulación matemática para definir parámetros y relaciones entre éstos.

Tabla C.1: Definiciones asociadas a un Modelo de Relaciones Paramétrico (PRM).

Conjunto de Formulación Básico (BFS)		Conjunto de Formulación Complejo (CFS)	
$D^{+} :: aD^{+}b \Rightarrow (\Delta a \rightarrow \Delta b)$	(1)	$D^{c} :: (\Delta a \rightarrow \Delta b) \wedge (\nabla a \rightarrow \nabla b) \equiv aD^{+}b \wedge aD^{\neg +}b$	(5)
$D^{-} :: aD^{-}b \Rightarrow (\Delta a \rightarrow \nabla b)$	(2)	$D^{t} :: aD^{c}b \wedge bD^{c}a$	(6)
$D^{\neg +} :: aD^{\neg +}b \Rightarrow (\nabla a \rightarrow \nabla b)$	(3)	$D^{\neg c} :: (\Delta a \rightarrow \nabla b) \wedge (\nabla a \rightarrow \Delta b) \equiv aD^{-}b \wedge aD^{\neg -}b$	(7)
$D^{\neg -} :: aD^{\neg -}b \Rightarrow (\nabla a \rightarrow \Delta b)$	(4)	$D^{i+} :: (\Delta a \rightarrow \Delta b) \wedge (\nabla a \rightarrow \Delta b) \equiv aD^{+}b \wedge aD^{\neg -}b$	(8)
		$D^{i-} :: (\Delta a \rightarrow \nabla b) \wedge (\nabla a \rightarrow \nabla b) \equiv aD^{-}b \wedge aD^{\neg +}b$	(9)

Influencia Acumulativa (ι)		Dependencia Acumulativa (δ)	
$\iota(a) = \lvert I_a \rvert = \lvert \{x \lvert x \rightarrow a \vee xRa, x \neq a, x \in P\} \rvert$	(10)	$\delta(a) = \lvert D_a \rvert = \lvert \{y \lvert a \rightarrow y \vee aRy, y \neq a, y \in P\} \rvert$	(11)
		$xRy \iff x \rightarrow y \vee \exists k \lvert k \in D_x \wedge k \in I_y$	(12)

Impacto al Incrementar (Δ) y Decrementar (∇) un Parámetro x

$$u(x,\omega) = \begin{cases} \Delta x & \text{if } \omega > 0; \\ \nabla x & \text{if } \omega < 0; \end{cases} \quad (13)$$

$$\Delta x \implies \forall y \lvert xRy, v(y) = v(y) + \Omega(R, \Delta x) \wedge u(y, \Omega(R, \Delta x)) \quad (14)$$

$$\nabla x \implies \forall y \lvert xRy, v(y) = v(y) + \Omega(R, \nabla x) \wedge u(y, \Omega(R, \nabla x)) \quad (15)$$

$\Omega(R, op(x))$, $R \in \{+, -, \neg+, \neg-, c, t, \neg c, i+, i-\}$, $op \in \{\Delta, \nabla\}$: (16)

	$+$	$-$	$\neg+$	$\neg-$	c	t	$\neg c$	$i+$	$i-$
Δx	$w_{x,+}$	$-w_{x,-}$	ntd	ntd	$w_{x,c}$	$w_{x,t}$	$-w_{x,\neg c}$	$w_{x,i+}$	$-w_{x,i-}$
∇x	ntd	ntd	$-w_{x,\neg+}$	$w_{x,\neg-}$	$-w_{x,c}$	$-w_{x,t}$	$w_{x,\neg c}$	$w_{x,i+}$	$-w_{x,i-}$

La limitación de un PRM se encuentra en que es una estructura estática, y debe ser ampliada para permitir la inclusión dinámica de información, en base a un contexto. Entienda, que tal y como se encuentra en este punto definido, el modelo PRM requiere ser modificado cada vez que conozcamos más información sobre el sistema a evaluar. Esto supone cambios constantes sobre los scripts de evaluación que implementan el modelo, y que detallaremos posteriormente.

C.4.2. Modelo paramétrico basado en el contexto

En base al paradigma actual y futura convergencia de las redes, definimos un modelo para estudiar las relaciones paramétricas basado en el contexto, denominado CPRM por sus siglas en inglés (*Context-based Parametric Relationship Model*). Dicho modelo define la estructura de un sistema en base a un conjunto de parámetros y sus relaciones, un conjunto de operaciones que definen efectos sobre los parámetros dependientes, y una estructura de pesos que define la relevancia subjetiva y no subjetiva de los componentes del modelo.

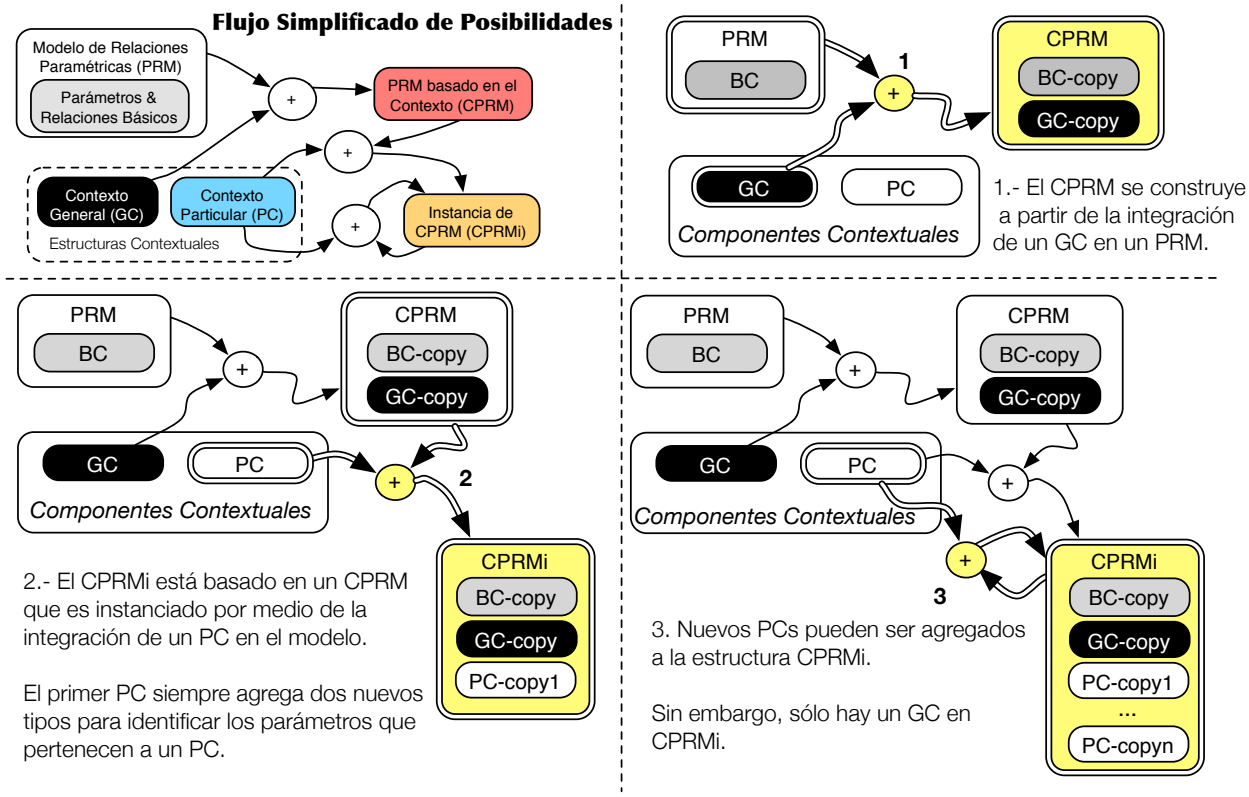


Figura C.3: Integración de Contextos y generación de Esquemas.

Por ejemplo, un administrador puede considerar subjetivamente que la confianza es un parámetro clave para la subsistencia del sistema de información. En ese caso, el parámetro confianza tendría un peso mayor en el sistema que otros parámetros menos relevantes dado el caso. A su vez, los mecanismos que implementen el valor de confianza podrían heredar la relevancia o peso de su parámetro padre, en este caso, el parámetro confianza. Estos valores subjetivos estarían sujetos a la variabilidad del contexto, de forma que en un momento dado, ya sea por las medidas de seguridad adoptadas o por el entorno donde está el individuo, su relevancia puede variar. Por ejemplo, en un entorno familiar bien definido, el parámetro confianza y los mecanismos estrechamente dependientes podrían relajar su relevancia de no existir otras dependencias que se lo impidan. Esto es así, porque en el contexto *hogar* el individuo podría asumir que la confianza viene dada por su ubicación. Aunque no tiene porqué ser así.

Además, el modelo también contempla valores no subjetivos; destinados a definir el im-

pacto o reacción en cadena que podría ocasionar una dependencia. Estos valores, se definen, en primer lugar, de forma aproximada en las dependencias del contexto general (GC, *General Context*), mientras que, una vez que los parámetros son instanciados, el peso es actualizado al contexto particular (PC, *Particular Context*).

La Figura C.3 resume el proceso de alternativas para la integración de contextos. Partimos de un conjunto de parámetros que definen, de forma general, el escenario a evaluar. Este contexto base (BC) es fijo y no varía, y se encuentra en el PRM[1]. Pueden variar los pesos/relevancia de los parámetros, pero el BC siempre queda presente a la espera de que sus parámetros, relaciones, niveles, tipos y operaciones tomen los valores de contexto definidos en el GC y, posteriormente, en los sucesivos PCs. El BC es el resultado de un proceso de análisis exhaustivo sobre las arquitecturas y el entorno donde la herramienta tendrá cabida. En nuestro caso, surge del estudio de mecanismos de Seguridad y QoS en el *Internet del Futuro*. Aunque la herramienta propuesta permite definir BC personalizados, siendo por tanto extensible a otros ámbitos de estudio, nuestro principal objetivo es su uso para el análisis de la compensación entre parámetros de Seguridad y QoS. En efecto, el BC que proporcionamos define dichos tipos de relaciones y no otros, que deberían ser agregados con posterioridad, según el caso.

En consecuencia, un modelo CPRM maneja los siguientes elementos:

- Estructuras de Modelo. Definen el estado actual de un sistema definido en base al CPRM. Es decir, conjunto de parámetros y dependencias definidas entre los parámetros y sus operaciones. Estas estructuras son: PRM, CPRM, $CPRM_i$, dependiendo de los contextos que integran (Figura C.3).
- Estructuras de Contexto. Definen el contexto a agregar, es decir, nuevo comportamiento que modifica el estado actual de una estructura de modelo.

Las estructuras de contexto generan nuevos estados de modelo. Como puede apreciarse en la Figura C.3, PRM $\in$ CPRM $\in$ $CPRM_i$, dado que la diferencia entre estos tres esquemas es que un CPRM contiene un GC, y un $CPRM_i$ contiene un GC y al menos un PC. La integración de los contextos en los modelos, debe realizarse verificando el cumplimiento de una serie de reglas, mostradas en la Tabla C.2 para garantizar que el sistema paramétrico final es coherente con la definición de modelo CPRM.

C.5. Herramienta para la compensación paramétrica

Para evaluar la usabilidad del modelo, se ha desarrollado una herramienta conforme al modelo CPRM a la que denominaremos SQT, por sus siglas en inglés *Security and QoS tradeoff Tool*. SQT proporciona un interfaz gráfico para la administración (Figura C.4) permitiendo al usuario importar esquemas de modelo (PRM, CPRM, $CPRM_i$) y de contexto (GCs y PCs), salvar cualquier esquema en ficheros para su posterior uso y modificación, así como el espacio de trabajo completo, con los modelos y contextos asociados. También es posible extraer o eliminar contextos de los esquemas de modelo contextuales (CPRM,

[1]De cara a nuestro estudio, el BC no representa una estructura contextual, ya que los parámetros en el BC (en el PRM) carecen de pesos.

Tabla C.2: Reglas (R) y Reglas de Actuación (AR).

