H.D. Mustafa · Shabbir N. Merchant
Uday B. Desai · Brij Mohan Baveja

Green Symbiotic Cloud Communications

Springer

Green Symbiotic Cloud Communications

H.D. Mustafa · Shabbir N. Merchant
Uday B. Desai · Brij Mohan Baveja

Green Symbiotic Cloud Communications

Springer

H.D. Mustafa
Department of Electrical Engineering
Indian Institute of Technology Bombay
Mumbai, Maharashtra
India

Shabbir N. Merchant
Department of Electrical Engineering
Indian Institute of Technology Bombay
Mumbai, Maharashtra
India

Uday B. Desai
Indian Institute of Technology Hyderabad
Hyderabad, Andhra Pradesh
India

Brij Mohan Baveja
Government of India
Ministry of Communications and
Information Technology
New Delhi, Delhi
India

ISBN 978-981-10-3511-1 ISBN 978-981-10-3512-8 (eBook)
DOI 10.1007/978-981-10-3512-8

Library of Congress Control Number: 2016962048

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #22-06/08 Gateway East, Singapore 189721, Singapore

Preface

A cloud is often defined as a visible collection of particles of ice and water suspended in the air, usually at an elevation above the surface. It is generally a dim and obscure area in something otherwise clear and transparent. The clouds appearing in nature even though visible are abstract and virtual, i.e. we are unable to signify a definite boundary of a cloud. The cloud as defined in the field of computing is, however, very far away from this geographical definition and properties of a natural cloud. Though correlating a cloud with abstraction and virtualization, the existing cloud computing archetypes enfold as backend data or service stations providing bunched or specific services. In this book, we waver from the existing definition of clouds as outsourced services and define an approach to do justice to the geographical existence of clouds and its emulation in the technological domain. We aim to deviate from the traditional approaches of cloud computing and develop an entirely new way to build, deploy and scale technologies and devices of the future. Paradigms enabling convenient, on demand access to a shared pool of configurable resources that can be rapidly provisioned and released with minimal efforts and interactions constitutes our emblem of a Cloud environment. In this book we introduce the idea of cloud communications, wherein abstraction and virtualization currently limited to computing environment, is also embedded in the communications domain an archetype in published literature.

The rapid evolution of the technology used in telecommunication systems, consumer electronics, and specifically mobile devices has been remarkable in the last 20 years. Communication systems handle volumes of data generated by embedded devices, mobile users, enterprises, contextual information, network protocols, location information and such. It is a vast amount of information. For example, a global IP backbone generates over 20 billion records per day, amounting to over 1 Tera Bytes per day! Processing and analysing this "big data" and presenting insights in a timely fashion will become a reality with advanced analytics to understand the environment, to interpret events and to act on them. The existing communication systems are just designed as "dumb pipes" to carry information/data from destination to the source. This book is a positive development that helps unleash the intelligence in communications systems where networks are no longer

labelled “dumb pipes” but highly strategic and smart cognitive networks. The next quality of service leap which is fundamentally expected to come from improvements in network topologies, cooperative communication, virtualization and abstraction schemes, the amalgamation of cognitive symbiotic networks and evolving intelligent protocols, all of which is systematically addressed in developing the Green Symbiotic Cloud Communications (GSCC) architecture.

Mumbai, India H.D. Mustafa
Mumbai, India Shabbir N. Merchant
Hyderabad, India Uday B. Desai
New Delhi, India Brij Mohan Baveja

Contents

About the Authors

H.D. Mustafa has graduated Summa Cum Laude (University Gold Medal) in 2001, from TU, New Orleans, USA with a Double Major in Computer Science and Computer Engineering and a Ph.D. in Electrical Engineering. He is the first recipient of the prestigious Chancellor Fellowship in 1998 for pursuing Undergraduate Education. He is awarded the Honoris Causa, Doctor of Engineering Sciences in May 2015. Currently, he is the President R&D of Transocean Inc. and RIG Ltd. He is a fellow of the ACS and ICCA. He has published over 40 journal papers including publications in reputed international journals. He was the Associate Editor of Hydrocarbon Processing Journal from 2007 to 2009. He is a radical and dynamic multidisciplinary scientist and executive, leading interdisciplinary multibillion dollar R&D and Industrial projects. A multidisciplinary researcher his interest is not bound to any specific field but spans all domains of engineering and science. He is a strategic visionary and team leader amalgamating institutional and industrial research and a skilled expert in observing and translating concepts into reality, with operational excellence in diverse environments.

H.D. Mustafa is the Chief Inventor of the Crude Oil Quality Improvement (COQI®) technology and concept, which has market capitalization of US$ 2.5 Billion as of 2014. He was awarded the Merit Recognition Award by NASA, USA in 2005 and the President of India EMPI Award in 2006 by Government of India for his contributions to Research and Development. He was nominated and shortlisted for the prestigious Global Innovator of the Year Award for 2013. In July 2013, H.D. Mustafa achieved a World Record in India for the deepest oil exploration using the COQI® process.

Shabbir N. Merchant received his B.Tech., M.Tech., and Ph.D. degrees all from Department of Electrical Engineering, Indian Institute of Technology Bombay, India. Currently, he is Professor in Department of Electrical Engineering at IIT Bombay. He has more than 30 years of experience in teaching and research. Dr. Merchant has made significant contributions in the field of signal processing and its applications. His noteworthy contributions have been in solving state of the art signal and image processing problems faced by Indian defence. His broad area

of research interests are wireless communications, wireless sensor networks, signal processing, multimedia communication, and image processing and has published extensively in these areas. He is a co-author with his students who have won Best Paper Awards. He has been a chief investigator for a number of sponsored and consultancy projects. He has served as a consultant to both private industries and defence organizations. He is a Fellow of IETE. He is a recipient of 10th IETE Prof. S. V. C. Aiya Memorial Award for his contribution in the field of detection and tracking. He is also a recipient of 9th IETE S. V. C. Aiya Memorial Award for 'Excellence in Telecom Education'. He is a winner of the 2013 VASVIK Award in the category of Electrical & Electronic Sciences & Technology.

Uday B. Desai received the B.Tech. degree from Indian Institute of Technology, Kanpur, India, in 1974, the M.S. degree from the State University of New York, Buffalo, in 1976, and the Ph.D. degree from The Johns Hopkins University, Baltimore, USA, in 1979, all in Electrical Engineering. Since June 2009 he is the Director of IIT Hyderabad. From 1979 to 1987 he was with the Electrical Engineering Department at Washington State University, Pullman, WA, USA From 1987 to May 2009 he was a Professor in the Electrical Engineering Department at the Indian Institute of Technology—Bombay. He has held Visiting Associate Professor's position at Arizona State University, Purdue University and Stanford University. He was a visiting professor at EPFL, Lausanne during the summer of 2002. From July 2002 to June 2004 he was the Director of HP-IITM R and D Lab. at IIT-Madras. His research interests are in wireless communication, IoT and statistical signal processing. He is the Editor of the book "Modeling and Applications of Stochastic Processes" (Kluwer Academic Press, Boston, USA 1986). He is also a co-author of four books dealing with signal processing and wireless communication. Dr. Desai is a senior member of IEEE, a Fellow of INSA (Indian National Science Academy), Fellow of Indian National Academy of Engineering (INAE). He is on the board of Tata Communications Limited. He was also on the Visitation Panel for University of Ghana.

Brij Mohan Baveja received a B.Tech. from IIT Delhi and two postgraduate degrees from England, one in Electrical Engineering from London University and another in Public Management from Birmingham University, UK. He is presently working as Senior Director & Group Coordinator in Department of Electronics and IT (DeitY), Ministry of Communications and IT, Government of India. He has about 35 years of work experience and out of this he has been working in government for last 29 years. At DeitY, his responsibilities include spearheading R&D in ICT sector through favorable policy measures and by research grants. He has been promoting growth of indigenous Convergence, Communications, and Broadband, Networks including technologies for 5G, IoT, Smart Cities and Strategic Electronics. He has been responsible for setting up various ICT Centers of Excellences and autonomous organizations of DeitY throughout the country. Besides Chairing various National Level Conferences, Workshops, Seminars, Panels, he has represented DeitY at various international forums, such as 3GPP standards meeting in USA, Hungary and Czechoslovakia, World Summit on

Information Society meetings at Geneva. He has been speaker on sessions in bridging the digital divide at South Korea and Japan. He has chaired sessions on ICT meetings at SAARC countries. He has coordinated international programs of Indo-German, UNDP, Indo-Swiss, Indo-US, Indo-Ireland, Indo-Dutch and Israel. Prior to joining the government in 1986, he has worked in Bharat Electronics Ltd for 3 years at Ghaziabad Unit and prior to this, for 2 years in the UK with a test software company. He has authored over 12 publications as well as 2 book chapters. He is an ISO certified lead assessor and Licensed Amateur Radio (Ham) Operator. He is also fellow member of IETE. He is an IEEE member since 1974.

Chapter 1
Introduction

Fast evolution in grid computing, high performance computing and cloud computing has facilitated the growth of improved networking and resource management philosophies. However, the deployments of these technologies are severely stunted owing to their entirely economic motives. Grids and clouds today are primarily to facilitate large organizations and enterprises, and only limited efforts have been made towards establishment of local, flexible clouds, independent of data-centers and service providers. The evolution of communications and computing technologies is driven by the creation and development of new services for mobile devices, and is enabled by advancement of the technology available for mobile systems. There has also been an evolution of the environment in which mobile systems are deployed and operated, in terms of competition between mobile operators, challenges from other mobile technologies, and new regulation of spectrum use and market aspects of mobile systems. The rapid evolution of the technology used in telecommunication systems, consumer electronics, and specifically mobile devices has been remarkable in the last 20 years. Moores law illustrates this and indicates a continuing evolution of processor performance and increased memory size, often combined with reduced size, power consumption, and cost for devices.

A paradigm enabling convenient, on-demand access to a shared pool of configurable resources that can be rapidly provisioned and released with minimal efforts and interactions constitutes the emblem of a cloud environment. Though correlating a cloud with abstraction and virtualization, the existing archetypes enfold as back-end data or service stations providing bunched or specific services. In this book, we aim to deviate from the traditional approaches of cloud computing and develop an entirely new way to build, deploy and scale technologies and devices of the future. The idea of cloud communications is introduced, wherein abstraction and virtualization is embedded in the communications domain, an archetype in published literature. Furthermore, with emerging emphasis on stewardship towards the environment, greener approaches have become a fundamental necessity. We collectively coin this approach **Green Symbiotic Cloud Communications** (*GSCC*).

H.D. Mustafa et al., *Green Symbiotic Cloud Communications*,
DOI 10.1007/978-981-10-3512-8_1

Our paramount achievement is implementing cloud functionalities symbiotically over the existing infrastructure in an abstract sense creating a virtual cloud communications scenario. In this outlook, the individual users now become elements constituting the cloud. Key to this approach is the ability to manage the use of communications resources on individuals as the cloud infrastructure and make the users element of the cloud. The cloud so formed is sufficiently non-intrusive to the users, that they will permit its operation in their domains. This approach entails extensive reformulation of existing and development of new algorithms and schematics, embedding cognition in protocol designs and development of new constraints and metrics. This book addresses these issues in a structured and systematic approach and develops a test prototype for *GSCC*.

1.1 Cloud Computing and Communications

A cloud is often defined as a visible collection of particles of ice and water suspended in the air, usually at an elevation above the surface. It is generally a dim and obscure area in something otherwise clear and transparent. The cloud computing paradigm accepted within the scientific community, however, is far from this geographical definition. This book purposes an approach to do justice to the classical definition and forms a rational basis for advancement towards the same in evolving technologies of the future.

Evolution has been fast. With research in High Performance Computing (*HPC*), processors were able to use parallel processing algorithms and software to divide programs into little pieces and execute them simultaneously [1]. Grid computing, a development on this, brought you the ability to make more cost-effective use of a given amount of computer resources and essentially a way to solve problems that couldnot be approached without an enormous amount of computing power [2]. It suggested that the resources of many computers could be cooperatively harnessed and managed towards a common objective. In the envisioned paradigm, computers will truly collaborate rather than being directed by one managing computer.

Virtualization, a heavily researched area, expedited the process of scaling standards and concepts through networks in a hierarchical and cyclic manner [3]. A further development and an explosive research topic today, cloud computing and communication, developed by borrowing a little from grid computing, distributed computing and remote client facilitation. Working over a network of distributed servers, it provides "*X*" (IT, computing resource, etc.) as a service to users in a cost-effective, on-demand manner [4]. It enables the user to associate certain smart devices to this cloud server, e.g. *VMWare, Microsoft Azure, Amazon EC2* [5]. At the enterprise level, cloud service providers let clients link limited devices at the enterprise through a server to the cloud [6]. This, however, is no different from outsourcing your resource and service requirements to a third party, especially a very concentrated (not distributed) third party. Further, the services offered are limited in nature through data centers and are minimal over the ubiquitous local networks,

a very potential and ubiquitous resource by itself [7]. Cloud computing today is a disjoint collection of independent cloud services each handled and facilitated by a certain service provider running a data center.

Cloud computing, grid computing, *HPC* and data center computing largely fall within the realm of parallel computing [5]. While *HPC* focuses on scientific computing, with high processing performance and low delay as its cornerstones, grid computing, simply based on *HPC* centers, utilizes multiple connected *HPC* centers to form a large grid which owns a powerful underlying concept—service-oriented architectures. Some other impressive concepts such as utility computing and autonomic computing are more economy-oriented and are addressed vaguely [4]. Cloud computing, which today is based on data centers, is spreading far more quickly than traditional grid computing [8].

Technological progress in *HPC* largely revolves around two well-defined cornerstones. Transitory in nature, both trace changes over time, the two decades since the introduction of the technology. One, representing gross performance (hypothetical) measured in floating-point operations per second (flops), progressing toward the brink of teraflops performance [3]. The other, evaluating gross purchase price per unit of performance shows a constant downward trend [9]. Considered together, they question the present state in which phenomenally high performance is available at prices that were unimaginable a few years ago [10]. The surviving architectural class is slim. The only remaining exclusively distributed-memory solutions are the IBM's *RS/6000* machines and *SGI*/Cray's *T3D* and *T3E* systems [9].

In pursuit of scalability and intensiveness, the original *HPC*, post technological maturity, evolved into grid computing. A more advanced environment, grid computing enhanced our capabilities to model and simulate complex systems arising in commercial, scientific and engineering applications. The principle of grid computing has been ubiquitous availability of computational resources to a computing task, in the same manner in which electricity is provided readily through electrical power grids [8, 11]. The computational grid entails on-demand access to computation and communication resources, as well as access to storage systems and virtualization systems. Further, grid computing can be visualized as a network that is not isolated in one place but is a collective of distributed resources such as computers, peripherals, switches, instruments and data. While its ownership may be sporadically distributed, being a special type of middleware, it enables sharing and management of grid components based on resource attributes and user requirements [3].

The first generation grid, contrived in the early '90s, was a model of metacomputing in which supercomputers shared resources and came equipped with the ability to share data [2, 4]. The late '90s saw the emergence of a solid framework for second generation grids, characterized by their use of grid middleware systems to amalgamate different technologies together [9]. The combination of web technologies with second generation grids paved the way for emergence of third generation grids [12]. However, there are fundamental gaps between current grid technologies and the prospective Next Generation Grid vision which places ubiquity and the ability to self-manage as the top two grid priorities [13].

Ubiquity, a composite feature made up of other primitives, mainly accessibility, user-centricity and dynamic interaction, was carried forward as a leading guideline for emerging grid technologies and cloud computing [1, 9]. Emerging grid technologies can be classified into the 4 categories: accessible grids, user-centric grids, interactive grids and manageable grids [5, 6]. Accessible grids (ad hoc grids, mobile grids and wireless grids) are different from traditional grids in that they have no predefined entry points, support mobility of clients and services, and support wireless connections between nodes and interfaces [8]. Interactive grids (explicit interactive grids and context-aware grids) support explicit real-time interaction with end users and are inherently capable of interacting with their surroundings to build context and adapt their behavior [8, 12, 13]. User-centric grids (personal grids and personalized grids) are the sort that are owned and operated by individuals and implement highly customizable grid portals [14]. Manageable grids (autonomic grids, knowledge grids and organic grids) emulate genetic schemes such as the human bodys autonomic nervous system or ant colonies to support self-managing [11].

A discussion on cloud computing typically begins with a talk on virtualization. Virtualization, a critical component to cloud computing, simplifies delivery of services by providing a platform for optimizing *IT* resources in a scalable manner [15, 16]. This is precisely what makes cloud computing so cost-effective. Virtualization can be applied very broadly to just about everything you can imagine including memory, networks, storage, hardware, operating systems and applications. We envision virtualization to take cloud computing further to an entire new level where no entity, in particular, is a service provider or a client. One or both of the devices may act at any given instant and exist as multiple instances.

Cloud computing has improved computation efficiency while reducing its cost for users [8, 13]. A typical cloud, today, relies on back-end data-centers consisting of thousands of servers and switches connected hierarchically. By sharing computing resources through services such as software as a service, users can eliminate the cost of hardware and software [1, 6, 11]. To ease system upgrades and maintenance, virtual machines (*VM*s) are often employed to provide services. Further, their migration across the physical hosts results in higher resource utilization. Although most familiar research surrounds sharing *CPU* and storage facilitation, the sharing and partitioning of network resources is only understood vaguely. Resource virtualization, however, remains the key component of cloud computing for providing computing and storage services [17].