Regla	Regla de Actuación
R1 Un parámetro en el PC está relacionado con al menos un parámetro en el PRM (padre).	AR1 De lo contrario, el parámetro independiente es considerado como un nuevo parámetro y agregado al PRM para mantener la consistencia.
R2 Sean P(x) y P(y) los conjuntos de padres de x y de y, respectivamente: $\|P(x) \cup P(y) \subset PRM$. Si existe d(x,y), entonces $\exists k \in P(x) \wedge \exists z \in P(y) \| d(k,z)$.	AR2 De lo contrario, dicha relación entre padres debe ser agregada al PRM para mantener la consistencia del modelo, con $w_d(k,z) = 0$.
R3 Un parámetro en el PC hereda las relaciones de sus padres, por defecto: si z pertenece a P(x), y d(z,k), entonces d(x,k) es posible, con peso w(z,k) por defecto.	AR3 De lo contrario, la relación d(x,k) es agregada con peso w(z,k) $\forall k$. Si $\exists p \| k \in P(p)$ y por tanto, conforme R2, $d(x,p)$, w(x,p) no es modificado.
R4 Un parámetro x hereda el nivel de sus padres y el tipo de sus padres. Cuando $\exists k, z \in P(x) \| type(k) \neq type(z)$, entonces $type(x) = [type(k)type(z)]$.	AR4 De lo contrario, el modelo de decisión puede fijar el tipo del nivel de un parámetro, pero, incluso entonces, el nivel y el tipo final concueran con el nivel y el tipo de un parámetro p en P(x).

$CPRM_i$). El objetivo final es el análisis dirigido a la obtención de mediciones sobre el modelo de dependencias:

1. Incremento y decremento de parámetros.
2. Selección de conjuntos de parámetros por tipo y nivel.
3. Selección de parámetros instanciados (llamados padre) o bien de sus instancias (llamadas hijos) para distinguir entre diferentes opciones de configuración.
4. Calcular árboles de dependencias específicos para un parámetro, con el fin de posibilitar un examen más exhaustivo sobre el proceso de incremento/decremento.

SQT permite visionar los resultados mediante diagramas de barras superpuestas que indican el impacto de un conjunto de parámetros en el resto de parámetros del sistema, o bien sobre un conjunto específico, en base al tipo/nivel, etc. También es posible seleccionar un parámetro en particular, como veremos en el caso de estudio. Otro modo de representación empleado es el uso de grafos, por medio de GraphViz. Así, el modelo de dependencias es representado mediante un grafo, en el que los parámetros se muestran acorde con la representación del tipo y agrupados por niveles según se define en el modelo.

Para posibilitar el cumplimiento de tales requisitos, la herramienta implementa un conjunto de reglas de coherencia definidas para el modelo CPRM, mostradas en la Tabla C.2. El cumplimiento de estas reglas garantiza que el sistema paramétrico final mantiene la coherencia entre las dependencias.

El primer prototipo de SQT proporciona un interfaz de usuario (GUI) para la administración, diseñada para mostrar todas las opciones disponibles al usuario en una única ventana (Figura C.4). Se encuentra dividido en siete secciones o paneles:

1. Panel de trabajo de PRM. Permite seleccionar un PRM desde un fichero, o bien cargar un PRM definido por defecto a modo de ejemplo. Esto incluye también a las estructuras CPRM y $CPRM_i$, dada que ambas siguen el formato de un PRM ampliado.

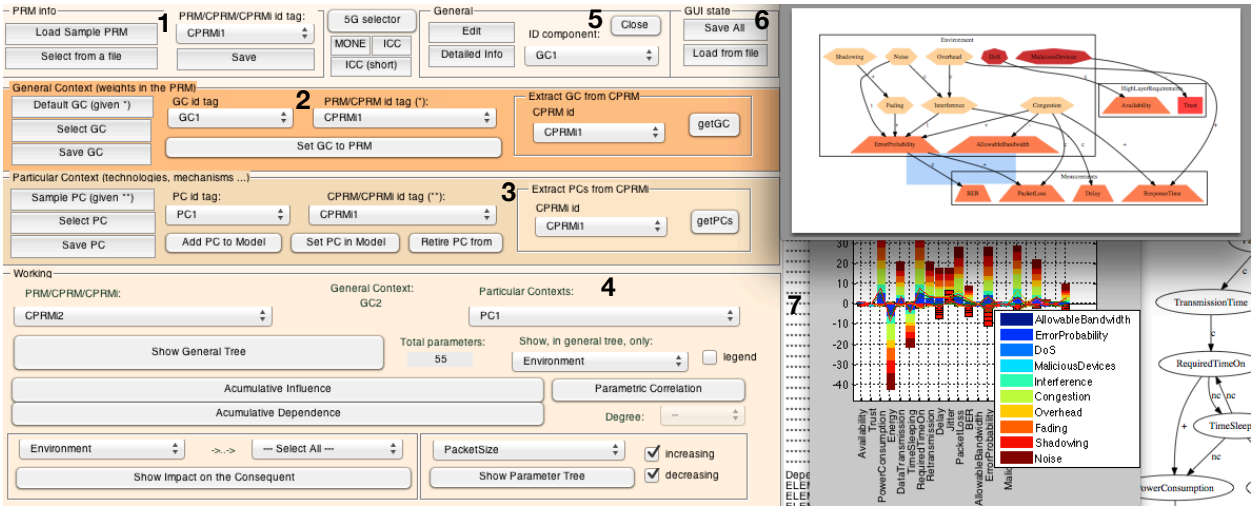


Figura C.4: Interfaz de Administración SQT.

2. Panel para Elementos de Contexto Generales (CPRM, GC). Destinado a la selección de GCs y estructuras de modelo que permitan agregar un GC, o extraerlo.
3. Panel para Elementos de Contexto Instanciados ($CPRM_i$, PC). Destinado a la selección de PCs y estructuras de modelo que permitan agregar PCs, o extraerlos.
4. Panel de Trabajo. Permite la extracción de información desde contextos. Incluye las operaciones definidas por el modelo y reflejadas en el diagrama de componentes.
5. Panel General. Panel para operar con cualquiera de los elementos definidos en el espacio de trabajo (S). Desde este panel puede cerrarse ó eliminar de S cualquier elemento.
6. Panel de Estado de la GUI. Permite guardar el espacio de trabajo (todos los contextos y estructuras de modelo) así como importar un espacio de trabajo guardado en un fichero.
7. Panel informativo. Permite mantener información de registro visible sobre las operaciones realizadas y los errores posibles detectados.

Cabe destacar, que cualquier estructura de modelo puede salvarse en un nuevo fichero. Esto permite que podamos modificar un ejemplo existente para realizar las primeras pruebas.

C.6. Sistema de recomendaciones

El sistema de recomendaciones para SQT (SQT-RS) es un módulo adicional a SQT que se define con el objetivo de mitigar el efecto que el número de parámetros ocasiona en la visualización de los resultados. SQT muestra resultados gráficos del impacto y dependencia de los parámetros en el sistema, y conforme aumenta el número de parámetros estos resultados son difícilmente distinguibles e interpretables por un usuario no experimentado. Para facilitar la toma de decisiones en estos casos, así como en otros más simples que puedan ayudar a entender mejor la herramienta, SQT-RS muestra recomendaciones en lenguaje natural sobre las acciones a realizar para satisfacer los objetivos seleccionados por un usuario, usando interfaces gráficas proporcionadas por el módulo.

El fin perseguido con SQT-RS es la gestión de la información dinámica deducible del sistema basado en CPRM, de forma que puedan mostrarse recomendaciones sencillas de entender por el usuario, sobre las acciones a realizar para conseguir una serie de objetivos, indicados también por el usuario.

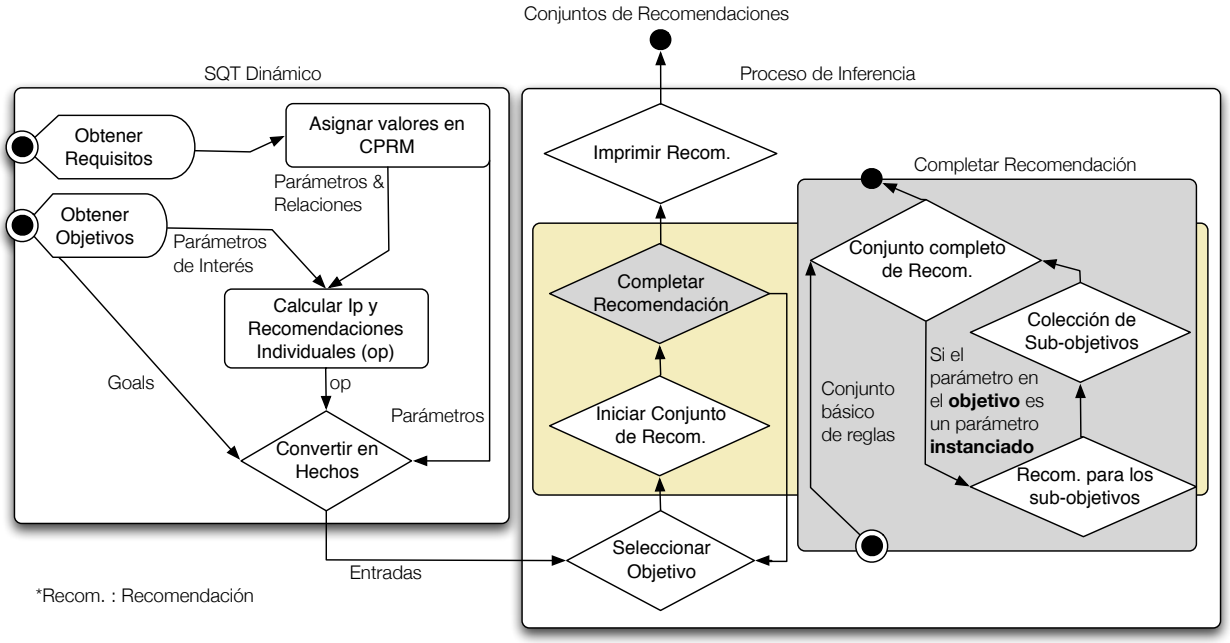


Figura C.5: Pasos hasta la obtención de Recomendaciones.

Se definen por tanto las estructuras de objetivo y requisito. Un objetivo se basa en nuestro sistema en un criterio maximizar o minimizar y un parámetro que es el foco del objetivo. Por otra parte, un requisito es un valor que toma un parámetro en el sistema. Se proporciona un interfaz para que el usuario pueda incluir esta información en SQT-RS y, en base a esta nueva información, SQT-RS procesa los parámetros del CPRM activo en SQT para proporcionar recomendaciones que satisfagan, en la medida de lo posible, los objetivos, o bien que indique las contraindicaciones o incompatibilidades para su satisfación.

SQT-RS se basa en la formulación relativa a la influencia y dependencia acumulativa, y a las mediciones del impacto al incrementar y decrementar parámetros. Así mismo, considera un conjunto de propiedades y definiciones del modelo CPRM:

Definición (Prop.1): Si hay una recomendación individual para un parámetro instanciado, es decir, $\exists op_j(x_j), x_j \in I_P, type(x_j) == instantiated | g(P) = 1$, entonces, hay una recomendación para cada instancia del parámetro.

Definición (Prop.2): Un sistema de recomendaciones basado en CPRM considera el problema de maximizar/minimizar un parámetro instanciado como el problema de maximizar/minimizar las instancias del parámetro. Es decir: si $\exists g(p)\ | type(p) == instantiated$, entonces, $R_p = R_{p_i}\ | p \in P(p_i)$.

Definición (Prop.3): La información sobre una instancia es proporcionada como hecho cuando cualquiera de sus padres son considerados en un objetivo o en una recomendación individual. Además, si el parámetro es parte de un objetivo, entonces las recomendaciones individuales para la instancia son proporcionadas también.