Data-centers use virtualization methods to abstract the commonness of infrastructure at different levels. Comprised of physical networks connected via switches and virtual networks made of *VM*s running inside hosts, data-center networks are capable of servicing multiple users while running thousands of tasks concomitantly [15, 16]. Several *VM*s may co-exist within a host and each such *VM* has at least one Virtual Network Interface Card (*VNIC*). The *VNIC*s communicate with external networks through the host's Physical *NIC*s (*PNIC*s). A software layer facilitates the traffic multiplexing between the *VNIC*s and *PNIC*s. This software layer can be either rudimentary Ethernet bridges or full-fledged virtual Ethernet switches. Notably, virtual switching at *NIC*s leads to better resource utilization at the hosts. From the

purview of performance improvement, the packet processors at *NIC*s, optimized for packet processing, end up providing better packet-switching performance than the host *CPU*s. Further, the isolation of computing and packet switching through *VM* computing and virtual-switch packet processing significantly reduce the switching overheads and complexity of buffer management [12].

While the approach observed is intuitive in nature, it nevertheless poses a valid management predicament. Each host has up to hundreds of *VM*s and *VNIC*s. Even the most heuristic of tasks such as allocating the *IP* addresses are challenging and error-prone [16]. The virtualization software layer must adopt optimal administrative schemes in all *VM*s across different physical switches. Another obvious challenge is the complexity of packet multiplexing [1, 3, 10, 11, 16]. When a packet arrives, the host is expected to determine the packet's destination *VM* based on its header. The growing diversity in network protocols makes it difficult to parse the headers and look up the bridging table [14]. For example, it's not trivial to deliver broadcast packets to only the *VM*s belonging to a particular virtual local area network. Further, with significantly higher line rates expected to appear at hosts, a profound weight is added to the workloads of the host *CPU*s in future data centers, which handle not only the *VM* computation but also the network *I/O* virtualization at hundreds of *Gbps*. The scalability of multicore *CPU*s remains uncertain under such demanding tasks.

Security fall-outs stemming from virtualization, a growing concern in cloud-service architecture development, have prompted a fairly intense discussion on the robustness of virtualization worlds [16]. Cloud security compromises can originate in poor management of keys/passwords, vague service-level agreements or deficient service-oriented architectures. However, being fundamental to several cloud characteristics, virtualization is retained as is compensated by an increased emphasis on security. Virtualization lets users simultaneously run multiple isolated machines on a single physical machine (the host machine). The hypervisor, or *VM* monitor, is the software that sits between the host machine and the *VM*. The hypervisor allocates and manages the physical resources among the *VM*s. In cloud computing, this translates as service providers creating customized *VM*s for the service user.

1.2 Heterogeneous Networks

Spectral efficiency and increased throughput in wireless communications are topics of paramount importance owing to the substantial increase in user demands [12]. Easy and affordable availability of communication devices like smart mobile phones, laptops, tablets, etc. has resulted in its proliferation and subsequent exponential surge in radio traffic [13]. The next quality of service leap is fundamentally expected to come from improvements in network topologies, cooperative communication and virtualization schemes, the amalgamation of cognitive heterogeneous networks and standardization of protocols on such networks [12–16]. Imagine a scenario where a smartphone user makes a video call requiring a capacity of 6 Mbps for uplink and

downlink. The user is equipped with unlimited *WLAN* and Long Term Evolution (*LTE*), with each of these mediums providing a capacity of 4 Mbps respectively. In such a case, neither of the mediums can provide the required capacity on a standalone basis. Scrutinizing the scenario, one realizes that if simultaneous usage of both the mediums is allowed, the desired throughput can be achieved. Ubiquitous access to all available networks, thus, is the key for the end users to have guaranteed quality of service. The proposed paradigm attempts to integrate this thought evolving a green symbiotic cloud communications paradigm.

In [18], a software-based approach is adopted which virtualizes a single wireless adapter as multiple entities. The paradigm, termed as "*Multinet*", is defined as virtualizing a singular *WLAN* adapter as multiple mirrors, where each can connect to a different wireless network. This facilitates connections to multiple wireless networks by the user, by multiplexing the single network adapter. The software manages an "adoptive network-hopping scheme", which assigns a specific time slot to each virtual adapter during which it is able to send its information. However, due to availability of only one communication port and allotting of specific time slots based on connection strength, it practically means automation of selection of the network having the highest throughput, which the users otherwise perform manually. Furthermore, the adaptive scheme, which continuously scans the various networks to find the strongest one, is taxing from an operational power dynamics viewpoint.

In [19], an augmented system of *WLAN* and *cellular 3G* is proposed. The paradigm, termed "*Wiffler*", scans for the availability of *WLAN* networks for the users and subsequently offloads the cellular data through these access points. The system assigns the routing based on the actual throughput offered by the strongest available access point (*AP*s) coupled with a prediction of the number of such *AP*s that will be available in the future. If the delay tolerance of a *WLAN AP* is found to be within a predefined threshold, then data from the cellular network is offloaded to it. It also leverages "low-level, link-layer information" to enable fast switching to 3G in the face of poor *Wi-Fi* conditions. However, this system only uses either *cellular* or *WLAN* at any given instance of time, hence not availing the combined benefits both the environments have to offer. Furthermore it is complicated to implement, as it requires the knowledge of delay tolerance threshold and *QoS* requirements of every application that uses the network. It also operates on proxy support, which might not be available on all platforms.

Coordinated Multipoint (*COMP*) is introduced in [20], wherein the concept of spatial reuse is adapted in cellular networks to increase throughput. The authors propose schemes like "*Interference aware detection*", "*Joint multicell scheduling*" and "*Joint multicell signal processing*" where different base stations help in relaying information cooperatively. These proposed schemes mainly aim to avoid or exploit interference to obtain a better throughput in the cell edge regions. The system allows more efficient usage of a single communication medium but is devoid of the advantages availed from multiple mediums. Moreover, the computational complexity and high data exchange between base stations, to identify interference in real time, requires high bandwidth and low latency.

In [21] and references therein, the authors describe a multipath *TCP* (*mpTCP*) architecture. The paradigm allows users to access multiple communication paths simultaneously by proposing an extension over the regular *TCP* protocol. A basic assumption which *mpTCP* makes is that in every session, both the hosts are "multi-homed or multi-addressed", which is logically not always the case. Furthermore, the protocol is not able to function through middle boxes like *NAT*, firewalls and proxies. It also adapts a master-slave strategy which exposes security threats, wherein a user who has knowledge of the master *IP* can easily route information to itself.

Generalized heterogeneous networks as proposed in [22–25] employ the use of low-powered nodes like femto-cells and pico-cells, which are in turn powered by macro base stations. These approaches avail the benefit of energy-efficient operations and cover dead holes in network coverage, but increase the infrastructural cost of implementation. Furthermore, they do not utilize the complete spectrum and radio resources available in modern day *UE*'s. Unlicensed Media Access is the 3rd Generation Partnership Project global standard for subscriber access to mobile circuit, packet and *IMS*-based services over any *IP*-based access network. From the point of view of mobile operators, the *UMA* solution can take the cost and performance advantages of *IP* access technologies to deliver high-quality, low-cost mobile voice and data services, thus enabling mobile operators to extend cellular network coverage through *Wi-Fi* hotspots with minimal additional investment. From the point of view of subscribers, the *UMA* solution can provide a better user experience since there is better *Wi-Fi* signal coverage indoors, which is very important in some countries where adequate cellular coverage indoors is not yet available. In the proposed paradigm, which follows in the subsequent chapters, we aim to overcome these drawbacks of existing heterogeneous networks.

To the best of our knowledge, existing prototypes of heterogeneous networks only concentrate on throughput improvement on the uplink. Communication, however, is a two-way process with the downlink being just as critical as the uplink. In fact, more often than not, downlink carries a larger throughput demand as compared to the uplink [14]. Hence, a significant overall throughput increase necessitates an improvement in both downlink and uplink data rates. Further, in light of greenness of recent emerging trends and governmental regulations, this too needs to be energy-efficient [17, 18]. Consequently, prevention of energy drain and complementing the lifetime of battery-operated devices like cellular phones, PDAs, tablets, etc. forms an essential criterion in the architectural design [17]. As a part of the proposed paradigm, we make the following novel contributions in a distributed and hierarchical manner. We propose a paradigm which enables *UE*s to simultaneously utilize all Communications Links (*CL*s) in the vicinity where it is authorized to, creating a virtual and abstract communication cloud. A virtualized communication port and *IP* schematic is proposed, which enables both uplink and downlink communications via all links in a distributed and symbiotic manner. We establish the networking theoretic capacity within the operational constraints and identify the power dynamics of the proposed prototype by framing a cognitive decision function.

Incorporation of varied signal transmission through a singular facet is a major advantage offered in the *GSCC* architecture. However, this advantage has not entirely

been exploited owing to non-amalgamation of connectivity and functionality. This book will present a fresh dimension towards instituting an innovative prototype of a Universal Green Modem (*UGM*) moulding a comprehensive functional architecture. The astutely designed paradigm allows for one modem to control multitude signal routing, effectively delineating a universal communication system. *UGM* is efficiently supported by an Optimized Coulomb Energy Neural Network (OCENN) with a novel energy allocation, training and testing procedure, which results in more efficient hardware implementation of signal classification and routing. The modified *OCENN* results in a 50% reduction in the overlap of the hyperspheres and a 30% reduction in the false negatives in the classification tier, exponentially enhancing the operability of the *UGM* in real-time situations. We propose an innovative addressing schematic, enabling bidirectional communications over the *GSCC* system, an archetype in published literature. The unique addressing scheme enables bidirectional integration of various communication models (viz. telephone, internet, radio and television). A unique archetype of embedded Channel Estimation is proposed to support the classification of the *OCENN* structure. Experimental results of the proposed *UGM* are evaluated over a real-time *GSCC* system.

The major advantage of a *GSCC* paradigm is the unification of the communication architecture into a singular medium. In our efforts towards this unification, we propose a non-invasive, reflection coefficient-based channel estimation technique over low voltage transmission lines. The novel model accounts for all the parameters of the signal, viz. quality, strength and attenuation, by simplistic inference from the measured magnetic field intensity. To validate the proposed model, experimental results are conducted over real-time low voltage PLC architecture. Results of the extensively measured values are compared with its respective actual conditions. We obtain an average accuracy of 97.61, 97.49 and 97.86% for the varying frequency, load and voltage profiles respectively. We attain a mean error of 2.5% which is lower than all the existing models, thus promising a strong candidature for *GSCC*-based systems.

References

1. A. Iosup, S. Ostermann, M.N. Yigitbasi, R. Prodan, T. Fahringer, and D. H J Epema. Performance analysis of cloud computing services for many-tasks scientific computing. *Parallel and Distributed Systems, IEEE Transactions on*, 22(6):931–945, 2011.
2. Kwang Mong Sim. Agent-based cloud computing. *Services Computing, IEEE Transactions on*, 5(4):564–577, 2012.
3. C. Papagianni, A. Leivadeas, S. Papavassiliou, V. Maglaris, C. Cervello-Pastor, and A. Monje. On the optimal allocation of virtual resources in cloud computing networks. *Computers, IEEE Transactions on*, 62(6):1060–1071, 2013.
4. D. Talia. Clouds meet agents: Toward intelligent cloud services. *Internet Computing, IEEE*, 16(2):78–81, 2012.
5. P. Kalagiakos and P. Karampelas. Cloud computing learning. In *Application of Information and Communication Technologies (AICT), 2011 5th International Conference on*, pages 1–4, 2011.

6. M. Bourguiba, K. Haddadou, I. El Korbi, and G. Pujolle. Improving network i/o virtualization for cloud computing. *Parallel and Distributed Systems, IEEE Transactions on*, pp(99), 2013.
7. M. Shiraz, A. Gani, R. Khokhar, and R. Buyya. A review on distributed application processing frameworks in smart mobile devices for mobile cloud computing. *Communications Surveys & Tutorials, IEEE*, PP(99), 2013.
8. Frederica Darema. Grid computing and beyond: The context of dynamic data driven applications systems. *Proceedings of the IEEE*, 93(3):692–697, 2005.
9. M. AbdelBaky, M. Parashar, Hyunjoo Kim, K.E. Jordan, V. Sachdeva, J. Sexton, H. Jamjoom, Zon-Yin Shae, G. Pencheva, R. Tavakoli, and M.F. Wheeler. Enabling high-performance computing as a service. *Computer*, 45(10):72–80, 2012.
10. M. Lemay, Kim-Khoa Nguyen, B. St. Arnaud, and M. Cheriet. Toward a zero-carbon network: Converging cloud computing and network virtualization. *Internet Computing, IEEE*, 16(6): 51–59, 2012.
11. B. Kantarci and H.T Mouftah. Designing an energy-efficient cloud network [invited]. *Optical Communications and Networking, IEEE/OSA Journal of*, 4(11):B101–B113, 2012.
12. N. Xiong, W. Han, and A. vandenBerg. Green cloud computing schemes based on networks: a survey. *Communications, IET*, 6(18):3294–3300, 2012.
13. FeiFei Chen, J.-G. Schneider, Yun Yang, John Grundy, and Qiang He. An energy consumption model and analysis tool for cloud computing environments. *Green and Sustainable Software (GREENS), 2012 First International Workshop on*, pages 45–50, 2012.
14. C.J. Sher Decusatis, A. Carranza, and C.M. DeCusatis. Communication within clouds: open standards and proprietary protocols for data center networking. *Communications Magazine, IEEE*, 50(9):26–33, 2012.
15. Ismail Sengr Altingvde and zgr Ulusoy. Exploiting interclass rules for focused crawling. *Intelligent Systems, IEEE*, 19(6):66–73, 2004.
16. Hsin-Yi Tsai, M. Siebenhaar, A. Miede, Yu-Lun Huang, and R Steinmetz. Threat as a service?: Virtualization's impact on cloud security. *IT Professional*, 14(1):32–37, 2012.
17. Jagadish Ghimire and Catherine Rosenberg. Resource allocation, transmission coordination and user association in heterogeneous networks: A flow-based unified approach. *IEEE Transactions on Wireless Communications*, 12(3):11, 2013.
18. R. Chandra, P. Bahl, and P. Bahl. Multinet: connecting to multiple ieee 802.11 networks using a single wireless card. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, page 12, 2004.
19. Ashish Sharma, Elizabeth M. Belding, and Cliarles E. Perkins. Augmenting mobile 3g using wifi. In *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*, page 5, 2009.
20. R Irmer, H Droste, P Marsch, M Grieger, G Fettweis, S Brueck, H.P. Mayer, L Thiele, and V Jungnickel. Co-ordinated multipoint: Concepts, performance, and field trial results. *Communications Magazine, IEEE*, 49(2):102–111, 2011.
21. A. Kostopoulos, Warma, T Leva, B Heinrich, A Ford, and L Eggert. Towards multipath tcp adoption: Challenges and opportunities. In *Next Generation Internet (NGI), 2010 6th EURO-NF Conference on*, pages 1–8, 2010.
22. Stefan Brueck. Heterogeneous networks in lte-advanced. In *Wireless Communication Systems (ISWCS), 2011 8th International Symposium on*, page 5, 2011.
23. X. Chen, Z. Dai, W. Li, Y. Hu, J. Wu, H. Shi, and S Lu. Prohet: A probabilistic routing protocol with assured delivery rate in wireless heterogeneous sensor networks. *IEEE Transactions on Wireless Communications*, pp(99):8, 2013.
24. H. ElSawy and E. Hossain. Two-tier hetnets with cognitive femtocells: Downlink performance modeling and analysis in a multi-channel environment. *IEEE Transactions on Mobile Computing*, pp(99):1, 2013.
25. Aamod Khandekar, Naga Bhushan, Ji Tingfang, and Vieri Vanghi. Lte-advanced: Heterogeneous networks. In *Wireless Conference (EW), 2010 European*, page 4, 2010.

Chapter 2
Green Symbiotic Cloud Communications

Abstract Cloud computing, a *TCP/IP* based development, is essentially an integration of computer technologies such as *HPC*, massive memory resource handling, high-speed networks and reliable system architecture. A unified definition of cloud computing doesn't exist with researchers and industrialists globally having listed up to 22 definitions to provide a comprehensive analysis of all the characteristics of Cloud Computing. However Cloud Computing mainly entails as a service that is outsourced, and does not symbolically represent a cloud as we observe in Nature. Classified exhaustively, clouds fit into the following categories—public, private, community and hybrid—however, without much exclusivity. This chapter views the emblem of cloud computing from a different perspective by emulating the geographical cloud as it appears in nature with properties of abstraction and virtualization. The chapter further introduces a first of its kind concept of Cloud Communications. To the best of our knowledge this is an archetype approach of incorporating the communications infrastructure into the cloud. The chapter proposes a Green Symbiotic Cloud (*GSC*) paradigm, which is an amalgamation of all sorts of clouds, with the elimination/minimization of reliance on data-centers, agent-based cooperative approaches and self-managed platforms inherent to systems of the future. Backed by concepts of abstraction and virtualized infrastructure and shared resource pools in one's own local area network, the proposed paradigm offers impetus to revolutionize cloud computing. Taking virtualization to an entirely new level by offering a more local, energy-efficient, synergistic system comprised of individual agents sharing not just resources but knowledge/intelligence in the cloud, it basically emulates the cloud as it appears in nature.

Cloud computing, a *TCP/IP* based development, is essentially an integration of computer technologies such as *HPC*, massive memory resource handling, high-speed networks and reliable system architecture [1–3]. However, its reliability rests on optimality of the connecting protocols, maturity of data-center technologies and service-level agreements. Because a unified definition of cloud computing doesn't exist, proposing a change makes it of utmost importance to lay out essential elements of cloud computing and define it. This will assist us in determining the scope of research and suitability of applications running as cloud services. The key

H.D. Mustafa et al., *Green Symbiotic Cloud Communications*,
DOI 10.1007/978-981-10-3512-8_2

characteristics of cloud computing are agility, low cost, device and location independence, multi-tenancy, high reliability, high scalability, security and sustainability. Researchers and industrialists globally have listed up to 22 definitions and provide a comprehensive analysis of all the characteristics of Cloud Computing [3–6]. "Clouds are large pools of virtualized resources which are easy to access, secure and reliable. There are 10 characteristics of cloud computing in their sum-up: user-friendliness, scalability, resource optimization, pay-per-use, virtualization, Internet-centric, variety of resources, learning-based adaptation, service *SLA*s (Service-Level Agreements) and infrastructure *SLA*s." Cloud computing, as a platform, possesses characteristics of both clusters and grids and primarily provides services to users without knowing much about the infrastructure.

Classified exhaustively, clouds fit into the following categories—public, private, community and hybrid—however, without much exclusivity [4, 7]. A private cloud is hosted within an enterprise, behind its firewall, and intended for use inside the enterprise's offices only; in such cases, a single firm invests in and manages its own cloud infrastructure. It avoids establishment of networks based on large number of low performance systems and instead benefits from the pooling of a smaller number of centrally maintained computing and storage resources [8]. In contrast, a public cloud is hosted on the Internet and designed to be used by any user with an Internet connection to provide a similar range of capabilities and services [8–11]. While they may not differ much under technical considerations, security considerations may be substantially different for services (applications, storage, and other resources) that are made available by a service provider (independent owner and operator) for a public audience and when communication is effected over a non-trusted network.

Community clouds, a method to share infrastructure between several enterprises of a community, stems from the firms' need to address certain common concerns (security, compliance, jurisdiction, etc.) [10]. Being managed either internally or by a third party, the costs are spread over fewer users than in a public cloud. Hybrid clouds are formed out of the composition of two or more clouds (private, community or public) that remain unique entities bound together and offer the benefits of multiple deployment models [12]. Such composition promotes new synergies and implementation options for cloud services while permitting organizations to use public cloud computing resources only sporadically. This capability enables hybrid clouds to scale elastically. The proposed Green Symbiotic Cloud (*GSC*) paradigm is an amalgamation of all sorts of clouds, with the elimination/minimization of reliance on data-centers. A self-configuring scheme GSC Communications proposed in this book treats all networked devices in a locality as the cloud infrastructure.

Cloud services are a discrete set consisting of elements of the sort *XaaS* ('*X*' as a service) [1, 4, 7]. For instance, a user can get service from a full computer infrastructure through the Internet. This kind of service is called Infrastructure as a Service (*IaaS*). Internet-based services such as storage and databases are part of the *IaaS*. Other types of services on the Internet are Platform as a Service (*PaaS*) and Software as a Service (*SaaS*). *PaaS* offers full or partial application development that users can access, while *SaaS* provides a complete application, such as Organization resource management, through the Internet. In the typical cloud service model, providers of

IaaS offer physical machines or more often, virtual machines and other resources. A hypervisor runs the virtual machines as guests. A collective of such hypervisors within the cloud's executive support system help support large numbers of virtual machines and offer the ability to scale services up and down according to customers' varying requirements. *IaaS* clouds often offer additional resources such as a virtual machine disk image library, block- and file-based storage, firewalls, load balancers, *IP* addresses, virtual local area networks and software bundles. *IaaS*-cloud providers supply these resources on-demand from their large pools installed in data-centers [6, 13, 14].

In the *PaaS* model, cloud providers deliver a computing platform typically including operating system, programming language execution environment, database and web server. Application developers can develop and run their software solutions on a cloud platform without the cost and complexity of buying and managing the underlying hardware and software layers. With some *PaaS* offers, the underlying computer and storage resources scale automatically to match application demand such that the cloud user does not have to allocate resources manually [12, 15, 16].

In the *SaaS* model, cloud providers install and operate application software in the cloud and cloud users access the software from cloud clients. Cloud users do not manage the cloud infrastructure and platform where the application runs. This eliminates the need to install and run the application on the cloud user's own computers, which simplifies maintenance and accommodates a large number of cloud users that cloud applications support [17]. Cloud applications are different from other applications in their scalability—which can be achieved by cloning tasks onto multiple virtual machines at run-time to meet changing work demand. Load balancers distribute the work over the set of virtual machines. This process is transparent to the cloud user, who sees only a single access point. To accommodate a large number of cloud users, cloud applications can be multitenant, i.e. any machine serves more than one cloud user organization. It is common to refer to special type of cloud-based application software with a similar naming convention: desktop as a service, business process as a service, test environment as a service, communication as a service [6, 7, 10].

A category of cloud services where the capability provided to the cloud service user is to use network/transport connectivity services and/or inter-cloud network connectivity services. *NaaS* involves the optimization of resource allocations by considering network and computing resources as a unified whole. Traditional *NaaS* services include flexible and extended VPN and bandwidth on-demand. *NaaS* concept materialization also includes the provision of a virtual network service by the owners of the network infrastructure to a third party [4, 7, 12, 17].

2.1 Transcending Generic Cloud Computing

In essence, ***GSC*** as shown in Fig. 2.1 proposes a dynamic computing infrastructure free of the data-centers, agent-based cooperative approaches and self-managed platforms inherent to systems of the future. Backed by virtualized infrastructure and

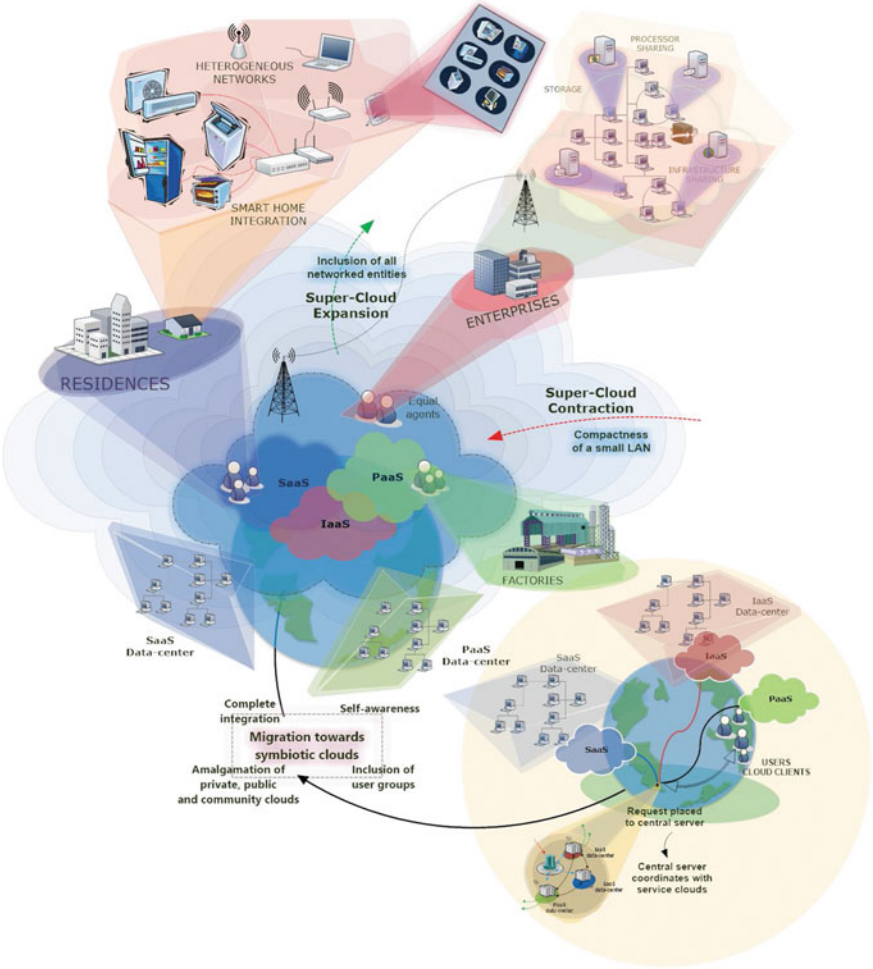

Fig. 2.1 The figure shows the migration from traditional cloud computing/communications architecture to the proposed *GSCC*. Current cloud services rely on dedicated data centres for *IaaS, SaaS* and *PaaS*—which may reside at different data centres. Systems of the future will be strongly networked with networks spanning across residences, enterprises, factories, geographies, etc. The figure depicts user groups in different locales being connected symbiotically and maximally utilizing their resources. Smart home systems and other traditionally standalone devices are networked and accessible. The cloud, now a super-cloud, expands and contracts to include/exclude arriving/departing devices accordingly. Users are abstracted and multiple virtualization worlds emerge every instant

shared resource pools in one's own local area network, the proposed paradigm offers impetus to revolutionize cloud computing. Taking virtualization to an entirely new level by offering a more local, energy-efficient, synergistic system comprised of individual agents sharing not just resources but knowledge/intelligence in the cloud, it basically emulates the cloud as it appears in nature.

User experience, a subject of human computer interaction, is an important criterion when evaluating success and user-friendliness of such a paradigm. In cloud computing, user experience improves a lot over ancestors of the likes of grid computing. The core of the experience demands that the services be provided in an inexpensive and easily accessible manner to the cloud user. *GSC* envisions a far more holistic user experience offering cloud services in a more diffused, faster and energy-efficient manner. It offers the user an opportunity to exploit resources in the vicinity rather than relying on a distant data-center. New synergies between both intelligent (with computing power) and unintelligent devices (without computing power) can be explored, promoting progress towards a more integrated super-cloud.

While abstraction is a key component of a consumer-oriented cloud, it needs to make way in a system where there are administrators. If the users are systems administrators and integrators, what they care is how things are maintained in the cloud. They upgrade, install, and virtualize servers and applications. If the users are consumers, they do not care how things are run in the system, rather they look for speed, economy, security and some level of service quality. Grid computing requires the use of software that can divide and farm out pieces of a program as one large system image to a great number of computers. One concern about grids is that if one piece of the software on a node fails, other pieces of the software on other nodes may fail. This is alleviated if that component has a failover component on another node, but problems can still arise if components rely on other pieces of software to accomplish one or more grid computing tasks. Relying on an equal access model on the other hand, *GSC* proposes an advanced level of abstraction for all users. Unless a developer mode is accessed, the diffused cloud and its operatives should remain fully abstracted from the user. Some of these departures are not serious gains, but clouds, like any other ecology, benefit from diversity.

2.2 Pedestals for Systems of Future

2.2.1 Data Caching—*Downloading the WWW onto the Cloud*

Why should a device have to connect to the World Wide Web every time it requires a resource, more so in case of frequently accessed resources? Consider a locally configured cloud network. For instance, all host computers on the local area network at a school may constitute such a network. Notably, most devices connected to such a network sit behind a gateway server and do not possess public *IP* addresses; further,

these devices are configured to fit a standard network topology and are provided *IP* addresses either dynamically or by using a static addressing scheme.

Routinely or not, if a computer *A* were to access a particular webpage *X*, it sends a request that is usually referenced by the *DNS* and forwarded by the gateway, after it updates its *NAT*. Now, if this webpage were recently accessed, referenced and an instance be stored for a fairly large duration, the next time a user requests the same webpage within this duration, the gateway's crawler hops through the *LAN* scavenging all stored instances to locate the desired page. If the page were to be stored in a well-referenced and hierarchical manner, it'd be fairly easy to locate and an appreciable hit-rate can be expected. This model inherently makes the *LAN* cloud more self-reliant. It mitigates the necessity to request and access distant web-servers on the World Wide Web, thereby optimizing the overall throughput. In the process, overall utilization of the abundant local network resources, such as storage and computational capacity, and minimizes load on the *WWW*. Over time, the system is expected to learn from requests, hits and web accesses, and progress towards significantly higher hit-rates.

A very fundamental technique would look like this:

- Heuristic search: release crawlers from both ends (gateway and user equipment). This provides a top-down and a bottom-up radial search. The exponential complexity growth can be fixed by using better, even slightly intelligent crawling algorithms, adapted from web-crawling techniques. A time to live needs to be set adaptively.
- References to high-frequency web resources can be cached on all nodal network elements, viz. routers, bridges, switches and hubs. The terminal link elements possess the densest cache tables while intermediate entities possess only higher frequency results.
- The network is now constantly in a transitory phase updating reference tables at routes. Learning algorithms develop condensed higher priority digests at switches higher up in the hierarchy.

Improvement in performance may be achieved by adapting [16] web-crawling algorithms and [18] search methods to the proposed elastic topology. The proposed scenario is illustrated graphically in Fig. 2.2.

2.2.2 Smart Home Integration

The popularity of home and industrial automation has been increasing in recent years due to much higher affordability and simplicity though smart devices. The provision of remote accessibility has taken automation and security to a whole new level. For instance, a home automation system integrates electrical devices in a house with each other. The techniques employed in home automation include those in building automation as well as the control of domestic activities, such as home entertainment systems, houseplant and yard watering, use of domestic robots, centralized control

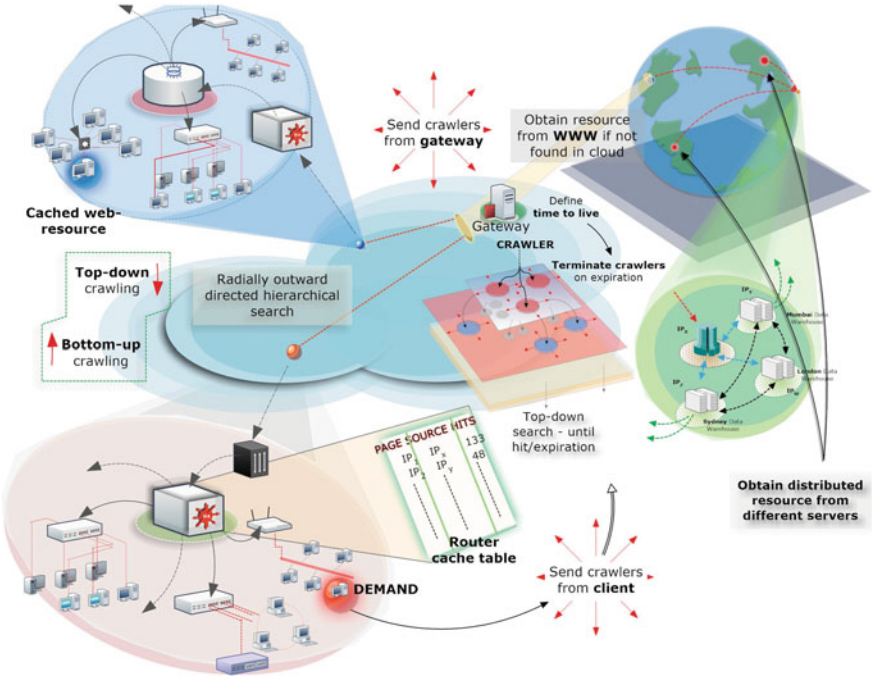

Fig. 2.2 An intelligent data caching scenario based on design postulates of *GSCC* wherein a particular users (*red colour coded*) webpage request is satisfied by another device (*blue colour coded*) in its vicinity, thereby avoiding the *WWW*. It relies on the fact that over time, the network learns the locations popular web-resources cached on the network, and is essentially termed as downloading the *WWW*. A very rudimentary technique has been depicted. **a Heuristic search**: release crawlers from both ends (**gateway-top-down** radially outward and **user-equipment-bottom-up** radially inward). The exponential complexity growth can be fixed by using a better, even slightly intelligent crawling algorithm; adapted from web-crawling technique. A time to live needs to be set adaptively; **b** References to high-frequency web resources can be cached on all nodal network elements, viz. routers, bridges, switches and hubs. The terminal link elements possess the densest cache tables while intermediate entities possess only higher frequency results; **c** The network is now constantly in a transitory phase updating reference tables at routes. Learning algorithms develop condensed higher priority digests at switches higher up in the hierarchy

of lighting, *HVAC* (heating, ventilation and air conditioning), appliances, security locks of gates and doors and other systems, to provide improved convenience, comfort, energy efficiency and security. True incorporation into the cloud necessitates that devices be connected through a computer network to allow control by a personal computer/mobile tablet device, and allow remote access through the Internet. Through the integration of information technologies with the home environment, systems and appliances are able to communicate in an integrated manner which results in convenience, energy efficiency and safety benefits, and fosters an intelligent home environment. We consider a case where super-clouds facilitate secure offloading of data logs from smart home systems.

A smart home automation system typically capacitates home owners to monitor and control equipment, utilities and security in a closed space. With a robust sensor network—spanning the entire house/apartment block—monitoring timely changes in equipment performances and surrounding environment, large amounts of data logs are generated and offloaded onto the network. The running footage from safety and security systems—cameras, temperature sensors, sprinkler systems and distributed-locking—impose a stiff load on the storage requirement. Notably, the data stream generated is almost constant in size and more often than not demands large amounts of space for relatively short periods of time. Data that transcends the rest, sporadic in nature, may be stored inexpensively over the local cloud of networked devices rather than on a distant dedicated data storage hub, accessing which repeatedly is cumbersome. Further, relevant data and footage are made available to networked computers and tablet devices; thus home-owners stay updated remotely. The local cloud spanning networked equipment within say, the apartment block, may additionally feed the data logs to the security office for perpetual vigilance. Wavelet-based algorithms providing temporal, spatial and *PNR* scalability are utilized to improve efficiency and accessibility of offloaded data, particularly videos and images [15, 17, 19, 20]. Thus, true smart-home integration can be achieved rather inexpensively sans a service provider or systems maintenance specialist. It can be taken to the extent that even basic lighting and switchboards are automated and intelligence is incorporated therein by a networked system.