Las propiedades anteriores, ayudan a la elección de reglas que son implementadas en CLIPS y usadas desde Jess para la generación de un .jar que es llamado por SQT-RS. El archivo de reglas en CLIPS (Anexo B) puede ser modificado para modificar la forma de visualizar los datos, o para incluir nuevas reglas de inferencia. Estas modificaciones pueden hacerse al margen de la herramienta. A su vez, SQT-RS fue implementado en MATLAB e integrado en SQT. En general, este módulo combina los lenguajes: MATLAB, para recibir información desde SQT y para aportar la información de las recomendaciones calculadas a SQT, CLIPS para los hechos y las reglas de inferencia, y Jess para realizar los cálculos dinámicos. La parte dinámica de SQT-RS se encuentra en la captación de hechos dado un CPRM seleccionado. Estos hechos serían la entrada para el programa en Jess que usaría el conjunto de reglas escritas en CLIPS para inferir los resultados. SQT-RS orquesta todo este proceso y permite la interactuación con el usuario para la entrada de valores.

Los hechos se generan en función de los objetivos seleccionados y el tipo de parámetro en los objetivos. Los parámetros instanciados se tratan de forma especial, ya que para ellos existe más información en el sistema, ya que las instancias proporcionan mecanismos específicos para ellos. El ciclo de vida completo de SQT-RS desde la captación de objetivos hasta la obtención de recomendaciones se ilustra en la Figura C.5.

Durante el proceso de determinar las recomendaciones, también se consideran los posibles conflictos que puedan impedir o dificultar el cumplimiento de los objetivos. En nuestra implementación, consideramos tres tipos de conflictos:

c1. Cuando la modificación de un parametro x, ya sea para incrementarlo o decrementarlo, impide que el objetivo se satisfaga.
c2. Cuando un parámetro intermedio i, requiere ser incrementado y también decrementado para satisfacer el objetivo (operaciones opuestas).
c3. Cuando diferentes objetivos requieren operaciones opuestas para satisfacer sus respectivas recomendaciones.

La identificación de conflictos c1 y c2 puede ser llevado a cabo a partir del análisis de los atributos de los hechos CLIPS generados a partir del CPRM seleccionado. Luego este análisis podría ser efectuado al inicio del proceso de recomendación. Sin embargo, los conflictos de tipo c3 pueden ser más complejos, porque involucran múltiples objetivos y sus respectivas recomendaciones. Por lo tanto, su identificación debería efectuarse al final del proceso de recomendación.

Respecto al efecto en el rendimiento de esta solución, el conjunto de hechos seleccionable por SQT-RS es minimizado para procurar optimizar su uso. Es posible convertir a hechos toda la información en un CPRM, pero esto resulta bastante abusivo desde el punto de vista del espacio y computacional. Por ello los hechos en SQT-RS son seleccionados en base a una lista de parámetros de interés que es determinada por los parámetros objetivo y las recomendaciones individuales (op), que se deducen de las operaciones previamente mencionadas.

C.6.1. Ejemplo

Aportamos un ejemplo para mostrar cómo proceder ante una recomendación de SQT-RS. Supongamos que disponemos de los parámetros (instanciados e instancias) y las relaciones mostradas en la Figura C.6.

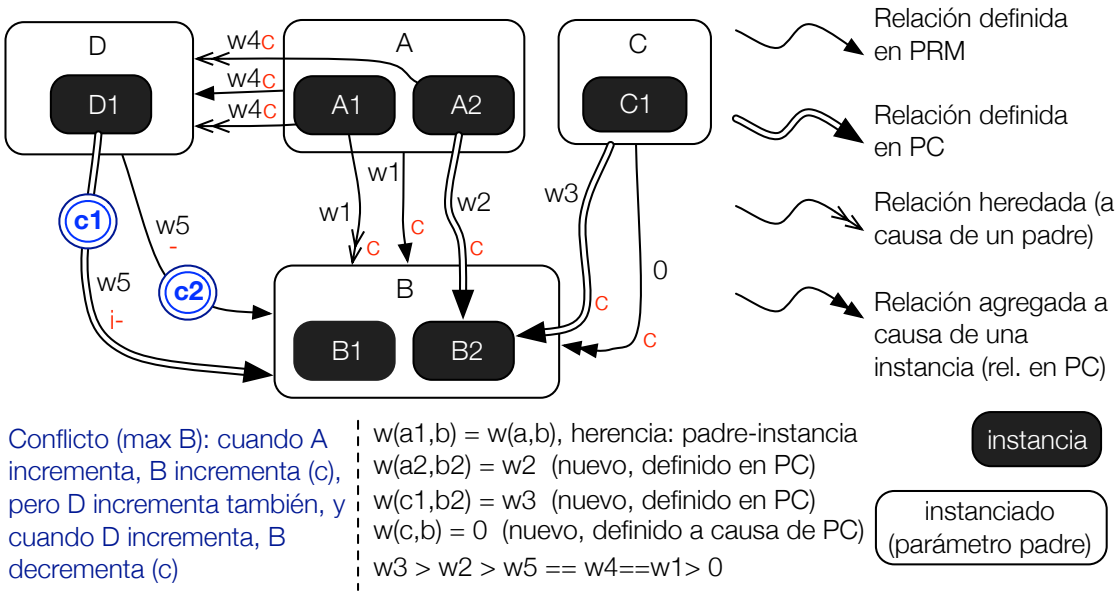


Figura C.6: Ejemplo de instanciaciones y conflictos.

Si observamos las relaciones sin parámetros instanciados, tenemos los parámetros A, B, C y D. Las instancias, serían los parámetros A1, A2, B1, B2, C1 y D1. si pretendemos el objetivo de maximizar B, el sistema de recomendaciones debe indicarnos qué debemos hacer.

Recordamos que, las relaciones complejas (c) indican que cuando el antecedente incrementa, entonces el consecuente incrementa también, y que cuando el antecedente decrementa, el consecuente también decrementa. La relación negativa implica que cuando incrementa el antecedente, decrementa el consecuente, y la relación independientemente negativa ($i-$) hace que siempre que se modifique el parámetro del antecedente el parámetro del consecuente decremente.

La Figura C.7 muestra el resultado del sistema de recomendaciones para el esquema anterior, cuando se requiere maximizar B.

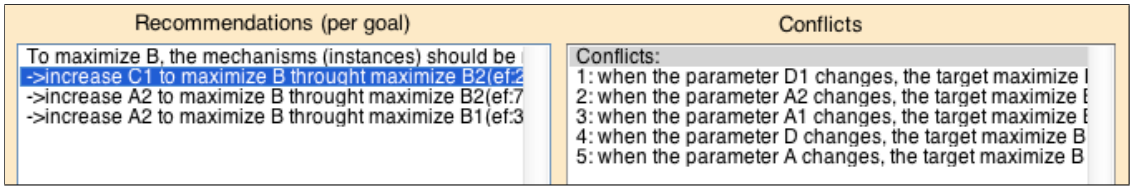


Figura C.7: Ejemplo: Recomendaciones y conflictos en la GUI.

En este caso, el sistema recomienda incrementar C1 y A2 para maximizar las instancias de B (y por lo tanto, la propiedad B). Pero, además, muestra una lista de conflictos que

deben ser considerados. Los conflictos en este caso indican que la solución no es trivial, dados los conflictos, la solución pasa por decidir qué parámetros son más relevantes para el usuario que esté evaluando el sistema. Como ya se apreciaba en la Figura C.7, A es un parámetro conflictivo, porque cuando incrementa, puede incrementar B, pero también D, que a su vez decrementa B. Como puede observarse, es finalmente a través de C1 que podría incrementarse B, ya que los conflictos no afectan al parámetro C. Los resultados también nos indican que otra posibilidad, es sopesar el efecto que el incremento de A tiene sobre B comparado con el efecto que decrementar D tiene sobre B. Esto formaría parte de un análisis más detallado, fruto de nuevos tradeoffs.

C.7. Casos de uso y resultados

Para evaluar la usabilidad del modelo CPRM y la funcionalidad de SQT-RS para el análisis de la compensación entre parámetros de seguridad y QoS, hemos desarrollado un conjunto de pruebas tomando como casos de uso escenarios estrechamente relacionados con la Internet del Futuro. Conforme al primer capítulo de la tesis, la relevancia de las redes de sensores, las redes celulares y las redes MANET para la futura convergencia de redes en la IF quedó justificada. Es por este motivo que los casos de uso propuestos pretenden cubrir casos posibles en las tres redes mencionadas.

C.7.1. Caso de uso 1: autenticación en WSN

El ejemplo propuesto para el caso de análisis está basado en el funcionamiento de una red de sensores. Como tal, considera como parte del conjunto de parámetros del contexto base (BC) aquellos parámetros generales que pueden estar relacionados con una red de sensores, así como las relaciones entre éstos. Adaptado al caso que nos ocupa, los parámetros del BC son mostrados en la Tabla C.3[2]

Aunque el GC por defecto para estos parámetros es inicialmente establecido con peso igual a 1 para todos los parámetros ($\forall p | p \in PRM, w_p = 1$), es posible establecer un GC subjetivo, basado en nuestras prioridades de administración. Por ejemplo, aumentar la relevancia/impacto del cifrado (*Encryption*), de tal forma, que todos los parámetros que tengan una dependencia en la que *Encryption* se encuentre en el antecedente serán más afectados que el resto de parámetros. Los parámetros afectados por el incremento del parámetro *Encryption*, pueden consultarse usando el árbol paramétrico particularizado para un parámetro. El efecto, sin embargo, podrá variar dependiendo del tipo de relación definida entre los parámetros y de los pesos definidos para las relaciones. Por ahora, todos los pesos para las relaciones tienen valor unitario ($w_d = 1, \forall d : A \rightarrow B | d \in PRM$). Estos pesos pueden modificarse con un GC, pero en nuestro caso lo haremos con un ejemplo de instanciación de parámetros.

[2]Las dependencias entre los parámetros no son mostradas debido a su extensión, aunque diversas imágenes en esta tesis muestran secciones donde se reflejan dependencias entre los parámetros.

Tabla C.3: Parámetros del Contexto Base (BC)

	REQUISITOS DE ALTO NIVEL - *HIGH-LEVEL REQUIREMENTS*
QoS	Reliability, Fault Tolerance, Availability
Seguridad	Authentication, Authorization, Confidentiality, Integrity, Trust, Privacy
	PROPIEDADES LOCALES - *LOCAL PROPERTIES*
Recursos	PowerConsumption, Memory, Rayleigh Channel, Energy, ComputationTime
Seguridad	Anti-Tampering, Encryption, Public Key Cryptography, Symmetric Cryptography, Secure Key Exchange, Secure Key redistribution, Key Generation, Signature Scheme, Certificate
	COMUNICACIÓN - *COMMUNICATION*
QoS	Data Rate, Packet Size, Signal Strength, Data Transmission, Transmission Time, Transmission Power
Característica	Time-sleeping, Required-time-on
Consecuencia	Retransmmission
	MEDICIONES - *MEASUREMENTS*
QoS	Throughput, Delay, Jitter, Packet Loss, Response Time, Bit Error Rate (BER)
	ENTORNO - *ENVIRONMENT*
QoS	Allowable Bandwidth, Error Probability
Amenaza / Ataque	DoS, Malicious Devices
Consecuencia	Interference, Congestion, Overhead, Fading, Shadowing, Noise

C.7.1.1. Agregación de un Contexto

Una vez aplicado el GC, podemos aplicar diferentes PCs sobre el CPRM resultante. Este hecho conduce a lo que denominamos *instanciación del modelo paramétrico*. A modo de ejemplo, mostraremos los cambios producidos en el sistema al aplicar el contexto particular mostrado en la Tabla C.4, cuyos pesos son estimaciones acorde al trabajo [1].

La Tabla C.4 muestra los parámetros generales que serán instanciados (Authentication y SignatureScheme) y los parámetros instancia (CAS, DAS, ECDSA, PairingBased). Una vez integrado el nuevo contexto, los parámetros de Authentication y SignatureSheme pasarían a ser niveles, y como tales pueden ser consultados. Esta es una ventaja adicional del modelo, ya que permite comprobar el efecto que este último cambio de contexto tuvo sobre parámetros que ya se encontraban en el modelo. En el nuevo contexto final, cada vez que se incremente el parámetro Authentication o SignatureScheme, también serán incrementados los parámetros instancia, y con ellos los parámetros dependientes de éstos, que han podido introducir nuevas dependencias para hacer el modelo coherente.

Tabla C.4: Pesos w_d conforme [1]

	Dependencia			Peso
Parámetro General	Antecedente	R	Consecuente	w_d
Authentication	CAS	$+$	ECDSA	1
	DAS	$+$	ECDSA	1
	CAS	$\neg c$	Memory	0
	DAS	$\neg c$	Memory	5
	CAS	c	PacketSize	5
	DAS	c	PacketSize	1
	CAS	c	Certificate	2
Signature Scheme	ECDSA	$\neg c$	Energy	1
	PairingBased	$\neg c$	Energy	5
	ECDSA	c	Computation Time	1
	PairingBased	c	ComputationTime	5

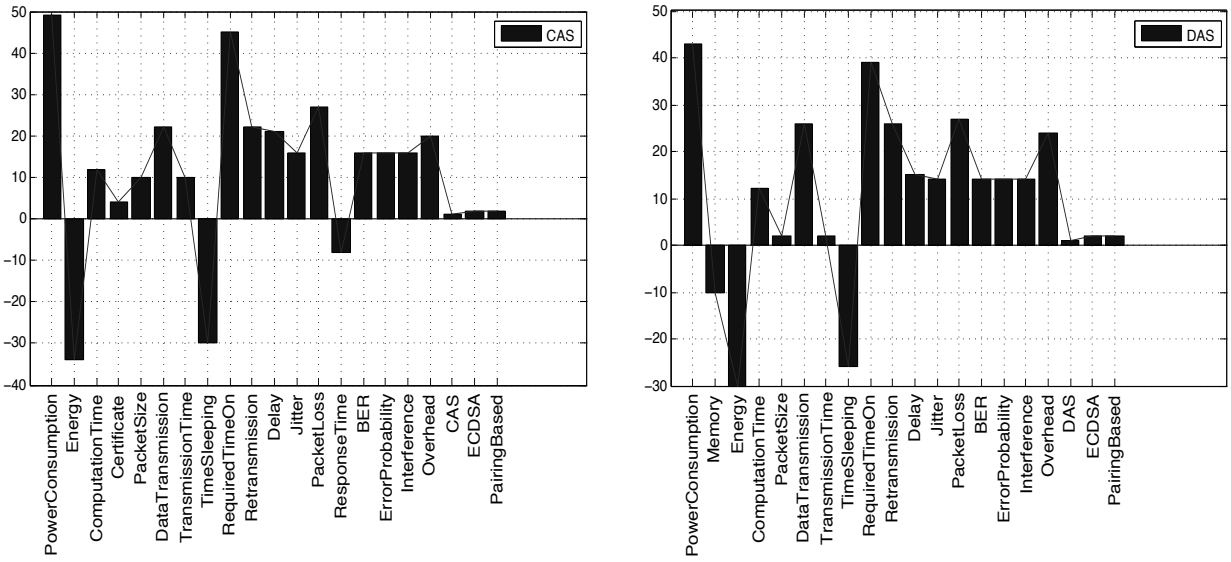


Figura C.8: Impacto de CAS y DAS sobre el Rendimiento.

Finalmente, el proceso de ajuste entre parámetros de Seguridad y QoS, se realiza en base al BC definido y la instanciación del modelo con los mecanismos cuyo impacto en el sistema resultante queremos medir. Por ejemplo, una vez introducido el último contexto (Tabla C.4), podemos evaluar el impacto que los mecanismos de autenticación CAS y DAS tienen sobre los parámetros de rendimiento (Figura C.8) o de cualquier otro tipo. Note que los parámetros sobre los que se percibe el efecto no fueron obtenidos de [1], sino que son resultado de la integración con el BC definido a priori. La información del sistema será mucho más fiable y enriquecedora conforme el número de PC integrados sea mayor.

C.7.2. Caso de uso 2: redes 5G Green

Para este caso de uso, consideramos el despliegue de SQT-RS junto con un generador de esquemas CPRM para redes 5G Green en escenarios *relay*. El selector 5G recibe las características seleccionadas por el usuario y genera esquemas CPRM considerando también la inclusión de los escenarios de *eavesdropping* y *jamming*.

La Figura C.9 muestra el esquema de uso de SQT-RS considerando este último componente desarrollado para facilitar la implementación del caso de uso, y con ello la comprensión del modelo de toma de decisiones propuesto. Además, la ejecución de las reglas devuelve las recomendaciones en formato matricial para que MATLAB pueda interpretarlas y adaptarlas al modelo vigente automáticamente.

Los parámetros empleados para este caso de uso pueden consultarse en las Tablas C.5 y C.6. La Tabla C.5 muestra los parámetros empleados en el contexto base. A diferencia del caso de uso anterior, en este caso de uso aumentamos el número de parámetros que pueden tomar valores de relevancia, ya que precisamente intentamos ilustrar cómo SQT-RS puede ser útil en aquellos casos en los que el número de parámetros implicados dificulta la visualización de los resultados.

La Tabla C.6 muestra los dos contextos considerados. La información específica para la

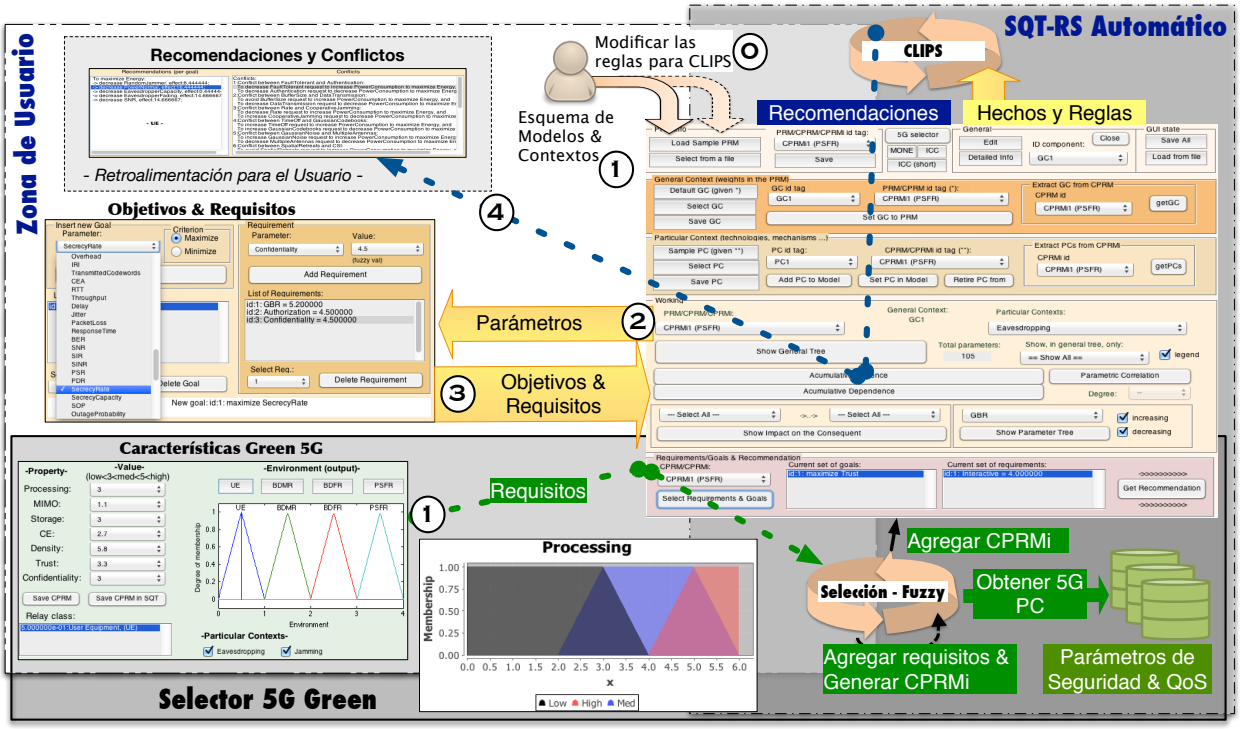


Figura C.9: Despliegue de SQT-RS usando el Selector 5G.

definición de los contextos (relaciones y valores de pesos) se ha obtenido en su mayor parte de los trabajos [128, 129] y [60]. En el caso de los casos de escucha ó *eavesdropping*, nos apoyamos en un trabajo previo en el que se analiza cómo los mecanismos de seguridad a nivel físico podrían ayudar a evitar este tipo de escenarios [62].

El caso de análisis emplea esquemas de contexto instanciados $CPRM_i$ construidos dinámicamente en base a los parámetros escogidos por el usuario empleando un interfaz de usuario y lógica difusa (fuzzy). Los esquemas generados modelan el comportamiento para diferentes tipos de *relay* en escenarios 5G Green, a alto nivel.

Para efectuar el análisis de SQT-RS en escenarios 5G Green escogemos dos parámetros objetivo, que se analizan de forma separada. Por una parte, mostramos los resultados para la selección de SecrecyRate como parámetro objetivo, y también mostramos los resultados para Energy como nuestro segundo parámetro objetivo. En ambos casos, el criterio perseguido es la maximización del valor de efecto de ambos parámetros, en los escenarios donde tienen cabida los siguientes tipos de relay:

- Equipo de Usuario (UE).
- Estación base móvil dependiente de baterías (BDMR).
- Estación base fija dependiente de baterías (BDFR).
- Estación base fija no dependiente de baterías (PSFR).

Antes de la fase de análisis, el entorno con los parámetros es debidamente probado hasta que el comportamiento se asemeja a un entorno en 5G Green, y los valores obtenidos para los parámetros del contexto base son coherentes. Por ejemplo, al incrementar la transmisión, se espera que la energía se vea reducida.

Tabla C.5: Parameters for a Base Context in 5G Green environments

REQUISITOS DE ALTO NIVEL - *HIGH-LEVEL REQUIREMENTS*	
Recurso	Guaranteed bit rate (GBR), non GBR (N-GBR).
Seguridad	Authentication, authorisation, accounting, confidentiality, integrity, non repudiation, trust, privacy.
QoE	Conversational, Interactive, Streaming, Background, User Experience (UE=6, BDMR=4, BDFR=0, PSFR=0).
Característica	Complexity, fault tolerant (BDFR=4, PSFR=6), availability (BDFR=4, PSFR=6), reliability.
PROPIEDADES LOCALES - *LOCAL PROPERTIES*	
Recurso	Battery, Memory, Processing, Storage.
Rendimiento	Node lifetime, power consumption.
Seguridad	Anti-tampering, signature, encryption, Asymmetric Cryptography (AC), Symmetric Cryptography (SC), key generation, Reputation.
Característica	Mobility(UE=4, BDMR=6, BDFR=0, PSFR=0), relay class.
Amenaza	Misbehaviour (UE=3).
COMUNICACIÓN - *COMMUNICATION*	
Recurso	Available time-slots, buffer size.
Rendimiento	Packet size, signal strength, data transmission, transmission time, transmission power (PSFR=4), reception power, time on, time off, transmission capacity (PSFR=4), rate.
Seguridad	collaborative jamming, gaussian codebooks.
Característica	Multiple antennas, MIMO, successive relaying, half-duplex (HD), full-duplex (FD), decode and forward (DF), amplify-and-forward (AF), channel surfing, spatial retreats, Channel State Information (CSI) (BDFR=3, PSFR=6) availability, multimode.
Consecuencia	Retransmission, congestion, overhead, inter-relay interference (IRI).
MEDICIONES - *MEASUREMENTS*	
Rendimiento	Max rate, min rate, transmitted codewords, Channel Estimation Accuracy (CE) , RTT, throughput, delay, jitter, packet loss, response time, bit-error rate (BER), Signal-to-noise ratio (SNR), Signal-to-interference ratio (SIR), Signal-to-interference-plus-noise ratio (SINR), packet sent ratio (PSR), packet delivery ratio (PDR).
Seguridad	Secrecy rate, secrecy capability, secrecy outage probability (SOP).
Consecuencia	Outage probability.
ENTORNO - *ENVIRONMENT*	
Característica	Density, participants, diversity, noise, channel symmetry, network lifetime, multi-path fading, eavesdropper fading, Handover.
Consecuencia	Error probability.
Amenaza	Denial of Service (DoS), Eavesdropping, Jamming.