2.2.3 Data Security—*Locally Distributed Storage*

While a traditional data-center based cloud leverages economies of scale to optimally store increasing amounts of data, the centralized storage often comes at the price of frequent data-security compromise. Security and threat aversion are critical fields of study within cloud computing, and strong emphasis is being laid on them [3, 21]. However, the community is still far from recognizing a robust and scalable solution to the problem. On the other hand, devices networked through a *GSC* paradigm, in an attempt to increase self-reliance and circumvent usage of data-centers, store data on the local cloud. While this may necessitate the storage of multiple instances of the same data and compromise *QoS* a bit, it nevertheless helps one capitalize on the locally available storage hardware. A consequence, fortunate however, is the fact that this inherently upsurges the security of stored data.

To understand this better, consider a scenario as shown in Fig. 2.3 wherein a 100 GB data file needs to be stored on a local cloud. The file is encrypted, broken down into smaller entities and multiple instances of each packet are stored randomly across the local cloud. Based on the confidentiality of data, symmetric key schemes such as *AES 128/192* or weaker schemes such as *DES* may be used for encryption. The encrypted file is split up and distributed across the network. An evaluation of general access and availability trends of devices on the cloud can help decide how many instances of each data packet are to be stored. The encryption scheme, data

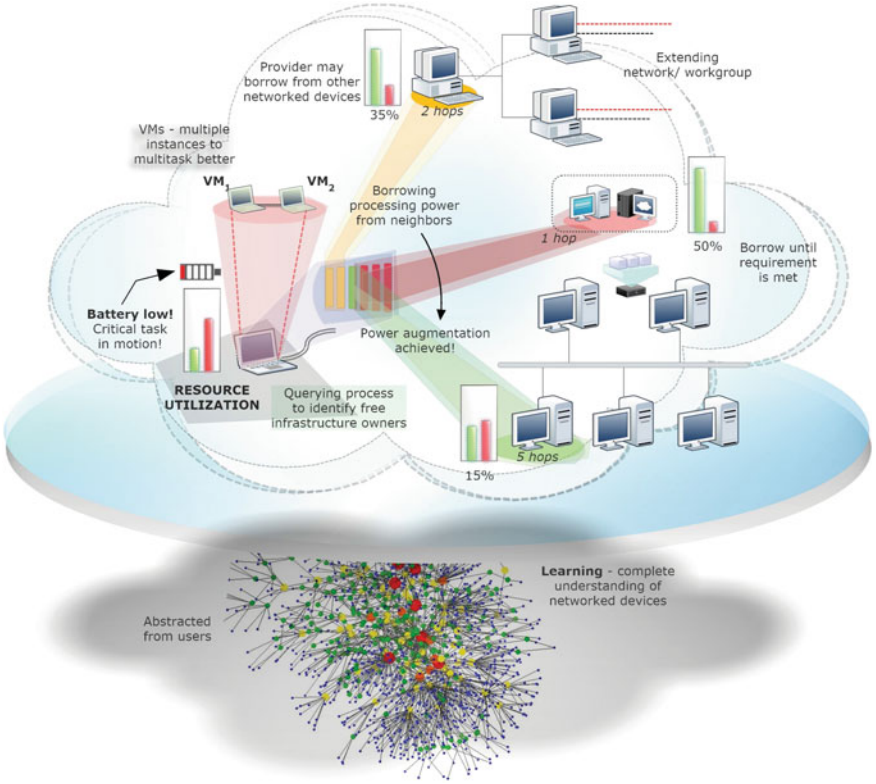

Fig. 2.3 A power augmentation and processor sharing scenario where a user boots into a device and performs certain operations. A significant processing power boost can be achieved by farming out pieces of data and processing them in parallel over co-operative peers on the network. The user is abstracted from the complexity of grid computing and parallel processing and just sees a significant leap in processor speed. An elaborate learning algorithm constantly maps networked devices and acts as a facilitator for services. Further, multiple virtual machines on devices offer parallel computing capacity and deployment of processor sharing. This symbiotic computing power improvement also manifests indirectly as a battery power augmentation, essential for portable devices and generally beneficial

fragmentation and distributed storage collectively enhance the overall security over the existing hardware. Thus, even if few of the hosts on the cloud network were to be compromised, the data remains inaccessible to the attacker and hence secure.

2.2.4 Incorporating Greenness

The energy consumption predicament in information technology equipment has been receiving increasing attention in recent years and there is growing recognition of the

need to manage energy consumption across the entire information and communications technology sector [1, 4–6, 13, 14, 17]. To keep pace with the rapidly growing demand for cloud-services, huge investments are being made towards the development and quick deployment of cloud infrastructure. As a consequence, energy-intensive data-centers are cropping up incessantly around the world. Data-centers and the transmission and switching networks in the Internet account for a significant fraction of total electricity consumption in broadband-enabled countries. In addition to the obvious need to reduce the greenhouse impact of the communications sector, this need to reduce energy consumption is also driven by the engineering challenges and cost of managing the power consumption of large data-centers and associated cooling. Against this, cloud computing will involve increasing size and capacity of data-centers and of networks, but if properly managed, cloud computing can potentially lead to overall energy savings.

While it is important to understand how to minimize energy consumption in data-centers that host cloud computing services, it is also important to consider the energy required to transport data to and from the end-user and the energy consumed by the end-user interface. Studies of energy consumption in cloud computing have focused only on the energy consumed in the data-center [5, 6, 11]. While this accounts for the major chunk of energy consumed, a more comprehensive analysis is required to obtain a clear picture of the total energy consumption of a cloud computing service and understand the potential role of cloud computing to provide energy savings.

2.3 Design Postulates for GSCC Systems

2.3.1 Virtualization

A cloud platform can be either virtualized or not. Virtualizing the cloud platform increases the availability of resources and the flexibility of their management (allocation, migration, etc.). It also reduces the cost through hardware multiplexing and helps energy saving. Virtualization is then a key enabling technology of cloud computing. System virtualization refers to the software and hardware techniques that allow partitioning one physical machine into multiple virtual instances that run concurrently and share the underlying physical resources and devices.

2.3.2 Abstraction

While abstraction may traditionally refer to concealing details from the user, here, it is used in the sense that users are entitled to a significantly larger chunk of IT resources and computing power than is visible physically. Simply put, this technique will not only abstract the unnecessary technical clutter behind services, but also offer

a huge leap in device performance and user experience. Coupled with learning and growing self-awareness, networked devices of the future are expected to perform at par with small-scale data-centers.

2.3.3 Distributiveness

The reliance of cloud-architectures on centralized data-centers, which require constant maintenance, data transportation and conditioning, makes them restrictive in nature. This debilitating characteristic is central even to the web technology wherein web servers are often centralized and remote. Systems of the future ought to promote distributiveness as a leading tenet in approaches and architectures. For instance, clouds could do with departure from the data-center model followed by migration to an agent-based model. In the agent-based model, servers and clients are undefined, rather dynamically defined based on demand and supply of resources within the cloud. At any instance, both servers/service-providers and clients are sporadically distributed. This is expected to yield a significant security improvement in data storage and minimize reliance on data-centers.

2.3.4 Greenness

Fragility of the environment and its heightened sensitivity towards emission of pollutants such as carbon-dioxide is pressurizing the scientific community to embrace greenness as a core characteristic in the development of futuristic systems. Systems such as the proposed distributed cloud are expected to reduce energy consumption significantly. Data-centers require immense amounts of brown-energies to sustain 24-h operation and back-forth data transfers around the globe. The data-center model doesn't appear to be very green intuitively, thus departure from the data-center model will inadvertently be a leap towards achieving greenness. To augment energy savings achieved by the new agent-based architecture, systems of the future need to develop and utilize inexpensive green technologies.

2.3.5 Symbiosis

Productive interactions between two or more devices, usually over significantly large periods of time, qualify as symbiosis. Symbiosis necessitates the availability of friendly architectures, protocols and perpetual learning. Symbiotic systems are expected to co-exist and co-operate, thus making most of their shared pool of resources. Fairness, acquired through equitable resource allocation and smoothness via increased interaction and self-awareness will improve productivity and

cooperation. A caveat though, is that typical selfish/scavenging-based approaches need to be mitigated.

2.3.6 Pervasiveness/Ubiquity

With the ubiquity of wired and wireless networks, it is not too hard to imagine a perpetual-access cloud anywhere, capable of elastically accommodating incoming and outgoing users. Provisioning almost-omnipresence, a feature often discussed, entails complete self-awareness (knowledge of the system state at any point), aggressive detection of incoming and outgoing users and elastic/friendly protocols. This virtue is expected to expand the super-cloud to all networked devices—portable and static, intelligent and stupid—and propagate ease of access across the cloud. While the data-center model offers better *QoS* and departure from the model poses significant risks of lower *QoS*, development on pervasive technologies only facilitates a more cloud-like on-demand access scenario. A progressive improvement in service quality is expected to transcend a pervasive and adaptive architecture.

2.3.7 Integration

Seamless administration of networks, virtual devices and platforms while elastically accommodating a growing number of devices and tasks into the cloud necessitates smooth integration of systems (smart and otherwise) and exploitation of common ground between networked entities. For instance, developing home automation systems in-sync with evolving cloud standards or more futuristic standards in general ought to be the premise. The integration process should promote accessibility, backward-compatibility, cross-platform movement, seamlessness and concomitant use of multiple platforms among other agenda.

2.3.8 Unification

Ongoing parallel research in IT and hardware, around the globe, is imposing variable methods to achieve similar technological goals. With the perpetration of devices running different technologies, networks using different hardware and protocols/standards and fast-paced evolution symbiosis in clouds necessitates unification of architectures and protocols to some extent. The rapid pace at which technological progress is being achieved, and the visible time-lag between that and adoption of the technologies, demands backward compatibility spanning multiple generations. Unification, achievable through standardization of technology/protocols, improving adaptability in hardware and software and time-tested cognition, has the potential to

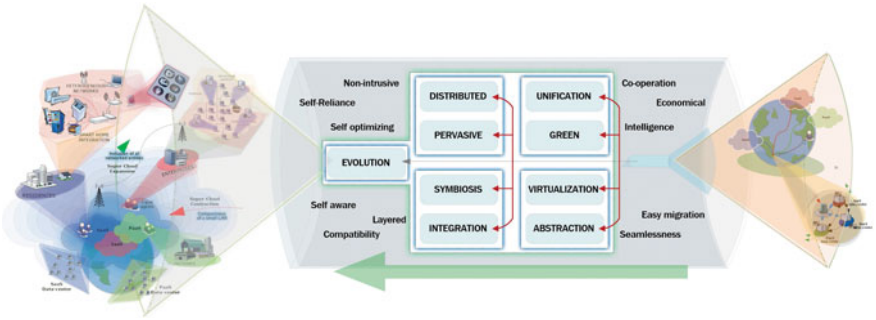

Fig. 2.4 Proposed design postulates for systems of the future

transform the cloud architecture into a more accessible, plug and play model, thus ensuring a multi-fold growth in the user base.

2.3.9 Evolution

Systems of the future need to keep up with the rapid growth in user demands and technological requirements of the age. Persistent migration of users, elastic nature of networks and ever-growing resource pools demand systems to learn constantly and self-optimize resources, addressing, networks, hardware and software. Self-awareness, which transcends time-tested cognition and optimization, may expedite realization of the cloud as a self-reliant, service-providing entity that demands minimal maintenance and user intervention. Not much unlike Darwinian evolution, the evolutionary nature of systems of the future will propagate only the greenest, smoothest, fastest and hopefully fairest architectural features; complete resource recycling and improving *QoS*, greenness and accessibility will serve as the primary guidelines (Fig. 2.4).

2.4 Architectural Design of GSCC Paradigm

This section discusses *GSCC*, a symbiotic and distributed communication paradigm for heterogeneous networks. The green and adaptive approach entails the simultaneous use of multiple communication interfaces, enabling efficient resource utilization. The judiciously designed architecture allows multiple users to access multiple mediums concomitantly, for both uplink and downlink, via virtualized communication ports and Internet Protocol (*IP*) schematic. Principal theoretic feasibility of the hypothesis is established by the linear increase in communication capacity, with minimal energy requirement. Promising simulation and experimental results of a static

scenario are shown, with User Equipment's (*UE*s) connecting to multiple Communication Mediums (*CM*s).

Spectral efficiency and increased throughput in wireless communications, is a topic of paramount importance owing to the substantial increase in user demands [20]. Easy and affordable availability of communication devices like smart mobile phones, laptops, tablets etc., has resulted in its proliferation and subsequent exponential surge in radio traffic [1]. The next quality of service leap is fundamentally expected to come from improvements in network topologies, cooperative communication and virtualization schemes, the amalgamation of cognitive heterogeneous networks and standardization of protocols on such networks [1, 3, 20–22]. Imagine a scenario where a smart phone user makes a video call requiring a capacity of 6 Mbps for uplink and downlink. The user is equipped with unlimited *WLAN* and Long Term Evolution (*LTE*), with each of these mediums providing a capacity of 4 Mbps respectively. In such a case neither of the mediums can provide the required capacity on a standalone basis. Scrutinizing the scenario, one realizes that if simultaneous usage of both the mediums is allowed the desired throughput can be achieved. Ubiquitous access to all available networks thus is the key for the end users to have guaranteed quality of service. The proposed paradigm attempts to integrate this thought evolving a green symbiotic heterogeneous network.

To the best of our knowledge, existing prototypes of heterogeneous networks only concentrate on throughput improvement on the uplink. Communication, however, is a two-way process with the downlink being just as critical as the uplink. In fact, more often than not, downlink carries a larger throughput demand as compared to the uplink [3]. Hence, a significant overall throughput increase necessitates an improvement in both, downlink and uplink data rates. Further, in light of greenness of recent emerging trends and governmental regulations, this too needs to be energy efficient [11, 14]. Consequently prevention of energy drain and complementing the lifetime of battery operated devices like cellular phones, *PDA*s, tablets, etc., forms an essential criterion in the architectural design [11]. As a part of the proposed *GSCC* paradigm we make the following novel contributions in a distributed and hierarchical manner:

1. Proposed is a paradigm, which enables *UE*s to simultaneously utilize all Communications Links (*CL*s) in the vicinity, where it is authorized to.
2. A virtualized communication port and *IP* schematic is proposed, which enables both uplink and downlink communications via all links in a distributed and symbiotic manner.
3. We establish a cognitive decision function and the theoretic networking capacity within the operational constraints and identify the power dynamics of the proposed paradigm.

It is evident that communication devices of the next generation are embedded with multiple radios and technologies. A single smartphone might have the ability to connect to varied *CM*s like cellular *LTE*, *WLAN*, Ethernet, Bluetooth, etc. [2, 5] While concepts like *MIMO*, cognitive radio, etc. enable spatial diversity gain, efficient spectrum utilization and increased *QoS*, their operations are limited to a singular *CM* [18, 19]. Furthermore, currently the limitations exist, where a user is able to only

connect to one *CM* at a particular time. For instance, if a user is connected to *WLAN*, the cellular *LTE* connection for data communication is rendered idle or stopped. The traditional approach prioritizes the available *CM*s based on their *QoS* and picks the best performing one. While it is logical to route majority of the capacity through the strongest medium, the other available mediums, which lie idle, could also be utilized in an energy-efficient way to increase throughput. Alternatively, in an antonymic scenario where high data rate is not required but power efficiency is desired, then only the *CM*s which offer least power-scavenging may be used. Hence, our motivation arises from embedding intelligence in the communication scenario, wherein the *UE* decides cognitively on the usage of *CL*s available to it, for maximizing its throughput and minimizing its operational power dynamics. The approach is coined as Green Symbiotic Cloud Communications (*GSCC*).

Consider a generic communication scenario as depicted in Fig. 2.5a, wherein a user with a smartphone is establishing a call to a user having a laptop. The cellular user has authorization to 4 access points while the laptop user has access to 3 at its end. With the proposed schematic of *GSCC*, both the devices will utilize all the available *CL*s by sending fractions of their data through cognitive splitting to achieve maximum throughput and optimum power usage. This multicast communication is achieved over standard communication protocols like *TCP/IP*, *UDP*, etc., and hence no change in protocol structure is required. In the following subsection, we lay out the basic process flow of realizing *GSCC*.

2.4.1 Process Flow

Consider a generic scenario where UE_a communicates with UE_b, and n' and m' *CL*s are accessible to UE_a and UE_b respectively. Let the *CL*s in the vicinity UE_a have public IP addresses $\{IP_{1a}, IP_{2a} \ldots, IP_{na}\}$ and those in the vicinity of UE_b have $\{IP_{1b}, IP_{2b} \ldots, IP_{mb}\}$. For each *CL* that a UE_x is connected to, it is associated with an *IP* address and port number pair $({}^xIP_i, P_i)$. Thus, the set of such identifier pairs being handled by the control layer in each UE_x is $\{({}^xIP_1, P_1), ({}^xIP_2, P_2) \ldots, ({}^xIP_n, P_n)\}$. As represented graphically in Fig. 2.5b, the flow entails 5 major steps:

1. *Scan/Identify*: The *UE*'s scan their environments, using existing provisions on radios/adapters of individual *CM*s. It identifies a list of all the *CL*s that are available and authorized to access.
2. *Connect*: Once all or most *CL*s are identified, the *UE* then establishes connection with all available *CL*s. We necessarily override the *UE*'s preset connection limitations by introducing a software-based control plane layer under the *IP* stack. This control plane comprises of a socket programming code that establishes multiple virtual communication ports. The control plane layer then enables the *UE*'s to handle multiple *IP* addresses for connectivity through different physical ports of *CM*s.