Se consideran las recomendaciones y los conflictos para escenarios de 5G Green combinados con los casos de Eavesdropping y Jamming, que son tomados como contextos particulares

Tabla C.6: Pesos w_d para las Relaciones en los PCs

Contexto	Dependencia			Peso
	Antecedente	Relación	Consecuente	w_d
Eavesdropping (E)	EavesdropperFading	c	SecrecyRate	1
	EavesdroppingFading	nc	Eavesdropping	1
	EavesdroppingCapacity	c	Eavesdropping	1
	EavesdropperCapacity	$\neg c$	SecrecyCapacity	1
Jamming (J)	ConstantJammer	+	DoS	4
	ConstantJammer	$\neg c$	PDR	3
	DeceptiveJammer	+	DoS	4
	DeceptiveJammer	+	TimeOn	4
	RandomJammer	+	DoS	2
	ReactiveJammer	+	DoS	3
	ReactiveJammer	+	TimeOn	3
	ReactiveJammer	i-	PSR	0
	ReactiveJammer	$\neg c$	PDR	5
	Jamming	c	PowerJamming	1
	PowerJamming	$\neg c$	SecrecyCapacity	1
	PowerJamming	$\neg c$	TransmissionCapacity	3

a integrar a petición del usuario.

C.7.2.1. Aportaciones y Evaluación de SQT-RS

Este caso de uso resultó interesante no sólo por la evaluación del componente SQT-RS, sino también porque se plantean directrices básicas para la composición de un modelo complejo, como es el caso del contexto base empleado para generar los $CPRM$ correspondientes a los casos UE, BDMR, BDFR y PSFR.

C.7.2.1.1. Anomalías. Una de las directrices básicas es la detección de anomalías en el comportamiento paramétrico. Éstas pueden ser identificadas cuando un comportamiento conocido entre parámetros no se da en nuestro modelo. Por ejemplo, que un incremento de casos de *jamming* mejore el parámetro SecrecyRate. Esto no es coherente, porque los casos de jamming pueden degradar tanto las comunciaciones que acaben aislando nodos de la red, y entendemos que en esos casos sin comunicación el parámetro SecrecyRate carece de sentido. Por ello, cuando se detectan éstos casos conviene identificar el foco del problema, que suele estar en que faltan o sobran relaciones de dependencia, o en un ajuste necesario en los pesos. Por ejemplo, otra anomalía que pudiera producirse, es que se detecte que parámetros que habitualmente tienen un gran impacto sobre la energía no afecten más que otros parámetros que no afectan en demasía. Este tipo de anomalías se solventa con la modificación de los pesos de relevancia, y los pesos de las relaciones entre parámetros para casos específicos.

C.7.2.1.2. Recomendaciones. Las recomendaciones mostradas por SQT-RS corresponden con lo esperado dada la definición de los modelos. Para el caso de la maximización del parámetro SecrecyRate (tasa de secreto) las ventajas del esquema para PSFR permiten que las mejoras de los parámetros aporten mayores beneficios que el resto de los esquemas. Los resultados para los casos UE y BDMR son muy similares, ya que los relay UE podrían entrar en la clasificación de BDMR. Sería distinto si consideráramos el caso de incluir aplicaciones

o mecanismos específicos para cada entorno, pero esto requiere un análisis y conocimiento mucho más profundo de cada uno de estos campos y escapa al ámbito de esta tesis, que es más general. En cuanto al caso BDFR, el efecto de las recomendaciones superaría al de los casos de los relays móviles pero los mejores resultados siguen siendo para PSFR. Este comportamiento entendemos que es razonable. En el caso de la maximización del parámetro Energy (energía), como contamos con más parámetros que afectan a este recurso, vamos a encontrarnos con diversos conflictos que son indicados por el interfaz de SQT-RS.

Cabe destacar, que aunque PSFR no es dependiente de la energía, en el análisis no suprimimos la relevancia del parámetro Energy para PSFR, porque al ser un estudio basado en Green, entendemos que el consumo energético sigue siendo relevante, no sólo por el tiempo de vida de un nodo, sino, como en el caso de PSFR, por el coste que ello supone y la huella energética.

C.8. Conclusiones

El principal objetivo de esta tesis es diseñar mecanismsmos para el desarrollo de sistemas seguros con calidad de servicio (QoS). Bajo esta premisa, en esta tesis presentamos una herramienta para el análisis de la compensación entre parámetros de Seguridad y QoS. La herramienta, denotada SQT por sus siglas en inglés *Security and QoS Tradeoff Tool*, fue definida e implementada en MATLAB, y su función es la gestión de componentes definidos conforme al modelo de relaciones paramétricas basado en el contexto o CPRM conforme sus siglas en inglés *Context-based Parametric Relationship Model*.

El modelo CPRM es propuesto en esta tesis para permitir la definición de un entorno ó sistema pueda ser definido en función de sus parámetros y relaciones paramétricas con un alto grado de abstracción. Los puntos que motivan nuestra decisión para la definición y el uso de éste nuevo modelo propuesto son, de forma resumida, los siguientes:

- Los entornos de la Internet del Futuro presentan parámetros de seguridad y QoS genéricos (propiedades) y específicos (mecanismos), con distinto grado de dinamismo. Por lo tanto, los mecanismos que se desarrollen para el asesoramiento de la seguridad y la QoS tienen que ser capaces de manejar diferentes tipos de parámetros. Más aún, tienen que continuar siendo útiles aún si los parámetros cambian.
- Los parámetros de seguridad y QoS se relacionan a través de diferentes tipos de parámetros. Es importante considerar en el análisis aquellos parámetros intermedios que son afectados por los parámetros de seguridad y que a su vez afectan a parámetros de QoS y viceversa.
- Las relaciones entre parámetros de seguridad y QoS se dan a distintos niveles de abstracción. Por lo que los parámetros de seguridad y QoS deben clasificarse atendiendo a dichos niveles para un análisis conjunto, y estos niveles pueden cambiar en cualquier momento, dependiendo de los objetivos o criterios de evaluación.
- La seguridad y la QoS son criterios subjetivos. Cuando el usuario forma parte del sistema, el valor o la relevancia de los parámetros puede variar de forma sustancial muy rápidamente. Por ejemplo, la seguridad carece de relevancia para un usuario si dificulta el uso normal del sistema, y la QoS también puede perder su relevancia si,

por ejemplo, el usuario percibe que su seguridad se ve amenazada en un momento dado. Por ello considerar valores de subjetividad en los esquemas para determinar la relevancia de los parámetros en un momento dado es crítico para entender el contexto en el que las dependencias tienen cabida.

La definición y posterior implementación de CPRM considera tales características, evaluando la compensación de la seguridad y la QoS en entornos heterogéneos dinámicamente, a nivel abstracto. Además, en el diseño los valores subjetivos para los parámetros y demás elementos del modelo son considerados.

Así como CPRM define el lenguaje para interpretar la composición de entornos basados en relaciones paramétricas, la herramienta SQT implementa las funciones definidas por el modelo CPRM y proporciona una prueba de concepto de esta solución que se basa en el conocimiento introducido sobre el entorno.

De hecho, la definición de dicho conocimiento, dado en función de las relaciones paramétricas, es una parte fundamental de este trabajo, y bastante compleja de realizar. Las consideraciones finales de diseño realizadas en el segundo caso de uso propuesto en el Capítulo 6 son sólo una muestra de dicho esfuerzo. Definir correctamente las relaciones paramétricas significa modelar adecuadamente el comportamiento de los sistemas usados en nuestro enfoque.

Gestionar los diferentes parámetros e implementar el proceso de instanciación definido en el modelo CPRM es realmente complejo. Sin embargo, una vez que dicho comportamiento fue definido y probado alcanzando una fase de estabilidad donde las anomalías son suprimidas, es posible realizar diferentes tests y ampliar el modelo usando un conjunto de parámetros de seguridad y QoS (e intermedios) bien definidos y debidamente probados.

Además, en esta tesis definimos e integramos en SQT un sistema de recomendaciones, denotado como SQT-RS, por las siglas en inglés *SQT - Recommendation System*. Dicho módulo permite extraer información dinámicamente de entornos expresados conforme al modelo CPRM e importados en SQT, e inferir información sobre las modificaciones a realizar en los parámetros del sistema para lograr los diferentes objetivos. SQT-RS es implementado usando MATLAB (para ser integrado en SQT), Java (para la construcción del .jar), y Jess para operar con el fichero escrito en CLIPS (.clp) desde Java e inferir los resultados conforme a los hechos dinámicos extraídos del modelo seleccionado. El fichero .clp puede ser modificado para mejorar la visualización de resultados y SQT-RS seguirá funcionando, si los principios de diseño de SQT-RS se mantienen. Es decir, si proporcionamos las salidas esperadas por SQT-RS.

Este módulo final para SQT fue propuesto para simplificar el uso de SQT, porque la representación visual de resultados proporcionada es difícil de entender para usuarios no experimentados en las relaciones paramétricas entre parámetros de seguridad y QoS. Además, SQT fue concebido para trabajar sobre la premisa de que diferentes usuarios puedan aportar información variada al modelo sobre su propio campo de investigación, por lo que diferentes usuarios se pueden encontrar con parámetros que no entienden pero que, finalmente, afectan a sus parámetros por la cadena de dependencias.

Finalmente, en cuanto a la aplicabilidad de nuestra propuesta para escenarios de la Internet del Futuro, consideramos que los dos casos de uso considerados se ajustan bastante bien a nuestro enfoque. Ambos casos cubren parámetros a distinto nivel de abstracción, la

presencia de dispositivos restringidos en recursos, y consideran la compensación de seguridad y QoS con los parámetros o propidades de más alto nivel. Estos casos de uso fueron seleccionados tras un análisis de parámetros de relevancia en las redes candidatas a formar parte de la Internet del Futuro precisamente para garantizar que los casos de uso pudiesen resultar aclaradores sobre la usabilidad de la propuesta.

Precisamente, es un objetivo primordial que esta propuesta pueda ser útil en el futuro. En esta tesis el código de la herramienta implementada no se adjunta por su extensión, pero propondremos mecanismos adecuados para su difusión para que esté a disposición de la comunidad científica. Esto ha de ser así, dado que entendemos que esta solución puede ser útil para abordar los desafíos abiertos expuestos a continuación, que pueden dar lugar a nuevas líneas de investigación relacionadas.

C.8.1. Desafíos abiertos

En esta tesis nos hemos centrado en el problema de la evaluación del impacto entre mecanismos de seguridad y QoS, en cómo realizar esta evaluación y en definir aquellos criterios que consideramos debiera cubrir un mecanismo desarrollado a tal fin. Además, consideramos fundamental la opinión del usuario, y adaptar los mecanismos para fomentar su participación en la toma de decisiones.

Sin embargo, transversalmente a estas cuestiones, hay otros desafíos abiertos en los que no hemos trabajado, ya que, desde nuestra perspectiva, suponen nuevas líneas de investigación, demasiadas extensas para que tengan cabida en esta tesis.

C.8.1.1. Adquisición automática de datos y clasificación

Una parte fundamental de esta tesis es precisamente trabajar con información sobre parámetros y relaciones paramétricas. Esto ha supuesto un esfuerzo sustancial, pero la idea sería que pudiese ser automatizable, y pensamos que es factible.

La convergencia de redes heterogéneas genera grandes volúmenes de datos: información sobre preferencias de usuarios, rendimiento de redes coforme al volúmen de tráfico puntual por horas o bien los dispositivos que las forman, relación de intentos de ataque, por zonas o también clasificación de tipos de ataque y en algunos casos incluso los dispositivos desde los que se intentó el ataque. Toda esta información, está presente en entornos masivos, y, cuando existen elementos centrales donde estos datos se recolectan, esta información puede ser clasificada, estudiada y manejada para extraer información muy útil para un estudio de compensación de seguridad y QoS. Una muestra de entornos de este tipo se encuentra en nuestro segundo caso de uso, en redes 5G, considerado en el Capítulo 6.

Además, si esta información pudiese ser compartida garantizando el anonimato y la privacidad de las fuentes y los individuos, diferentes sistemas podrían beneficiarse sobre la información que el impacto de configuraciones ha tenido sobre los dispositivos de la red, y cuáles han sido más afectados, o cuáles de estos dispositivos suelen ser usados con más probabilidad para efectuar ataques.

Existen diversos problemas a solventar para ello. Este es un debate que tomamos de forma muy transversal en el Capítulo 2, sobre la colaboración de diferentes organismos para establecer bases de información comunes. Es un estudio que puede llevar a una línea

nueva sobre extracción y catalogación de información y su transformación en relaciones paramétricas para su análisis automático. Las ventajas pueden ser enormes, y una de ellas está en la identificación de efectos en cascada producidos porque paarámetros relevantes en determinadas redes queden desatendidos. O bien porque los parámetros entendidos como relevantes dejen de serlo momentáneamente.

C.8.1.2. Seguridad y QoS empotrada en el dispositivo

Nuestro enfoque, en su estado actual, se basa en la información proporcionada por el modelo. Por lo tanto, al ser un modelo basado en el conocimiento, los resultados son más específicos y concisos conforme hay más información definida en el esquema usado en la evaluación. Uno de los problemas de este enfoque, es que esto también incrementa el número de hechos a ser procesados por SQT-RS e icnrementa el cómputo y la memoria empleados por la solución, limitando considerablemente su aplicación en sistemas de timpo real.

De igual forma, esta solución y similares no serían viables para su integración en dispositivos restringidos en recursos para que éstos tomasen sus propias decisiones sobre la compensación de parámetros de QoS y seguridad. No sólo porque no es una solución ligera para estos dispositivos, sino también porque éstos suelen ser diseñados para tomar decisiones basándose en valores más cercanos a la red, menos abstractos.

Sin embargo, sí consideramos que el esquema propuesto podría ser adaptado para su uso en estos dispositivos, aunque limitando ó eliminando gran parte de la funcionalidad, que, por otra parte, puede no ser necesaria en tales dispositivos para un razonamiento autónomo. Por ejemplo, las recomendaciones en lenguaje de usuario no sería útil para estos dispositivos, y sí las recomendaciones automáticas devueltas por el intérprete. Habría que modificar también el lenguaje de implementación de la herramienta y otros factores.

De cualquier forma, entendemos que un campo abierto interesante para futuros estudios es precisamente cómo dotar a los dispositivos más restringidos de este razonamiento autónomo para decidir compensaciones entre seguridad y QoS. Algunos pasos en esta dirección se toman en el Capítulo 2, a través de la definición de los componentes de un nodo (Figura 2.3). Sin embargo, el coste de implementar esta solución y el tiempo de vida del nodo considerando tales componentes no ha sido evaluado aún, y nos quedamos en el plano teórico.

Además, esto abre la puerta a discusiones bastante interesantes sobre cómo prolongar la usabilidad de soluciones basadas en hardware que están optimizadas para un propósito (como el caso de los sensores), combinándolas con soluciones software basadas en la actualización automática, sin incrementar el coste del dispositivo final. Uno de los problemas de los componentes de seguridad es que limitan severamente el tiempo de vida de los dispositivos, no sólo por el consumo computacional, que puede ser paliado en mayor o menor medida, sino en cuanto a tiempo de vida por usabilidad. Los mecanismos de seguridad requieren de actualizaciones periódicas, y cómo proporcionar las medidas de seguridad sin afectar a la disponibilidad de sensores en entornos industriales es crucial para la adopción de este tipo de soluciones.

C.8.1.3. Sistema de recomendaciones de grano fino

En el Capítulo 5, se definió un sistema de recomendaciones para la herramienta SQT, llamado SQT-RS, que fue evaluado en el Capítulo 6 con un caso de uso. El objetivo perseguido en ese punto es definir las directrices básicas para extraer información de los sistemas definidos conforme al modelo CPRM. Aunque tenemos una evaluación factible que muestra la usabilidad de la solución propuesta, consideramos que SQT-RS tiene un potencial mucho mayor que el desarrollado en esta tesis.

Conforme más información se tnega del uso de sistemas basados en CPRM, las reglas empleadas para inferir información podrían ser mejoradas para permitir recomendaciones de grano fino sobre un sistema en particular.

Cabe destacar, que este desarrollo futuro sólo sería posible si el primer desafío propuesto (Sección C.8.1.1) es cubierto adecuadamente. Además, consideramos imprescindible proporcionar mecanismos para compartir la información gestionada por los sistemas cooperadores para el análisis de la compensación entre seguridad y QoS. Proveer de mecanismos de seguridad adecuados es imprescindible para este proceso, a fin de definir cómo los datos pueden ser compartidos, qué información debe ser compartida y cómo es transformada dicha información para proporcionar información útil para el análisis paramétrico sin comprometer las fuentes de la información. Preservar el anonimato, la privacidad, la confidencialidad y otras propiedades críticas debe ser un requisito fundamental.

Bibliography

[1] Rehana Yasmin, Eike Ritter, and Guilin Wang. An authentication framework for wireless sensor networks using identity-based signatures. In *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*, pages 882–889. IEEE, 2010.

[2] Jorge Carapinha, Roland Bless, Christoph Werle, Konstantin Miller, Virgil Dobrota, Andrei Bogdan Rus, Heidrun Grob-Lipski, and Horst Roessler. Quality of service in the future internet. In *Kaleidoscope: Beyond the Internet?-Innovations for Future Networks and Services, 2010 ITU-T*, pages 1–8. IEEE, 2010.

[3] Aristi Galani, Nikos Koutsouris, George Poulios, Kostas Tsagkaris, and Panagiotis Demestichas. Designing the governance framework for network and service management in future networks. In *Future Network and Mobile Summit (FutureNetworkSummit), 2013*, pages 1–8. IEEE, 2013.

[4] Eleni Patouni, Andreas Merentitis, Panagiotis Panagiotopoulos, Aristotelis Glentis, and Nancy Alonistioti. Network virtualisation trends: Virtually anything is possible by connecting the unconnected. In *Future Networks and Services (SDN4FNS), 2013 IEEE SDN for*, pages 1–7. IEEE, 2013.

[5] Mohsen Nader Tehrani, Murat Uysal, and Halim Yanikomeroglu. Device-to-device communication in 5g cellular networks: challenges, solutions, and future directions. *Communications Magazine, IEEE*, 52(5):86–92, 2014.

[6] Pedro Alipio, Solange Rito Lima, and Paulo Carvalho. A unified metric for quality of service quantification. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, page 84. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009.

[7] Avesh K. Agarwal and Wenye Wang. On the impact of quality of protection in wireless local area networks with ip mobility. *Mobile Networks and Applications*, 12(1):93–110, 2007. ISSN 1383-469X. doi: http://dx.doi.org/10.1007/s11036-006-0009-6.

[8] X. Fu, D. Hogrefe, S. Narayanan, and R. Soltwisch. Qos and security in 4g networks. In *First Annual Global Mobile Congress, Shanghai, China*, 2004.

[9] S. Mohan and N. Agarwal. A convergent framework for qos-driven social media content delivery over mobile networks. In *Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE), 2011 2nd International Conference on*, pages 1–7. IEEE, 2011.

[10] Mario Marchese. *QoS over heterogeneous networks*. John Wiley & Sons, 2007.

[11] Bin Lu and Udo W Pooch. Security in qos signaling systems for mobile ad hoc networks. In *Wireless And Mobile Computing, Networking And Communications, 2005.(WiMob'2005), IEEE International Conference on*, volume 3, pages 213–220. IEEE, 2005.

[12] Amitabh Mishra. *Security and quality of service in ad hoc wireless networks*. Cambridge University Press, 2008.

[13] F R. Yu, H. Tang, S. Bu, and D. Zheng. Security and quality of service (qos) co-design in cooperative mobile ad hoc networks. *EURASIP Journal on Wireless Communications and Networking*, 2013(1):1–14, 2013.

[14] Tarik Taleb, Yassine Hadjadj Aoul, and Abderrahim Benslimane. Integrating security with qos in next generation networks. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–5. IEEE, 2010.

[15] R. Roman and J. Lopez. Integrating wireless sensor networks and the internet: a security analysis. *Internet Research*, 19(2):246–259, 2009.

[16] Hala Mostafa, Nathaniel Soule, Nicholas Hoff, Partha Pal, and Patrick Hurley. Applying distributed optimization for qos-security tradeoff in a distributed information system. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 1261–1262. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

[17] D. Chen and P.K. Varshney. Qos support in wireless sensor networks: A survey. In *International Conference on Wireless Networks*, volume 233, 2004.

[18] Delphine Christin, Parag S. Mogre, and Matthias Hollick. Survey on wireless sensor network technologies for industrial automation: The security and quality of service perspectives. *Future Internet*, 2(2):96–125, 2010. ISSN 1999-5903. doi: 10.3390/fi2020096.

[19] Chunxiao Chigan, Yinghua Ye, and Leiyuan Li. Balancing security against performance in wireless ad hoc and sensor networks. In *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, volume 7, pages 4735–4739. IEEE, 2004.

[20] J.H. Cho and R. Chen. On design tradeoffs between security and performance in wireless group communicating systems. In *Secure Network Protocols, 2005.(NPSec). 1st IEEE ICNP Workshop on*, pages 13–18. IEEE, 2005.

[21] Kyoung-Don Kang and Sang H Son. Towards security and qos optimization in real-time embedded systems. *ACM SIGBED Review*, 3(1):29–34, 2006.

[22] Jianyong Chen, Cunying Hu, Huawang Zeng, and Jun Zhang. Impact of security on qos in communication network. In *Networks Security, Wireless Communications and Trusted Computing, 2009. NSWCTC'09. International Conference on*, volume 2, pages 40–43. IEEE, 2009.

[23] Fabio Martinelli and Ilaria Matteucci. Partial model checking for the verification and synthesis of secure service compositions. In *Public Key Infrastructures, Services and Applications*, pages 1–11. Springer, 2014.

[24] Fatih Karatas, Lars Fischer, and Dogan Kesdogan. Service composition with consideration of interdependent security objectives. *Science of Computer Programming*, 97: 183–201, 2015.

[25] Seungwon Shin, Phillip A Porras, Vinod Yegneswaran, Martin W Fong, Guofei Gu, and Mabry Tyson. Fresco: Modular composable security services for software-defined networks. In *NDSS*, 2013.

[26] Markus Tasch, Rahamatullah Khondoker, Ronald Marx, and Kpatcha Bayarou. Security analysis of security applications for software defined networks. In *Proceedings of the AINTEC 2014 on Asian Internet Engineering Conference*, page 23. ACM, 2014.

[27] Yu-Jia Chen, Feng-Yi Lin, and Li-Chun Wang. Dynamic security traversal in openflow networks with qos guarantee. *International Journal of Science and Engineering*, 4: 251–256, 2014.

[28] R Di Pietro, S Guarino, NV Verde, and J Domingo-Ferrer. Security in wireless ad-hoc networks–a survey. *Computer Communications*, 51:1–20, 2014.

[29] Majid I Khan, Wilfried N Gansterer, and Guenter Haring. Static vs. mobile sink: The influence of basic parameters on energy efficiency in wireless sensor networks. *Computer communications*, 36(9):965–978, 2013.

[30] Mourad Alia, Marc Lacoste, Ruan He, and Frank Eliassen. Putting together qos and security in autonomic pervasive systems. In *Proceedings of the 6th ACM workshop on QoS and security for wireless and mobile networks*, pages 19–28. ACM, 2010.