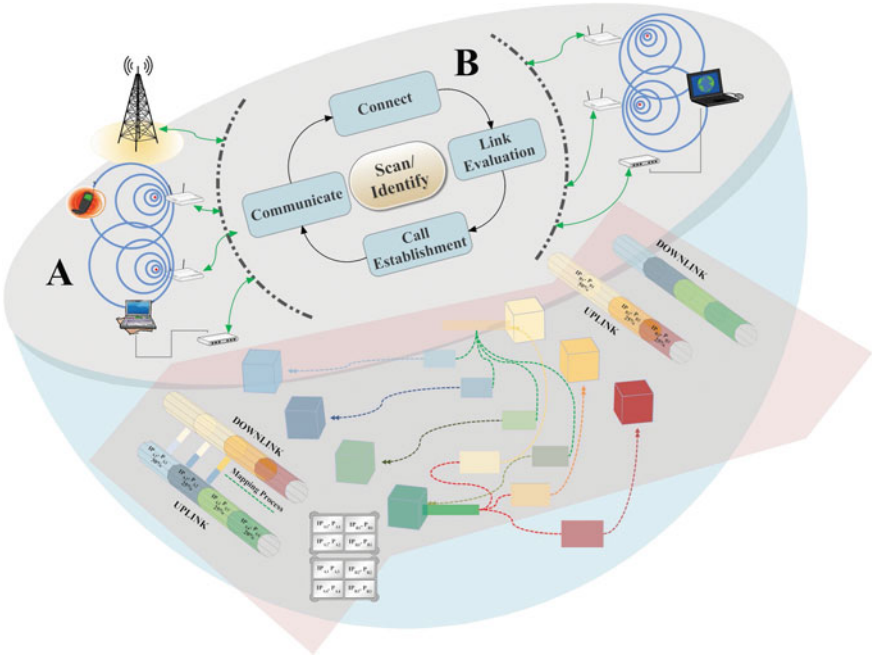

Fig. 2.5 The conceptual outlay of the proposed *GSCC* is shown where *A* shows multiple *CM*s connecting to multiple *UE*s. We further show a mobile phone connected to 4 *CL*s establishing a call to a laptop connected to 3 *CL*s. The connection process flow is outlined in *B*, which exchanges the virtualized port and *IP* information of both the *UE*s. The data flow and mapping for both uplink and downlink is shown in *C*, where the 4 *CL*s of UE_a map the data to 3 *CL*s of UE_b and vice versa. The final *IP* stack and port hash table is shown in *D*

3. *Link Evaluation*: The *CL*s are parametrically evaluated in terms of throughput, BER/data loss, energy consumption per unit data and the associated cost with each *CM* that the *UE* encounters. The cognitive splitting of the data is performed based on a decision function, which optimizes the throughput and power consumption.
4. *Communication Establishment*: In this phase, apart from handshaking, peers exchange instantaneous link cost information. This enables the peers to schedule packets on links and label them with appropriate destination *IP* addresses. Having knowledge of the connectivity situation at both ends allows users to schedule both uplink and downlink via multiple access points.
5. *Communicate*: Once the communication is established, data exchange takes place concomitantly over all the *CL*s using standard protocols such as *TCP/IP*, *UDP*, etc. The data reception is facilitated via physical ports of the *CM*s and combined together in the control layer plane of *GSCC*.

The process continues until a new *CL* is sighted or an existing one is lost. The new network is seamlessly stitched onto the architecture following the 5-step process flow.

2.4.2 Communication Process Flow

This section illustrates the working mechanism and communication flow of the proposed protocol. Consider a scenario where data transfer is to be initiated by a client to a remote server. Here, a link is considered as a connection established from the server to the client using a specific communication medium. The transport layer implements a cognitive decision function which predetermines parameters such as throughput, congestion, signal-to-noise ratio, power consumption, etc. From the calculated parameters, the transport is split into multiple, virtually parallel transport layers seen in Fig. 2.6 and splits the data packets among the different links with different sessions, each randomly over a predetermined distribution. As shown in Fig. 2.7, the virtual transport block on the server side has a logical one-to-one mapping with the virtual transport block on the client side.

The distribution of the packets transmitted over the network increases the challenge of reassembling at the server side. To counter this challenge, a buffer is introduced on both the server and the client side to store the unsorted data and reassemble the streamlined data segments at the application layer. The total number of links possible is a function whose parameters are the number of network adapters at both the client and the server side. So, the maximum number of logical links possible is $N_s \times N_c$, where N_s and N_c are the number of connections on the server and the client side respectively. This is due to the possibility of a many-to-many logical mapping. However, under the assumptions of a one-to-one mapping, the maximum possible throughput is the linear sum of throughputs of all individual links and maximum power consumption during the transfer of data is the combined power consumption of all links. Figure 2.7 describes the entire process.

With respect to the application, the virtual transport layer performs the functionality of transferring the data from the server to the client or vice versa.

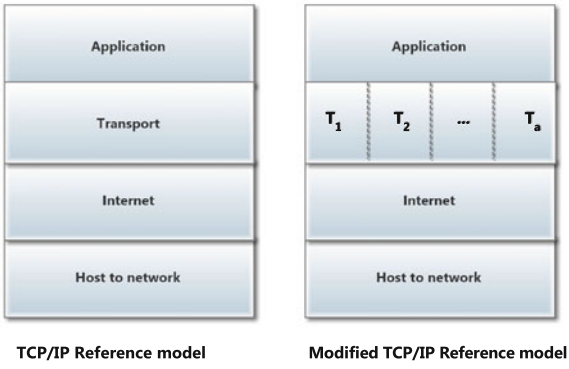

Fig. 2.6 Describes the different *TCP* layers and the modified *TCP* where we virtualize the transport layer to multiple transport layers embedding virtualization in the *GSCC* paradigm

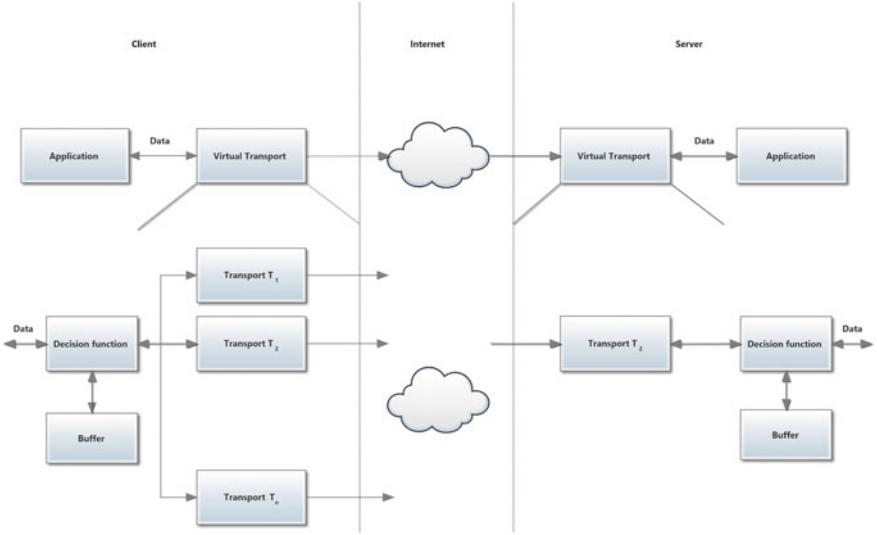

Fig. 2.7 Communication flow in the *GSCC* paradigm

Decision Function

It has a crucial role in the segmentation and reassembling of the data. The server divides the file into "n" different segments and transmission takes place simultaneously. Depending on the decision function, it will split the file in order to optimize total time taken and energy consumption, depending on the user requirements. Optimization of energy and throughputs is done for multiple logical connection scenarios. We classify the networks encountered as follows:

- Homogeneous network: When the links are homogeneous, the energy consumed for transferring the data is constant. Therefore, there is no energy factor involved during the splitting.
- Heterogeneous network: These links are different as they use different last hops such as *3G*, *Wi-Fi* and Ethernet. For instance, Ethernet usually consumes the least power and has maximum throughput. However, if the user still requires the data at a higher speed than offered by the Ethernet connection, the decision function acts accordingly by considering only maximizing throughput, though the energy consumed is much higher than by utilizing a low-energy medium like Ethernet. But if the user wants the system to run in a power-saving mode, then the decision function disables the *Wi-Fi* and *3G* networks, and uses only Ethernet.

The decision function is dependent on:

1. Capacity of the link $\{T_1, T_2, T_3, \ldots, T_n\}$
2. Power consumption of the link $\{P_1, P_2, P_3, \ldots, P_n\}$
3. Cost of the link $\{C_1, C_2, C_3, \ldots, C_n\}$

If there are 'n' network adapters for a machine, T_i, P_i and C_i denote capacity, power consumption and the cost of the link respectively for transferring data for the ith link.

Multiple Transport Layers

Data after division will be sent through different transport layers, which correspond to the available different network adapters. The modified *TCP/IP* stack will appear as shown in Fig. 2.6. The original transport layers $\{T_1, T_2, T_3, \ldots, T_n\}$ can be used for creating multiple sessions for transferring data in parallel, using the socket application programming interface (*API*). Thus, the virtual transport layers present in the stack are independent of the n different network mediums present, e.g. *Wi-Fi*, *3G/4G*, Ethernet, etc. as per the application point of view.

Session Generation

Sessions/connections are created depending on the decision function. Primary and secondary links are created according to the priority order of the connections depending on the decision function outputs. The client initiates a request and the server acknowledges this to initiate a handshake and generate sessions. For a session to be created, client creates a socket which will be used as the unique identity vector C_i containing {*clientip i, clientport i, serverip, serverport*} where *clientip i* and *clientport i* are the IP address and the port number allotted to the ith network interface respectively. Similarly, we have another vector S on the server which contains {*clientip i, clientport i, serverip, serverport*}. This unique vector is used by the sockets to differentiate different sockets. When client needs to download/upload data, this vector will be used in address headers to identify the client. The data will be sent along with the packet numbers in order to assemble at the other end. The problem occurs when data is sent in multiple sessions, as the data needs to be assembled correctly at the other end. Thus, after dividing the data into packets, they will have a local packet number which will be embedded in the data field of the packet frame and a sessions packet number which will be included in the packet number field of the packet frame, present in the header of the packet. Session packet numbers, which are in sequence, are present in the header of the packet as the firewall will start discarding the packets if continuity in the packet number is not preserved.

The parameters are same as that of the client but an additional parameter status is used to record the current status of the link. During the time a client initiates a request and the server acknowledges it, a handshake is initiated where a list containing the priority order of the links along with the throughputs, cost and power consumption of the links is transferred from the client to the server. For simplicity, assume that the

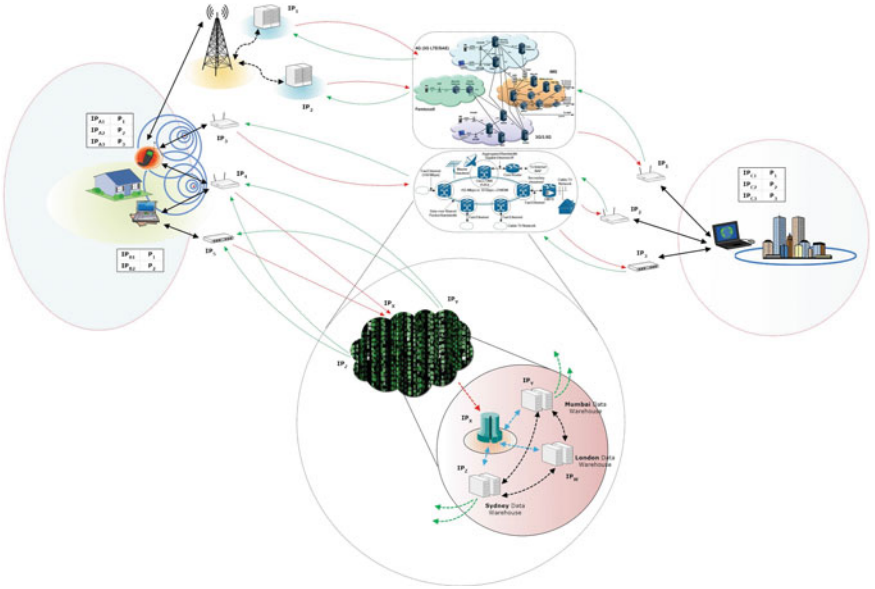

Fig. 2.8 *GSCC* case scenario: shows a smartphone user in a residential block communicating with a laptop user elsewhere (*right*). The cellphone user simultaneously uses 2 *Wi-Fi* links and one *LTE* connection while the laptop user (*to the right*) uses 2 *Wi-Fi* links and one Ethernet-based broadband connection. An *IP* scheme has been laid out and the uplink and downlink fractions have been displayed next to each link based on a rudimentary calculation. The figure also shows a second laptop user (*left*; residential block) communicating with a distributed data server based in London, Mumbai and Sydney. A similar *IP* scheme has been described for this scenario

file is split uniformly into n segments. Since all n segments are sent simultaneously, the ith segment of the data will travel on the ith link. Packet reassembling is done at the application layer. However, we will require an additional field data sequence in the packet frame which will store the segment number. The connections established are shown in Fig. 2.8.

Creating Multiple Virtual Sessions

In the current scenario, the server stores a session vector for defining the session. If multiple sessions are created from the same client but from a different network adapter (different *IP* address), the server will not be able to recognize whether the link is from the same client or a different one. Creating multiple sessions from the same client with n different network adapters connected can create multiple sessions and use the *GSCC* paradigm for increasing the throughput. Since each network adapter

has a specific *IP* address, the client has a pool of *n IP* addresses. Therefore, while initiating multiple connections to the server, we have to make certain changes to the default *TCP* protocol in order for the server to recognize that the same client is creating multiple sessions. We will define each connection from the client to the server as a communication link. We would require two different kinds of links:

(a) **Primary link**
(b) **Secondary link**

The primary link will first initiate a handshake with the server. This link is chosen by the decision function according to the priority order. This link is used for sending information regarding other available links from client, which can be used for establishing the connection. Apart from sending the information, it will act like a normal link sending the packet data.

Other links which are created serve as secondary links, which will be used for sending the packet data. The main functionality of the buffer is storing the data till the layer receives acknowledgement from the other side that the packet is received. If data is not received, re-transmission of the data is done directly from the buffer. It is also used for sorting the received data before sending to the application layer, i.e. if the server is sending/receiving data through multiple links, there will be a delay where some packets get missed. Thus, until the ordered sequence is received, data will be stored in the buffer which will be later assembled and sent to the application layer.

Data Transfer on Virtualized Communication Links

1. **Initialisation**: The decision block initializes by detecting the number of network adapters present on the machine. Decision function calculates the priority of links using the parameters such as throughput of the link, user-defined power consumption and cost of the link.
2. **Prioritizing the links**: The decision block creates a table containing the possible available links to the server which are sorted according to the priority. These are stored in an array using *IP*'s and ports ordered according to the priority.
3. **Creation of link**: The highest priority link will be created first and considered as the primary link. While handshaking, it will discover whether the server is *GSCC*-enabled. If the server is *GSCC*-compatible, remaining connection links will be created according to the priority and are considered as secondary links. Once these connections are established, they are executed in parallel using the socket *API*.
4. **Multiple sessions handling**: The server has a table of *IP* addresses, port numbers, primary session ID's and the status (connected/disconnected) information of each

link. The server needs to know all the *IP* addresses linked up with the client to recognise the connections as arising from the same client. Once the primary link is created, it transfers the priority table to the server. Primary session ID is used as a unique key for identifying the client. Once the secondary connections are established, status indicates connected' in the table. During uplink, the server receives data from all the connections. Using the table, it will assemble the packets in the buffer and send them to the application layer. During downlink the server stores some data in the buffer and sends the data to all the links, which are having connection status as connected' in the table for that particular client.

5. **Closing the link**: Any link can send an '*ack*' to close the session. Closing the session will clear all the *IP* and port addresses attached to the client primary session ID.

Connection Errors

Connection errors are due to connection breakages and are detected when data cannot be transferred through that particular link. We consider the following connection error scenarios:

- **The primary link is broken**: When the primary link is broken, the client can no longer update the server table. So, in order to remove such error, the server chooses the next connected link from the priority table as the primary link. Same is the case at the client side; it will consider the next connected connection as the primary link from the priority table. Connection status of the link is updated to disconnected' and the decision block stops sending data through that link.
- **The secondary link is broken**: If the secondary link is broken, the connection status of the link is changed to disconnected' and the decision block stops sending data through the link.
- **Broken link is re-created**: Client keeps on trying to establish the broken link and if it succeeds in establishing a broken primary link, both client and server assume it again to be the primary link, change its status to connected' and restore the data transmission through it. The existing primary link is again converted to secondary status. When a secondary link establishes connection, the status of the link will be changed from disconnected' to connected' and the data transmission through the link is restored.

The following chapter develops on the architectural base and the concept of the *GSCC* paradigm to establish a theoretical framework validated by experimental results.

Algorithm 1: Client side process

```
Data: Initializes port numbers and IP addresses of the client
Input the desired location;
while connection is not established do
    check for host;
    if Host is unreachable then
        initiate random backoff;
        attempt for reconnection;
    else
        Create an end communication socket;
        Connect to host using request and wait for acknowledgement;
    end
end
Allocate Memory to the buffer and keep application on standby;
while Check for connectivity do
    initiate data transfer;
    check the status;
    if connection terminated then
        update the server and wait until connection re-established;
    else
        check transfer status;
    end
end
```

Algorithm 2: Server side process

```
Data: Initializes port numbers and IP addresses of the server
while Listen for incoming connections do
    if Connection using GSCC protocol then
        Accept initialization headers;
        Connect to GSCC client;
        Create database containing IP addresses and Port numbers;
        Create end point communication segments;
        initialize buffer and segment the data;
    else
        Create an end communication socket;
        Connect to host using request and wait for acknowlegement;
    end
end
Select decision criteria;
Commence data transfer;
Check and update the status of the transfer;
Terminate connection when data transfer complete;
```

References

1. A. Iosup, S. Ostermann, M.N. Yigitbasi, R. Prodan, T. Fahringer, and D. H J Epema. Performance analysis of cloud computing services for many-tasks scientific computing. *Parallel and Distributed Systems, IEEE Transactions on*, 22(6):931–945, 2011.
2. P. Kalagiakos and P. Karampelas. Cloud computing learning. In *Application of Information and Communication Technologies (AICT), 2011 5th International Conference on*, pages 1–4, 2011.
3. M. AbdelBaky, M. Parashar, Hyunjoo Kim, K.E. Jordan, V. Sachdeva, J. Sexton, H. Jamjoom, Zon-Yin Shae, G. Pencheva, R. Tavakoli, and M.F. Wheeler. Enabling high-performance computing as a service. *Computer*, 45(10):72–80, 2012.
4. C. Papagianni, A. Leivadeas, S. Papavassiliou, V. Maglaris, C. Cervello-Pastor, and A. Monje. On the optimal allocation of virtual resources in cloud computing networks. *Computers, IEEE Transactions on*, 62(6):1060–1071, 2013.
5. B. Kantarci and H.T Mouftah. Designing an energy-efficient cloud network [invited]. *Optical Communications and Networking, IEEE/OSA Journal of*, 4(11):B101–B113, 2012.
6. N. Xiong, W. Han, and A. vandenBerg. Green cloud computing schemes based on networks: a survey. *Communications, IET*, 6(18):3294–3300, 2012.
7. Frederica Darema. Grid computing and beyond: The context of dynamic data driven applications systems. *Proceedings of the IEEE*, 93(3):692–697, 2005.
8. C.J. Sher Decusatis, A. Carranza, and C.M. DeCusatis. Communication within clouds: open standards and proprietary protocols for data center networking. *Communications Magazine, IEEE*, 50(9):26–33, 2012.
9. Kwang Mong Sim. Agent-based cloud computing. *Services Computing, IEEE Transactions on*, 5(4):564–577, 2012.
10. D. Talia. Clouds meet agents: Toward intelligent cloud services. *Internet Computing, IEEE*, 16(2):78–81, 2012.
11. Jagadish Ghimire and Catherine Rosenberg. Resource allocation, transmission coordination and user association in heterogeneous networks: A flow-based unified approach. *IEEE Transactions on Wireless Communications*, 12(3):11, 2013.
12. M. Bourguiba, K. Haddadou, I. El Korbi, and G. Pujolle. Improving network i/o virtualization for cloud computing. *Parallel and Distributed Systems, IEEE Transactions on*, pp(99), 2013.
13. FeiFei Chen, J.-G. Schneider, Yun Yang, John Grundy, and Qiang He. An energy consumption model and analysis tool for cloud computing environments. *Green and Sustainable Software (GREENS), 2012 First International Workshop on*, pages 45–50, 2012.
14. Renchao Xie, F. Richard Yu, Hong Ji, and Yi Li. Energy-efficient resource allocation for heterogeneous cognitive radio networks with femtocells. *IEEE Transactions on Wireless Communications*, 11(11):10, 2012.
15. M. Shiraz, A. Gani, R. Khokhar, and R. Buyya. A review on distributed application processing frameworks in smart mobile devices for mobile cloud computing. *Communications Surveys & Tutorials, IEEE*, PP(99), 2013.
16. Ismail Sengr Altingvde and zgr Ulusoy. Exploiting interclass rules for focused crawling. *Intelligent Systems, IEEE*, 19(6):66–73, 2004.
17. M. Lemay, Kim-Khoa Nguyen, B. St. Arnaud, and M. Cheriet. Toward a zero-carbon network: Converging cloud computing and network virtualization. *Internet Computing, IEEE*, 16(6):51–59, 2012.
18. Gaeta R. and Sereno M. Generalized probabilistic flooding in unstructured peer-to-peer networks. *Parallel and Distributed Systems, IEEE Transactions on*, 22(12):2055–2062, 2011.
19. X. Chen, Z. Dai, W. Li, Y. Hu, J. Wu, H. Shi, and S Lu. Prohet: A probabilistic routing protocol with assured delivery rate in wireless heterogeneous sensor networks. *IEEE Transactions on Wireless Communications*, pp(99):8, 2013.

20. Stefan Brueck. Heterogeneous networks in lte-advanced. In *Wireless Communication Systems (ISWCS), 2011 8th International Symposium on*, page 5, 2011.
21. Hsin-Yi Tsai, M. Siebenhaar, A. Miede, Yu-Lun Huang, and R Steinmetz. Threat as a service?: Virtualization's impact on cloud security. *IT Professional*, 14(1):32–37, 2012.
22. D.M. Shila, Yu Cheng, and T. Anjali. Capacity of cooperative wireless networks using multiple channels. In *Communications (ICC), 2010 IEEE International Conference on*, 2010.

Chapter 3
Green Symbiotic Cloud Communications—Theory and Experimentation

Abstract The cloud communications concept supported by the Green Symbiotic Cloud Communications (GSCC) paradigm is theoretically and mathematically modelled in this chapter. The aim is to maximize the throughput of communications infrastructure through all connected communication links with characterization of corresponding operational power dynamics. The GSCC paradigm is subsequently verified by both simulation and experimental results. The simulation is performed by a GSCC simulator, to accommodate the joint operation of all existing communication standards (viz. *LTE* eNodes, Wi-*Fi* Access Points etc.). The concept of cloud communications is physically realized and experimentally tested by developing a working prototype of GSCC paradigm. The paradigm is tested with incorporation in all the existing communication standards. The results demonstrate the basic advantage offered by the GSCC paradigm—unified and integrated usage of all available resources in a virtualized and abstract environment.

Theoretical modelling of the *GSCC* paradigm is performed with respect to maximization of throughput through all connected communication links. Consider the generic scenario of *GSCC* as depicted in Fig. 2.8, wherein the *UE*s are connected to both wired and wireless *CL*s. Each access point serves as an ingress or egress for the related traffic of the *UE*s in its coverage area. The communication links further may be equipped with multiple radios each individually operating on orthogonal channels. The *UE*s and access points combine the incoming packets algebraically using random linear network coding (*RNC*) as described in [1, 2]. Assume that the *CM*s transmit indiscriminately in multicast mode and the *GSCC* architecture operates in a synchronous time-slotted mode. Thus, we have a random extended network with access points adhering to a Poisson point distribution of intensity n on a square $S(n) = [0, \sqrt{n}] \times [0, \sqrt{n}]$ [1]. We model the *GSCC* architecture as a directed hypergraph $G = (\mathrm{E}, \mathbb{N})$, where E represents the number of source *UE*s and $\mathbb{N}$ the number of *CM*s representing the hyperarcs. We denote $Q(i)$ as the number of radios that each *CM* has with $i \in N$. Given the *GSCC* directed graph $G = (\mathrm{E}, \mathbb{N})$, the aim is to establish U unicast connections from each E to the set of destination *UE*s $\mathbb{E} \subset \mathbb{N}'$, where $\mathbb{N}'$ are *CM*s at destination *UE*s. In the downlink scenario, the source becomes the destination and vice versa. We denote the traffic demand of connection m by

H.D. Mustafa et al., *Green Symbiotic Cloud Communications*,
DOI 10.1007/978-981-10-3512-8_3

r_m, $m = 1, 2, \ldots, T$, and determine the transmission rate based on the *generalized physical model* [1] also referred to as the Gaussian Channel model, which offers the best approximation in both wired and wireless *CM*s.

3.1 Capacity Maximization

We define the capacity vector $C_{n,r_m} = (C_{n,r_1}, C_{n,r_2}, \ldots, C_{n,r_t})$ as feasible if there exists a transmission scheme for sending packets over a distributed network such that the nth destination *CL* can receive all its packets at least min $\left(R_{n,r_m}\right)$ bits/sec. Considering a unicast rate vector, we can define the total throughput of such a feasible capacity vector as a rate vector $R_{n,r_m}(n) = \sum_{i=1}^{m} r_{n,r_i}$ and its corresponding minimum and maximum values as $R^L_{n,r_m} = \min_{(E_i \in \mathbb{E})} r_{n,r_i}$ and $R^H_{n,r_m} = \max_{(E_i \in E)} r_{n,r_i}$ [2]. Hence, under the *generalized physical model* with each *CM* transmitting with a power P_i, two *UE*s can establish a communication link in a channel of bandwidth B_i and rate $R_{m_i,m_j} = B_i \log\left(1 + \frac{P_i \cdot l(m_i,m_j)}{N_o + \sum_{m_i \in \frac{A_m}{r_i}} P \cdot l(m_i,m_j)}\right)$ where N_o is the ambient noise power, A_m is the nodes that transmits when *CL* r_i is scheduled and $l(m_i, m_j)$ is the power attenuation function.

For multiple *CM*s to communicate, each *CL* requires a common channel with each *UE* and to stay within each other's range. Even though we assume that access points of different *CM*s do not communicate amongst themselves, they may still interfere with each other's communication [2]. The protocol model of interference defines the interference range to be greater than the communication range by a factor $\beta \geq 1$ [1, 2]. This interference between the access points is modeled as a conflict graph $\mathbb{H} = (V, \mathbb{A})$, where $v \in V$, corresponding to a $CL(i, j) \in N$, is the set of vertices and $\mathbb{A}$ the set of edges. An edge $(u, v) \in \mathbb{A}$ if there is interference between the *CL*s of the corresponding access points $\forall u, v \in V$. We can now define the capacity on CL m as the minimum of all the capacities of its substituent links as shown in (3.1)

$$C_{n,r_i}(m) = \prod_{j \in J} \left[1 - l(m_i, m_j)\right] \min_{j \in J} R_{m_i,m_j}(m) \tag{3.1}$$

Considering a unicast connection by a tuple (s, u_n, r_m), we associate in each $CL(i, j) \in \bigcup_{m=1}^{M} \mathbb{N}(m)$ an injection value $x_{(ij)}(m)$, which determines how many packets each $UE_i \in E(m)$ should send to each *CL*. It signifies the average rate at which packets are injected into a $CL(i, j)$ on channel m, i.e. $x_{(ij)}(m) = \sum_{k \subset j} z_{ijk}(m)$.

Suppose we establish a connection (s, u_n, r_m) with operating rate $R_n \cdot r_m$ with a threshold waiting time of $x_{ij}(m)/(R_n \cdot r_m)$ the transmission can be supported over a time period τ if

$$R_n \cdot r_m \leq \min_{\psi \in \Psi(s,u_n)} \left\{ \sum_{k \not\subset \psi} \sum_{(i,j) \in B(\psi)} \underset{m=1}{M} z_{ijk}(m) \right\} \tag{3.2}$$

where $\Psi(s, u_n)$ represents the number of paths on a *CL* m and $B(\psi)$ is the set of flow hyperarcs of the path $\psi \in \Psi(s, u_n)$.

$$B(\psi) \doteqdot \{(i,j) \in | \; i \in \psi \text{ and } j - \psi \neq 0\} \tag{3.3}$$

Hence, for establishing a transmission over all *CL*s, m, the capacity vector, should satisfy the constraints enlisted in (3.4)

$$\sum_{\{j|(i,j)\in B\}} \sum_{j\in J} C_{n,r_j}^{(m)} - \sum_{\{j|(j,i)\in B, i\in I\}} C_{n,r_i}^{(m)} = \begin{cases} R_n \cdot r_m & \text{if } i = s \\ -R_n \cdot r_m & \text{if } i = u_n \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

for all access points $i \in N$ and $n = 1, 2, \ldots, U$ and

$$\sum_{j\in K} C_{n,r_j}^{(m)}(m) \leq \sum_{\psi \subset J | \psi \bigcap K \neq 0} z_{ijB}^{(m)}(m) \tag{3.5}$$

where $z_{ijB}^{(m)}(m)$ signifies the average throughput in *CL* m. On aggregate, the throughput rate $z_{ijB}^{(m)}(m)$ for all *CL*s cannot exceed $x_{(ij)}(m)$ for a non-negative rate, i.e.

$$\sum_{m=1}^{U} z_{ijB}^{(m)}(m) \leq x_{(ij)}(m) \tag{3.6}$$

$$R_{m_i,m_j}(m) \geq 0 \tag{3.7}$$

For throughput on all *CL*s m, we have the total capacity limited by

$$0 \leq \sum_{m=1}^{U} \sum_{j\in K} R_{m_i,m_j}(m) \leq \sum_{m=1}^{U} C_{n,r_i}(m) \tag{3.8}$$

Considering the conflict graph $\mathbb{H} = (V, \mathbb{A})$ and taking its subset $Q \subset V$, all the access points $v \in Q$ correspond to the *CL*s m that interfere with each other. To account for this interference, it is essential that at most one *CL* be scheduled for transmission at any given time. Hence, for all $Q \in \mathbb{H}$ and $m = 1, 2, \ldots, M$, the sum of fractional *CL* use that interfere with one another should not exceed its capacity $C_{n,r_i}^{(m)}$, i.e.

$$\sum_{v_{ij}\in Q} \frac{x_{ij}(m)}{C_{n,r_i}(m)} \leq 1 \tag{3.9}$$

Assuming each access point of all *CM*s participates in utmost R_{n,r_m}^{H} simultaneous communications, any valid *CL* in any given time slot τ is interference-free if

$$\sum_{m=1}^{U} \sum_{i,J \in \mathbb{N}(m)} \frac{x_{ij}(m)}{C_{n,r_i}(m)} + \sum_{m=1}^{U} \sum_{j,I \in \mathbb{N}(m) | i \in I} \frac{x_{ji}(m)}{C_{n,r_j}(m)} \leq R^{H}_{n,r_m} \forall i \in \mathbb{E} \tag{3.10}$$

Combining the above operational constraints together, we formulate the following linear program:

$$\max \mathbb{C}_{n,r_m}$$

subject to

$$R_n \cdot r_m \leq \min_{\psi \in \Psi(s,u_n)} \left\{ \sum_{k \not\subset \psi} \sum_{(i,j) \in B(\psi)} \sum_{m=1}^{M} z_{ijk}(m) \right\}$$

$$\sum_{\{j|(i,j) \in B\}} \sum_{j \in J} C^{(m)}_{n,r_j} - \sum_{\{j|(j,i) \in B, i \in I\}} C^{(m)}_{n,r_i} = \begin{cases} R_n \cdot r_m & \text{if } i = s \\ -R_n \cdot r_m & \text{if } i = u_n \\ 0 & \text{otherwise} \end{cases}$$

$$\sum_{j \in K} C^{(m)}_{n,r_j}(m) \leq \sum_{\psi \subset J | \psi \bigcap K \neq 0} z^{(m)}_{ijB}(m)$$

$$\sum_{m=1}^{U} z^{(m)}_{ijB}(m) \leq x_{(ij)}(m)$$

$$R_{m_i,m_j}(m) \geq 0$$

$$\sum_{v_{ij} \in Q} \frac{x_{ij}(m)}{C_{n,r_i}(m)} \leq 1$$

$$\sum_{m=1}^{U} \sum_{i,J \in \mathbb{N}(m)} \frac{x_{ij}(m)}{C_{n,r_i}(m)} + \sum_{m=1}^{U} \sum_{j,I \in \mathbb{N}(m) | i \in I} \frac{x_{ji}(m)}{C_{n,r_j}(m)} \leq R^{H}_{n,r_m} \forall i \in \mathbb{E}$$

$$0 \leq \sum_{m=1}^{U} \sum_{j \in K} R_{m_i,m_j}(m) \leq \sum_{m=1}^{U} C_{n,r_i}(m)$$

3.1.1 Proposition I

$$GSHN_{cc} = M \times U \left(\sum_{(i,J) \in \cup_{m=1}^{M} \mathbb{N}(m)} |J| \left(1 + \frac{M-1}{2} \right) + |E| + U \right) + \left(\sum_{(i,J) \in \cup_{m=1}^{M} \mathbb{N}(m)} 2^{|J|} M(U+1) \right)$$

The best-case scenario when the paradigm operates on all *CL*s of all *CM*s is achieved when $|J| = |E|$, yielding a linear summation of joint capacities.

$$\mathbb{C} = \sum_{m=1}^{M} \mathbb{C}_{n,r_i} = \sum_{m=1}^{M} \sum_{i=1}^{U} C_{n_m,r_i} \tag{3.11}$$