[31] Gene Tsudik. Some issues in wsn, manet and cellular security (position paper). *Proceedings of the ARO Planning Workshop on Embedded Systems and Network Security Held in Raeligh, North Carolina*, 2007.

[32] Pablo Najera, Rodrigo Roman, and Javier Lopez. User-centric secure integration of personal rfid tags and sensor networks. *Security and Communication Networks*, 6(10): 1177–1197, 2013.

[33] Geoff Weave Paul Schmit. Mipv6: New capabilities for seamless roaming among wired, wireless, and cellular network. Technical report, Intel, Developer UPDATE Magazine, 2002.

[34] K. Sohraby, D. Minoli, and T.F. Znati. *Wireless sensor networks: technology, protocols, and applications*. Wiley-Blackwell, 2007.

[35] Kemal Akkaya and Mohamed Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(3):325 – 349, 2005. ISSN 1570-8705. doi: DOI:10.1016/j.adhoc.2003.09.010.

[36] M. Noori and M. Ardakani. Characterizing the traffic distribution in linear wireless sensor networks. *Communications Letters, IEEE*, 12(8):554 –556, aug. 2008. ISSN 1089-7798. doi: 10.1109/LCOMM.2008.080488.

[37] A.V. Taddeo and A. Ferrante. Run-time selection of security algorithms for networked devices. In *Proceedings of the 5th ACM symposium on QoS and security for wireless and mobile networks*, pages 92–96. ACM, 2009.

[38] Ruben Rios and Javier Lopez. (un)suitability of anonymous communication systems to wsn. *IEEE Systems Journal*, PP(99):1–13, 2012. ISSN 1932-8184. doi: 10.1109/JSYST.2012.2221956.

[39] JosephK. Liu, Joonsang Baek, Jianying Zhou, Yanjiang Yang, and JunWen Wong. Efficient online/offline identity-based signature for wireless sensor network. *International Journal of Information Security*, 9(4):287–296, 2010. ISSN 1615-5262. doi: 10.1007/s10207-010-0109-y.

[40] I. Dietrich and F. Dressler. On the lifetime of wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 5(1):5, 2009.

[41] T. Zahariadis, H.C. Leligou, S. Voliotis, S. Maniatis, P. Trakadas, and P. Karkazis. Energy-aware secure routing for large wireless sensor networks. *WSEAS TRANSACTIONS on COMMUNICATIONS*, 8(9):981–991, 2009.

[42] J. Lopez, R. Roman, I. Agudo, and C. Fernandez-Gago. Trust management systems for wireless sensor networks: Best practices. *Computer Communications*, 33(9):1086–1093, 2010.

[43] Ana Nieto and Javier Lopez. Traffic classifier for heterogeneous and cooperative routing through wireless sensor networks. In *Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on*, pages 607–612. IEEE, 2012.

[44] Tamara Pazynyuk, JianZhong Li, George S. Oreku, and LiQiang Pan. Qos as means of providing wsns security. *International Conference on Networking*, 0:66–71, 2008. doi: http://doi.ieeecomputersociety.org/10.1109/ICN.2008.22.

[45] G. Bella. The principle of guarantee availability for security protocol analysis. *International Journal of Information Security*, 9:83–97, 2010. ISSN 1615-5262.

[46] AS Nargunam and MP Sebastian. Self-organized qos aware multicast routing scheme for ad hoc networks. *International Journal of Computers and Applications 2010*, 32 (2), 2010.

[47] Utz Roedig and Cormac J. Sreenan. Predictable and controllable wireless sensor networks. In *Proceedings of the Information Technology & Telecommunications Conference (IT&T2005), Cork, Ireland*, oct 2005.

[48] J.Z. Sun. Qos parameterization algorithm in data collection for wireless sensor networks. In *Proceedings of the 5th ACM symposium on QoS and security for wireless and mobile networks*, pages 57–64. ACM, 2009.

[49] M.H. Yaghmaee and D. Adjeroh. A model for differentiated service support in wireless multimedia sensor networks. In *Computer Communications and Networks, 2008. ICCCN'08. Proceedings of 17th International Conference on*, pages 1–6. IEEE, 2008.

[50] A. Hegland and E. Winjum. Securing qos signaling in ip-based military ad hoc networks. *Communications Magazine, IEEE*, 46(11):42 –48, november 2008. ISSN 0163-6804. doi: 10.1109/MCOM.2008.4689243.

[51] C. Zouridaki, M. Hejmo, B.L. Mark, R.K. Thomas, and K. Gaj. Analysis of attacks and defense mechanisms for qos signaling protocols in manets. In *Proc. WIS Workshop*, pages 61–70, 2005.

[52] Bin Lu. *Quality of Service (QoS) security in mobile ad hoc networks*. PhD thesis, Texas A&M University, 2005.

[53] M. Hejmo, B.L. Mark, C. Zouridaki, and R.K. Thomas. Design and analysis of a denial-of-service-resistant quality-of-service signaling protocol for manets. *Vehicular Technology, IEEE Transactions on*, 55(3):743–751, 2006.

[54] E.A. Panaousis, C. Politis, K. Birkos, C. Papageorgiou, and T. Dagiuklas. Security model for emergency real-time communications in autonomous networks. *Information Systems Frontiers Journal, Special issue on ubiquitous multimedia services*, 2010.

[55] Farooq Khan. *LTE for 4G Mobile Broadband: Air Interface Technologies and Performance*. Cambridge University Press, 2009. ISBN 9780521882217.

[56] Prodromos Makris, Dimitrios N Skoutas, and Charalabos Skianis. A survey on context-aware mobile and wireless networking: On networking and computing environments' integration. *Communications Surveys & Tutorials, IEEE*, 15(1):362–386, 2013.

[57] Hatem Abou-Zeid and HOSSAM S Hassanein. Toward green media delivery: location-aware opportunities and approaches. *Wireless Communications, IEEE*, 21(4):38–46, 2014.

[58] Yongsuk Park and Taejoon Park. A survey of security threats on 4g networks. In *Globecom Workshops, 2007 IEEE*, pages 1 –6, nov. 2007. doi: 10.1109/GLOCOMW.2007.4437813.

[59] R. Shankar and P. Dananjayan. Security enhancement with optimal qos using eap-aka in hybrid coupled 3g-wlan convergence network. *Arxiv preprint arXiv:1007.5165*, 2010.

[60] Wenyuan Xu, Ke Ma, Wade Trappe, and Yanyong Zhang. Jamming sensor networks: attack and defense strategies. *Network, IEEE*, 20(3):41–47, 2006.

[61] I Chih-Lin, Corbett Rowell, Shuangfeng Han, Zhikun Xu, Gang Li, and Zhengang Pan. Toward green and soft: a 5g perspective. *IEEE Communications Magazine*, 52 (2):66–73, 2014.

[62] N. Nomikos, A. Nieto, P. Makris, D.N. Skoutas, D. Vouyioukas, P. Rizomiliotis, J. Lopez, and C. Skianis. Relay selection for secure 5g green communications. *Springer Telecommunication Systems*, pages 1–35, 2014.

[63] Victor CM Leung, Tarik Taleb, Min Chen, Thomas Magedanz, L-C Wang, and Rahim Tafazolli. Unveiling 5g wireless networks: emerging research advances, prospects, and challenges [guest editorial]. *Network, IEEE*, 28(6):3–5, 2014.

[64] Magnus Olsson, Cicek Cavdar, Pål Frenger, Sibel Tombaz, Dario Sabella, and R Jantti. 5green: Towards green 5g mobile networks. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2013 IEEE 9th International Conference on*, pages 212–216. IEEE, 2013.

[65] Ali Imran, Ahmed Zoha, and Adnan Abu-Dayya. Challenges in 5g: how to empower son with big data for enabling 5g. *Network, IEEE*, 28(6):27–33, 2014.

[66] Xi Zhang, Wenchi Cheng, and Hailin Zhang. Heterogeneous statistical qos provisioning over 5g mobile wireless networks. *Network, IEEE*, 28(6):46–53, 2014.

[67] Nicola Cordeschi, Mohammad Shojafar, Danilo Amendola, and Enzo Baccarelli. Energy-efficient adaptive networked datacenters for the qos support of real-time applications. *The Journal of Supercomputing*, pages 1–31, 2014.

[68] R Guerzoni, R Trivisonno, and D Soldani. Sdn-based architecture and procedures for 5g networks. In *5G for Ubiquitous Connectivity (5GU), 2014 1st International Conference on*, pages 209–214. IEEE, 2014.

[69] L. Salgarelli, M. Buddhikot, J. Garay, S. Patel, and S. Miller. Efficient authentication and key distribution in wireless ip networks. *Wireless Communications, IEEE*, 10(6): 52–61, 2003.

[70] J. Al-Muhtadi, D. Mickunas, and R. Campbell. A lightweight reconfigurable security mechanism for 3g/4g mobile devices. *Wireless Communications, IEEE*, 9(2):60 – 65, april 2002. ISSN 1536-1284. doi: 10.1109/MWC.2002.998526.

[71] R. Muraleedharan and L.A. Osadciw. Increasing qos and security in 4g networks using cognitive intelligence. In *Globecom Workshops, 2007 IEEE*, pages 1–6. IEEE, 2007.

[72] N.L. Clarke and S.M. Furnell. Authenticating mobile phone users using keystroke analysis. *International Journal of Information Security*, 6(1):1–14, 2007. ISSN 1615-5262. doi: 10.1007/s10207-006-0006-6.

[73] Joel JPC Rodrigues and Paulo ACS Neves. A survey on ip-based wireless sensor network solutions. *International Journal of Communication Systems*, 23(8):963–981, 2010.

[74] A. Prasad, N. Prasad, and Inc ebrary. *802.11 WLANs and IP networking: security, QoS, and mobility*. Artech House, 2005.

[75] Vivek Gupta. Ieee p802.21 tutorial. Technical report, Institute of Electrical and Electronics Engineers (IEEE), 2011.

[76] X Fu, T Chen, A Festag, H Karl, G Schäfer, and C Fan. Secure qos-enabled mobility support for ip-based networks. In *IN PROC. IP BASED CELLULAR NETWORK CONFERENCE (IPCN*. Citeseer, 2003.

[77] Ismail Saadat, Fábio Buiati, Delfín Rupérez Cañas, and Luis Javier García Villalba. Overview of ieee 802.21 security issues for mih networks. In *ICIT 2011: Proceedings of the 5th International Conference on Information Technology*, 2011.

[78] Y Ohba, M Meylemans, and S Das. Media-independent handover security tutorial. Technical report, IEEE 802.21/Media Independent Handover Working Group, 2008.

[79] A. Pontes, D. dos Passos Silva, J. Jailton, O. Rodrigues, and K.L. Dias. Handover management in integrated wlan and mobile wimax networks. *Wireless Communications, IEEE*, 15(5):86 –95, october 2008. ISSN 1536-1284. doi: 10.1109/MWC.2008.4653137.

[80] G. Lampropoulos, C. Skianis, and P. Neves. Optimized fusion of heterogeneous wireless networks based on media-independent handover operations [accepted from open call]. *Wireless Communications, IEEE*, 17(4):78 –87, august 2010. ISSN 1536-1284. doi: 10.1109/MWC.2010.5547925.

[81] William Stallings. *Network security Essentials: Applications and Standars*. Prentice Hall, 3 edition, 2007.

[82] Sungmin Hong, Daeyoung Kim, Minkeun Ha, Sungho Bae, Sang Jun Park, Wooyoung Jung, and Jae-Eon Kim. Snail: an ip-based wireless sensor network approach to the internet of things. *Wireless Communications, IEEE*, 17(6):34–42, 2010.

[83] Jun-Cheol Park and Ah-Hyun Jun. A lightweight ipsec adaptation for small devices in ip-based mobile networks. In *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference*, volume 1, pages 5–pp. IEEE, 2006.

[84] Nandakishore Kushalnagar, Gabriel Montenegro, C Schumacher, et al. Ipv6 over low-power wireless personal area networks (6lowpans): overview, assumptions, problem statement, and goals. *RFC4919, August*, 10, 2007.