3.2 Operational Power Dynamics

To characterize the operational power dynamics, for every user $u \in U$ we define a channel matrix $H_u = X_u \Sigma_u^{1/2}$, where X_u's represent complex Gaussian random variables with zero mean and unit variance and $\Sigma_u^{1/2}$ is the correlation between the signals transmitted by user u. For a single user system we can now represent (3.1) as

$$C_{n,r_i} = \max_{tr(Q_u) \le P_u} E\left[\log\left|I_n + HQH^{\dagger}\right|\right] \tag{3.12}$$

where $Q_u = E[y_u, y_u^{\dagger}]$ represents the covariance matrix of user u having a power constraint P_u. Taking the singular value decompositions of $\Sigma = \mathbb{U}_{\Sigma} \Lambda_{\Sigma} \mathbb{U}_{\Sigma}^{\dagger}$ and $Q = \mathbb{U}_Q \Lambda_Q \mathbb{U}_Q^{\dagger}$ with Λ_{Σ} and Λ_Q representing the diagonal matrices of ordered eigenvalues and $\mathbb{U}$ the unitary matrix. If we assume $\mathbb{U}_{\Sigma} = \mathbb{U}_Q$, (3.12) reduces to

$$C_{n_m,r_i} = \max_{\sum_{i=1}^{n_t} \lambda_i^Q \le P_u} E\left[\log\left|I_n + \sum_{i=1}^{n_t} \lambda_i^Q \lambda_i^{\Sigma} \mathbb{z}_i \mathbb{z}_j^{\dagger}\right|\right] \tag{3.13}$$

where $\mathbb{z}_i \in \mathbb{Z}$ are zero-mean, identity covariance Gaussian random vectors. Extending (3.13) to all *UE*s U we get (3.14)

$$C_{n,r_i} = \max_{\substack{\sum_{i=1}^{n_t} \lambda_{u_i}^Q \le P_{u_i} \\ u=1,2,\ldots,U}} E\left[\log\left|I_n + \sum_{u=1}^{U} \sum_{i=1}^{n_t} \lambda_{ui}^Q \lambda_{ui}^{\Sigma} \mathbb{z}_{ui} \mathbb{z}_{uj}^{\dagger}\right|\right] \tag{3.14}$$

Taking the derivative of Lagrange of (3.14) and applying the KKT conditions yields (3.15)

$$E_{ui}(\lambda^Q) = E\left[\frac{\lambda_{ui}^Q \lambda_{ui}^{\Sigma} \mathbb{z}_{uj}^{\dagger} \mathbb{A}_{ui}^{-1} \mathbb{z}_{ui}}{1 + \lambda_{ui}^Q \lambda_{ui}^{\Sigma} \mathbb{z}_{uj}^{\dagger} \mathbb{A}_{ui}^{-1} \mathbb{z}_{ui}}\right] \le \max \mu_u \tag{3.15}$$

where $\mathbb{A} = I_n + \sum_{u=1}^{U} \sum_{i=1}^{n_t} \lambda_{ui}^Q \lambda_{ui}^{\Sigma} \mathbb{z}_{ui} \mathbb{z}_{uj}^{\dagger}$ and $\mathbb{A}_{ui} = \mathbb{A} - \lambda_{ui}^Q \lambda_{ui}^{\Sigma} \mathbb{z}_{ui} \mathbb{z}_{uj}^{\dagger}$. The inequality in (3.15) is satisfied when $\lambda_{ui}^Q \ge 0$. Simplifying further by multiplying λ_{ui}^Q in the RHS and the LHS of (3.15) yields

$$\lambda_{ui}^{Q} = \frac{E\left[\frac{\lambda_{ui}^{Q}\lambda_{ui}^{\Sigma}\mathbb{z}_{uj}^{\dagger}\mathbb{A}_{ui}^{-1}\mathbb{z}_{ui}}{1+\lambda_{ui}^{Q}\lambda_{ui}^{\Sigma}\mathbb{z}_{uj}^{\dagger}\mathbb{A}_{ui}^{-1}\mathbb{z}_{ui}}\right]}{\sum_{u=1}^{U}\sum_{i}^{n_t} E\left[\frac{\lambda_{ui}^{Q}\lambda_{ui}^{\Sigma}\mathbb{z}_{uj}^{\dagger}\mathbb{A}_{ui}^{-1}\mathbb{z}_{ui}}{1+\lambda_{ui}^{Q}\lambda_{ui}^{\Sigma}\mathbb{z}_{uj}^{\dagger}\mathbb{A}_{ui}^{-1}\mathbb{z}_{ui}}\right]} P_u \triangleq f_{ui}(\Lambda^{Q}) = \frac{E_{ui}(\Lambda^{Q})}{\sum_{u=1}^{U} E_{ui}(\Lambda^{Q})} P_u \quad (3.16)$$

We adapt the algorithm as shown in [2] to obtain the optimum eigenvalues, which updates the eigenvalue of *CL* $u+1$ as a function of u using $u = (n+1) \bmod U$.

$$\Lambda_u^{Q}(n+1) = f_u\left(\Lambda_1^{Q}(n+1), \ldots, \Lambda_{u-1}^{Q}(n+1), \Lambda_u^{Q}(n), \Lambda_{u+1}^{Q}(n), \ldots, \Lambda_U^{Q}(n)\right) \quad (3.17)$$

The proof of convergence is omitted here due to its close similarity to the one in [2]. The power consumption thus increases marginally as seen in (3.16) due to operation of multiple *CM*s, which is justifiable considering the linear increase in capacity obtained in (3.11).

3.3 Experimental Evaluation

The *GSCC* paradigm is verified by both simulation and experimental results in the following subsections. Both simulation and experimental results confirm a win-win situation for the proposed design.

3.3.1 Simulation with Combined LTE *and* Wi-Fi

We initiate the experiment with the simulation of *GSCC* by modifying an open source *LTE* simulator, to accommodate the joint operation of *LTE* eNodes and *Wi-Fi* Access Points. The performance of an indoor *LTE* system operating in the *TV* white space band which is interfered by other transmitters is evaluated. The devices considered here are mainly home *eNodeB*'s communicating with mobile user equipment (*UE*, Mode I portable device). Four scenarios are studied:

1. Two neighboring cells and inter-site distance (*ISD*) of 60 m,
2. Two neighboring cells and inter-site distance of 10 m,
3. 7-cell cellular femto-geometry with *ISD* of 60 m and
4. 7-cell cellular femto-geometry with *ISD* of 10 m

The number of users achieving full capacity reduces with decrease in the *ISD*. Also, the scenario with 7 cells has lesser number of users achieving full capacity as compared to the one with 2 cells. This is due to the increased inter-cell interference. The variation is also studied with varying load. As the load increases, lower percentage of users reach full rate capacity.

We explore and modify an open-source *LTE* system level simulator developed by Institute of Communications and Radio-Frequency Engineering, Vienna University of Technology, Austria. This simulator, based on *MATLAB*, is published under a non-commercial academic use license. The key limitation of the system is that it simulates downlink only and hence only the downlink scenario has been evaluated for the time being. Certain sections of the code have been modified appropriately and over 300 simulations are performed with different environment parameter settings. Select results have been posted in Figs. 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 3.10, 3.11, 3.12, 3.13, 3.14, 3.15, 3.16, 3.17, 3.18, 3.19 and 3.20. A snapshot of the parameters within the simulator has been posted in Table 3.1, as presented in the source documentation.

There are nine sections of modifiable parameters:

1. **General parameters**: Frequency in which the system is operating, system bandwidth, *UE* creation and regions of interest, number of transmit antennas, number of receives antennas, *TTL*s, *UE* throughput tracking.
2. **Cache options.**
3. **Network layout and macroscopic path loss parameters**: Path loss models, coupling models and antenna transmit power.
4. **Shadow fading**: Neighbors, mean, standard deviation, inter-site shadow fading correlation.
5. **Micro-scale fading**: Pedestrian channels, vehicular channels, length of channel trace.
6. ***UE* settings**: Receiver noise figure in dB, thermal noise density in dB-m/Hz, number of *UE*s per sector, speed at which the *UE*s move in m/s.
7. ***eNodeB* settings**: Mean antenna gain, patterns, schedulers, power allocation.
8. **Uplink channel options.**
9. **SINR averaging algorithms.**

The average results of the simulation cases of the 300+ scenarios are enlisted in Figs. 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 3.10, 3.11, 3.12, 3.13, 3.14, 3.15, 3.16, 3.17, 3.18, 3.19 and 3.20. The first set of figures discusses the energy consumption only in the *LTE eNodesB* operational scenario and the results are substantiated in Figs. 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8 and 3.9. In the second case, we explore the joint operation of *LTE* and *Wi-Fi* access points with the results demonstrated in Figs. 3.10, 3.11, 3.12, 3.13, 3.14, 3.15, 3.16, 3.17, 3.18, 3.19 and 3.20. We notice that in the joint operational scenario, with the same amount of energy, the region covered is more than twice that of the case with only *LTE eNodeB*'s, thus proving the energy efficiency of the proposed paradigm (Table 3.2).

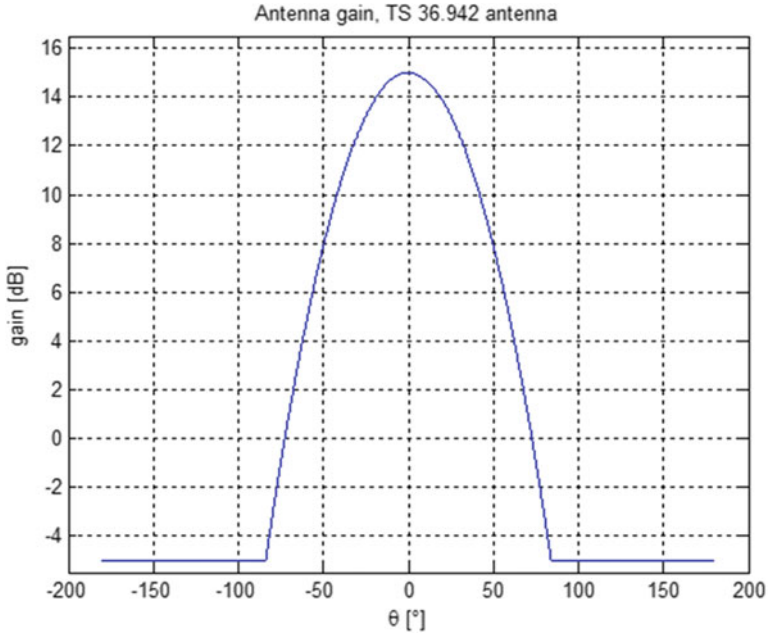

Fig. 3.1 Antenna gain TS 36.942 antenna

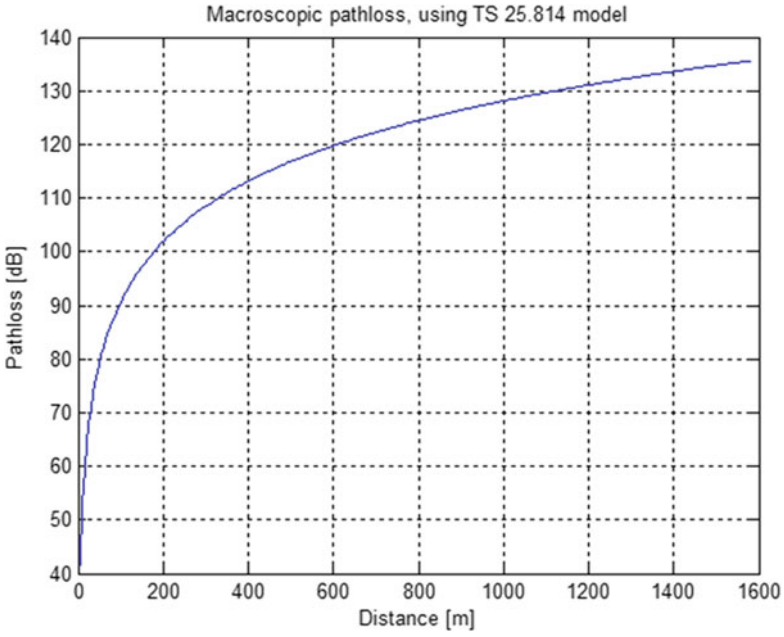

Fig. 3.2 Macroscopic path loss using TS 25.814 model

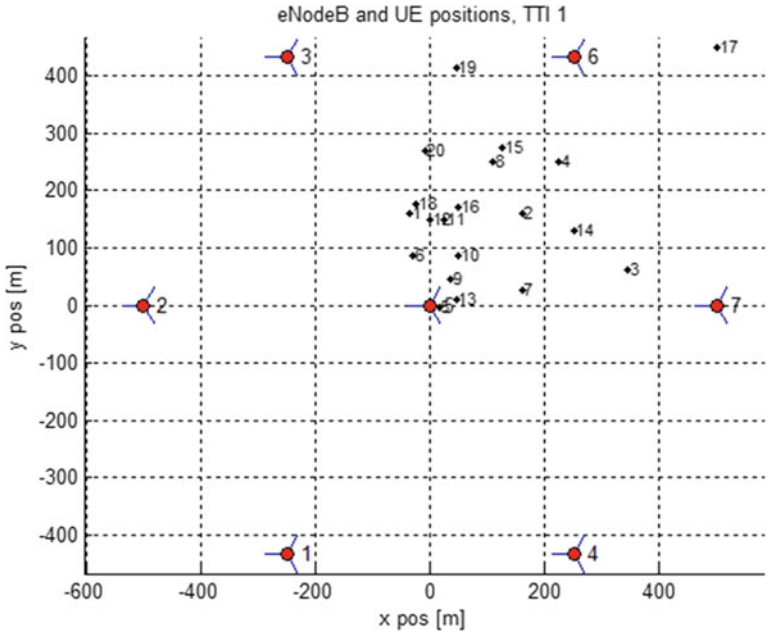

Fig. 3.3 *eNodeB* and *UE* positions at Time to Live (*TTL*) 1

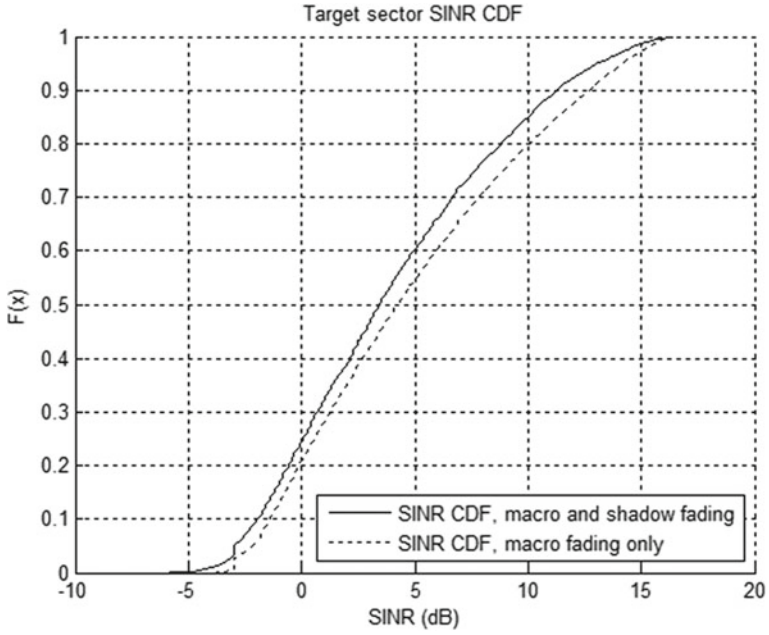

Fig. 3.4 Target sector *SINR* representation *CDF*

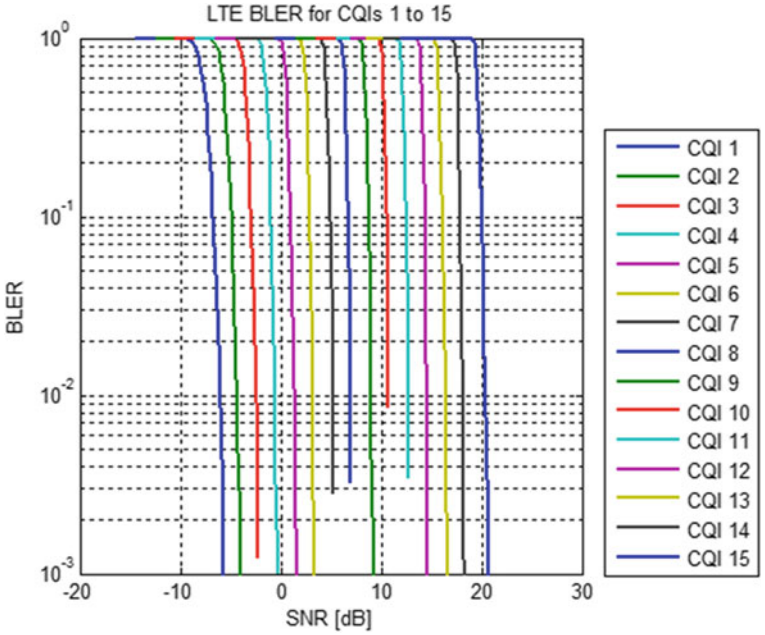

Fig. 3.5 *LTE BLER* for 15 channel quality indicators

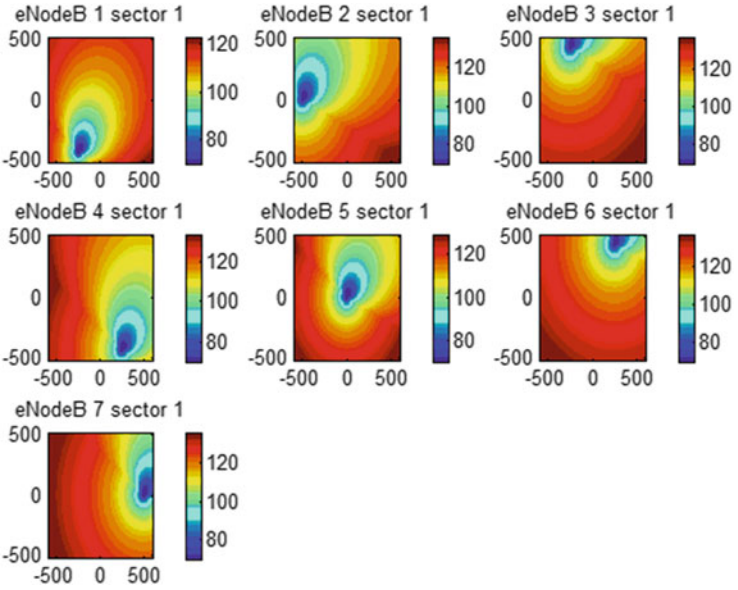

Fig. 3.6 Antenna diversity *eNodeB* sectorial *BLER* transmission contours (Sector 1)