[85] Edward A Lee. Cyber physical systems: Design challenges. In *Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on*, pages 363–369. IEEE, 2008.

[86] Ragunathan Raj Rajkumar, Insup Lee, Lui Sha, and John Stankovic. Cyber-physical systems: the next computing revolution. In *Proceedings of the 47th Design Automation Conference*, pages 731–736. ACM, 2010.

[87] Ayan Banerjee, Krishna K Venkatasubramanian, Tridib Mukherjee, and Sandeep KS Gupta. Ensuring safety, security, and sustainability of mission-critical cyber–physical systems. *Proceedings of the IEEE*, 100(1):283–299, 2012.

[88] Zach Shelby and Carsten Bormann. *6LoWPAN: The wireless embedded Internet*, volume 43. John Wiley & Sons, 2011.

[89] Clifford Neuman. Challenges in security for cyber-physical systems. In *DHS Workshop on Future Directions in Cyber-Physical Systems Security*, pages 22–24. Citeseer, 2009.

[90] Anhtuan Le, Jonathan Loo, Aboubaker Lasebae, Mahdi Aiash, and Yuan Luo. 6lowpan: a study on qos security threats and countermeasures using intrusion detection system approach. *International Journal of Communication Systems*, 25(9):1189–1212, 2012.

[91] Marc Emmelmann and Julius Mueller. Applying software defined networks and virtualization concepts for next generation mobile broadband networks. *Broadband Ukraine: 12th June 2013, Kiev, Ukraine.*, 2013.

[92] A. Y. Ding, J. Crowcroft, S. Tarkoma, et al. Software defined networking for security enhancement in wireless mobile networks. *Computer Networks*, 66:94–101, 2014.

[93] Lauri Suomalainen, Emad Nikkhouy, Aaron Yi Ding, and Sasu Tarkoma. Open source platforms, applications and tools for software-defined networking and 5g research. 2014.

[94] Yingli Sheng, Haitham Cruickshank, A Dev Pragad, Paul Pangalos, and A Hamid Aghvami. An integrated qos, security and mobility framework for delivering ubiquitous services across all ip-based networks. In *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pages 1–5. IEEE, 2008.

[95] A.K. Salkintzis, C. Fors, and R. Pazhyannur. Wlan-gprs integration for next-generation mobile data networks. *Wireless Communications, IEEE*, 9(5):112 – 124, oct. 2002. ISSN 1536-1284. doi: 10.1109/MWC.2002.1043861.

[96] B. Bhargava, X. Wu, Y. Lu, and W. Wang. Integrating heterogeneous wireless technologies: a cellular aided mobile ad hoc network (cama). *Mobile Networks and Applications*, 9(4):393–408, 2004.

[97] P. Mahonen, J. Riihijarvi, M. Petrova, and Z. Shelby. Hop-by-hop toward future mobile broadband ip. *Communications Magazine, IEEE*, 42(3):138 – 146, mar 2004. ISSN 0163-6804. doi: 10.1109/MCOM.2004.1273785.

[98] I.F. Akyildiz, S. Mohanty, and Jiang Xie. A ubiquitous mobile communication architecture for next-generation heterogeneous wireless systems. *Communications Magazine, IEEE*, 43(6):S29 – S36, june 2005. ISSN 0163-6804. doi: 10.1109/MCOM.2005.1452832.

[99] D. Cavalcanti, D. Agrawal, C. Cordeiro, Bin Xie, and A. Kumar. Issues in integrating cellular networks wlans, and manets: a futuristic heterogeneous wireless network. *Wireless Communications, IEEE*, 12(3):30 – 41, june 2005. ISSN 1536-1284. doi: 10.1109/MWC.2005.1452852.

[100] Pedro A. Aranda Gutierrez and Ilka Miloucheva. Automated qos policy adaptation for heterogeneous access network environments. *Systems and Networks Communication, International Conference on*, 0:65, 2007. doi: http://doi.ieeecomputersociety.org/10.1109/ICSNC.2007.23.

[101] J. Pérez, V. Zárate, and A. Montes. Geseq: A generic security and qos model for traffic priorization over ipsec site to site virtual private networks. *Next Generation Teletraffic and Wired/Wireless Advanced Networking*, pages 175–186, 2007.

[102] Mourad Alia and Marc Lacoste. A qos and security adaptation model for autonomic pervasive systems. In *COMPSAC '08: Proceedings of the 2008 32nd Annual IEEE International Computer Software and Applications Conference*, pages 943–948, Washington, DC, USA, 2008. IEEE Computer Society. ISBN 978-0-7695-3262-2. doi: http://dx.doi.org/10.1109/COMPSAC.2008.283.

[103] D. Vivian, E.A.P. Alchieri, and C.B. Westphall. Evaluation of qos metrics in ad hoc networks with the use of secure routing protocols. In *Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*, pages 1–14, April 2006. doi: 10.1109/NOMS.2006.1687606.

[104] I. Martinovic, F.A. Zdarsky, A. Bachorek, and J.B. Schmitt. Measurement and analysis of handover latencies in ieee 802.11 i secured networks. In *Proceedings of the 13th European Wireless Conference (EW2007), Paris, France*, 2007.

[105] DJ Kang, JJ Lee, BH Kim, and D. Hur. Proposal strategies of key management for data encryption in scada network of electric power systems. *International Journal of Electrical Power & Energy Systems*, 33(9):1521–1526, 2011.

[106] Y. Singh and Y. Chaba. Security and network performance evaluation of kk'cryptographic technique in mobile adhoc networks. In *Advance Computing Conference, 2009. IACC 2009. IEEE International*, pages 1152–1157. IEEE, 2009.

[107] Kire Trivodaliev, Biljana Stojkoska, Ace Dimitrievski, and Danco Davcev. Evaluation issues of different cryptography algoritms in wireless sensor networks. In *International Workshop on Information Security in Wireless Networks*, September 2006.

[108] S. Yau, M. Yan, and D. Huang. Design of service–based systems with adaptive tradeoff between security and service delay. *Autonomic and Trusted Computing*, pages 103–113, 2007.

[109] Xiangqian Chen, Kia Makki, Kang Yen, and Niki Pissinou. Sensor network security: a survey. *Communications Surveys & Tutorials, IEEE*, 11(2):52–73, 2009.

[110] H. Aiache, F. Haettel, L. Lebrun, and C. Tavernier. Improving security and performance of an ad hoc network through a multipath routing strategy. *Journal in computer virology*, 4(4):267–278, 2008.

[111] Alessandro Aldini and Marco Bernardo. A formal approach to the integrated analysis of security and qos. *Reliability Engineering & System Safety*, 92(11):1503–1520, 2007.

[112] A. Avritzer, R. Tanikella, K. James, R.G. Cole, and E. Weyuker. Monitoring for security intrusion using performance signatures. In *Proceedings of the first joint WOSP/SIPEW international conference on Performance engineering*, pages 93–104. ACM, 2010.

[113] Hongli Luo and Mei-Ling Shyu. Differentiated service protection of multimedia transmission via detection of traffic anomalies. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 1539–1542, July 2007. doi: 10.1109/ICME.2007.4284956.

[114] Sílvia Farraposo, Philippe Owezarski, and Edmundo Monteiro. Contribution of anomalies detection and analysis on traffic engineering. In *INFOCOM*. Citeseer, 2006.

[115] I. Askoxylakis, B. Bencsáth, L. Buttyán, L. Dóra, V. Siris, D. Szili, and I. Vajda. Securing multi-operator-based qos-aware mesh networks: requirements and design options. *Wireless Communications and Mobile Computing*, 10(5):622–646, 2010.

[116] Hoon Ko, Kitae Bae, Sang Heon Kim, and Kyung Jin An. A study on the security algorithm for contexts in smart cities. *International Journal of Distributed Sensor Networks*, 2014, 2014.

[117] Jong M Choi, Hoon Ko, Marek R Ogiela, and Goreti Marreiros. Advance in safe and useful social network services with context-sensitive data in cyber-physical system. *International Journal of Distributed Sensor Networks*, 2014, 2014.

[118] Kyungeun Park, Yanggon Kim, and Juno Chang. Semantic reasoning with contextual ontologies on sensor cloud environment. *International Journal of Distributed Sensor Networks*, 2014, 2014.

[119] T. Yu and K. Lin. Service selection algorithms for composing complex services with multiple qos constraints. In *Service-Oriented Computing-ICSOC 2005*, pages 130–143. Springer, 2005.

[120] Mohamad Mehdi, Nizar Bouguila, and Jamal Bentahar. Probabilistic approach for qos-aware recommender system for trustworthy web service selection. *Applied Intelligence*, pages 1–22, 2014.

[121] F. Karatas, M. Bourimi, and D. Kesdogan. Towards visual configuration support for interdependent security goals. In *Online Communities and Social Computing*, pages 375–384. Springer, 2013.

[122] Ernest Friedman-Hill. *JESS in Action*. Manning Greenwich, CT, 2003.

[123] Bruce L Golden, Edward A Wasil, and Patrick T Harker. *Analytic hierarchy process*. Springer, 2003.

[124] Paul Hansen and Franz Ombler. A new method for scoring additive multi-attribute value models using pairwise rankings of alternatives. *Journal of Multi-Criteria Decision Analysis*, 15(3-4):87–107, 2008.

[125] Kui Ren, Shucheng Yu, Wenjing Lou, and Yanchao Zhang. Multi-user broadcast authentication in wireless sensor networks. *Vehicular Technology, IEEE Transactions on*, 58(8):4554–4564, 2009.

[126] Kui Ren, Wenjing Lou, and Yanchao Zhang. Multi-user broadcast authentication in wireless sensor networks. In *Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON '07. 4th Annual IEEE Communications Society Conference on*, pages 223–232, June 2007. doi: 10.1109/SAHCN.2007.4292834.

[127] Ismail Butun and Ravi Sankar. A brief survey of access control in wireless sensor networks. In *Consumer Communications and Networking Conference (CCNC)*, pages 1118–1119, 2011.

[128] Shengli Yuan and Daniel Stewart. Protection of optical networks against interchannel eavesdropping and jamming attacks. In *Computational Science and Computational Intelligence (CSCI), 2014 International Conference on*, volume 1, pages 34–38. IEEE, 2014.

[129] Quanyan Zhu, Walid Saad, Zhu Han, H Vincent Poor, and Tamer Basar. Eavesdropping and jamming in next-generation wireless networks: A game-theoretic approach. In *MILITARY COMMUNICATIONS CONFERENCE, 2011-MILCOM 2011*, pages 119–124. IEEE, 2011.

[130] Wenyuan Xu, Timothy Wood, Wade Trappe, and Yanyong Zhang. Channel surfing and spatial retreats: defenses against wireless denial of service. In *Proceedings of the 3rd ACM workshop on Wireless security*, pages 80–89. ACM, 2004.

[131] Hannes Ekstrom. Qos control in the 3gpp evolved packet system. *Communications Magazine, IEEE*, 47(2):76–83, 2009.

[132] Fatih Karatas and Dogan Kesdogan. A flexible approach for considering interdependent security objectives in service composition. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, pages 1919–1926. ACM, 2013.

[133] Roland Bless and M Rohricht. Secure signaling in next generation networks with nsis. In *Communications, 2009. ICC'09. IEEE International Conference on*, pages 1–6. IEEE, 2009.

[134] Cynthia Irvine and Timothy Levin. Toward a taxonomy and costing method for security services. In *Computer Security Applications Conference, 1999.(ACSAC'99) Proceedings. 15th Annual*, pages 183–188. IEEE, 1999.

[135] Siegfried Benkner and Gerhard Engelbrecht. A generic qos infrastructure for grid web services. In *Telecommunications, 2006. AICT-ICIW'06. International Conference on Internet and Web Applications and Services/Advanced International Conference on*, pages 141–141. IEEE, 2006.

[136] I Yen, MA HUI, Farokh B Bastani, MEI HONG, et al. Qos-reconfigurable web services and compositions for high-assurance systems. *Computer*, 41(8):48–55, 2008.