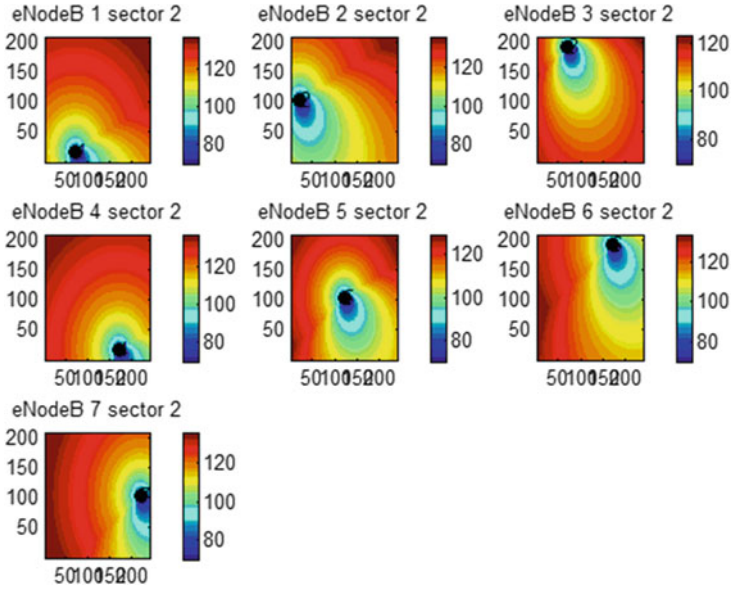

Fig. 3.7 Antenna diversity eNodeB sectorial *BLER* transmission contours (Sector 2)

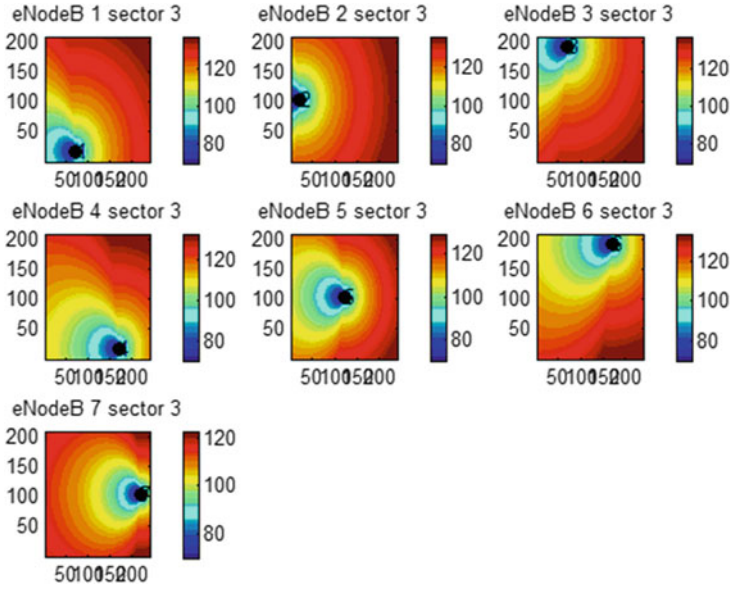

Fig. 3.8 Antenna diversity *eNodeB* sectorial *BLER* transmission contours (Sector 3)

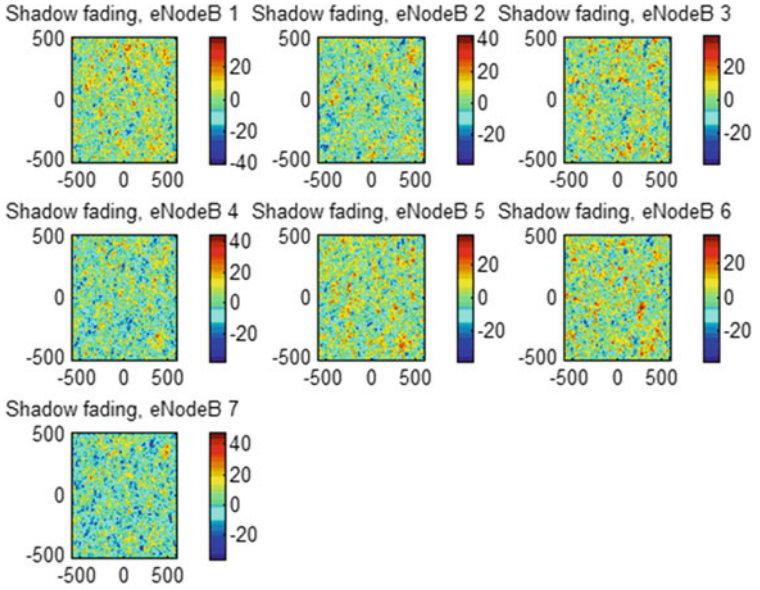

Fig. 3.9 Slow fading patterns at *eNodeB*'s in all Sectors 1–7

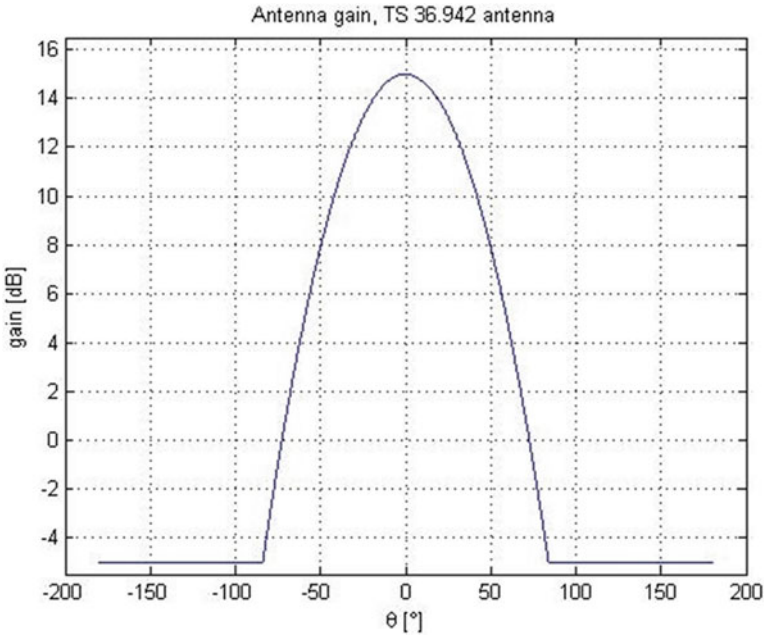

Fig. 3.10 Antenna gain TS 36.942 antenna

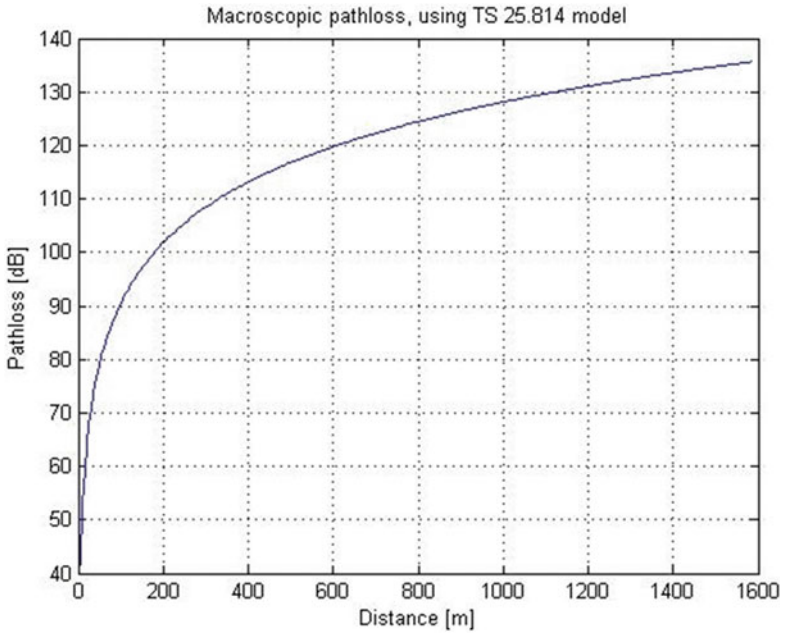

Fig. 3.11 Macroscopic path loss using TS 25.814 model which includes both *Wi-Fi* and *LTE*

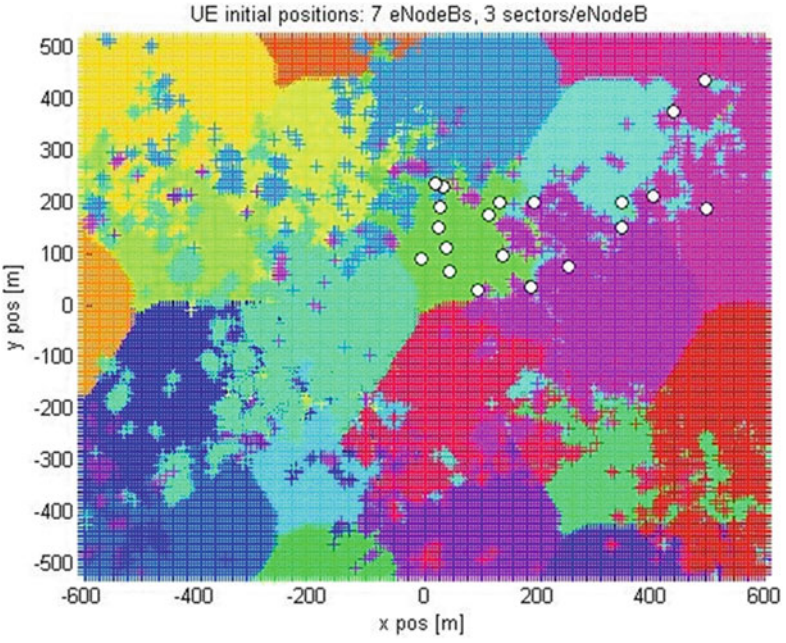

Fig. 3.12 Random *UE* generator for *LTE* and *Wi-Fi* Access Points

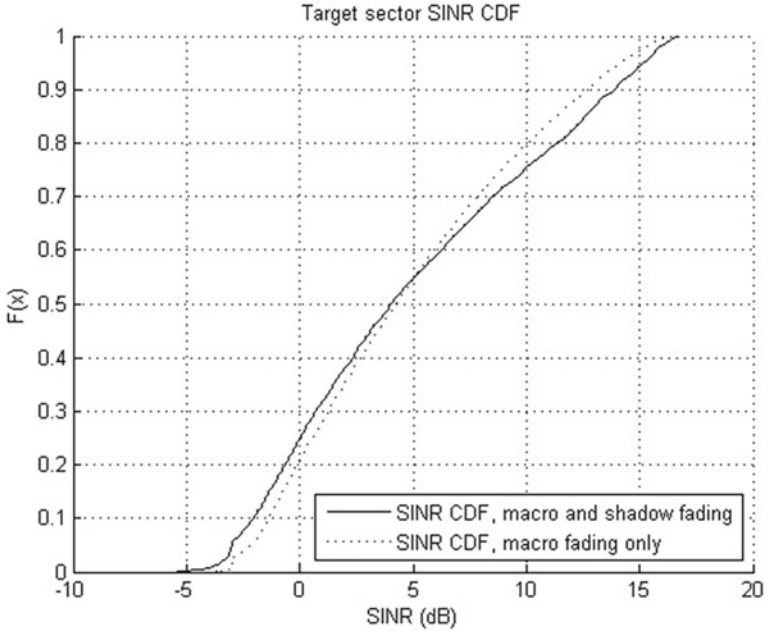

Fig. 3.13 Target sector *SINR* representations *CDF*

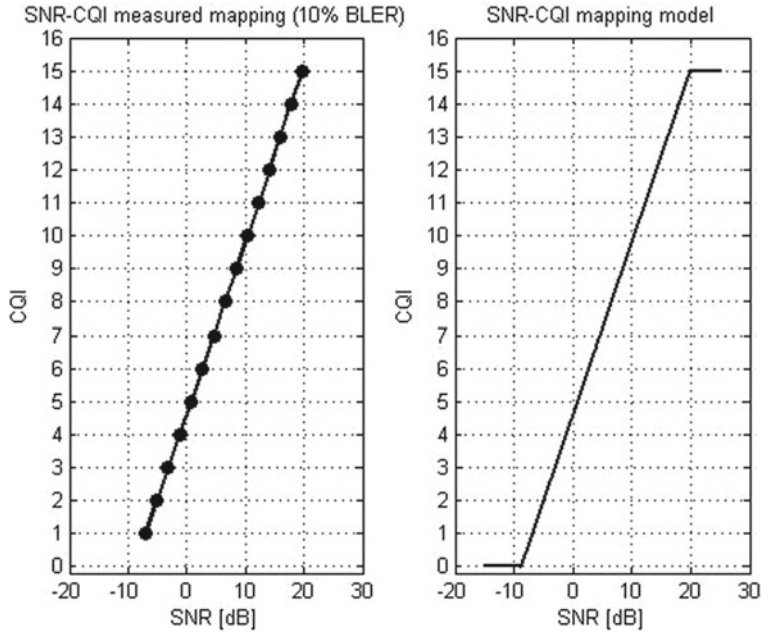

Fig. 3.14 SNR-CQI mapping for joint *Wi-Fi* and *LTE*, *GSCC* scenario

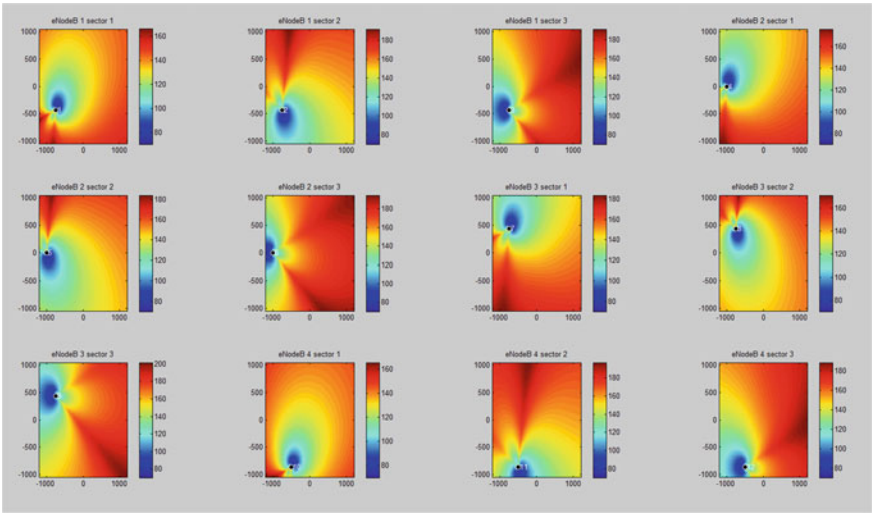

Fig. 3.15 Antenna diversity *eNodeB* and *Wi-Fi* combined sectorial *BLER* transmission contours (Access Points 1–4). We see that for the same amount of emitted energy, the area of coverage has doubled in the joint *Wi-Fi–LTE* scenario

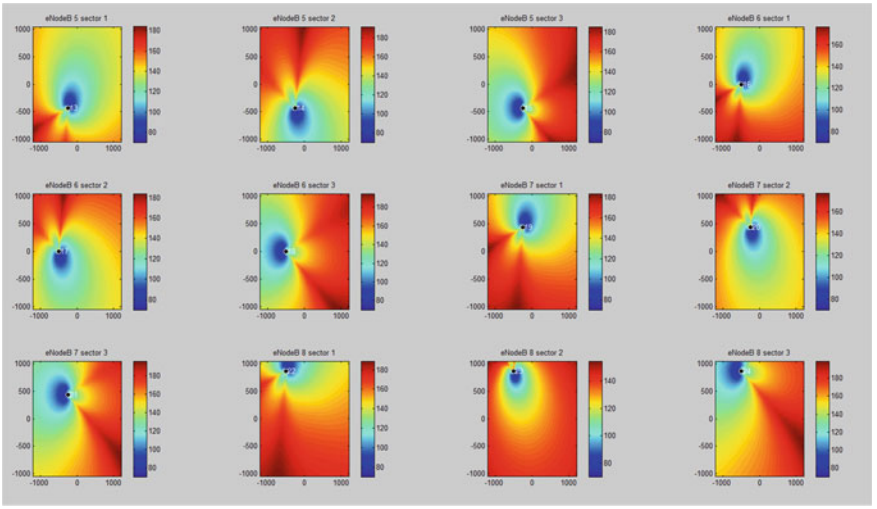

Fig. 3.16 Antenna diversity *eNodeB* and *Wi-Fi* combined sectorial *BLER* transmission contours (Access Points 5–8). We see that for the same amount of emitted energy, the area of coverage has doubled in the joint *Wi-Fi–LTE* scenario

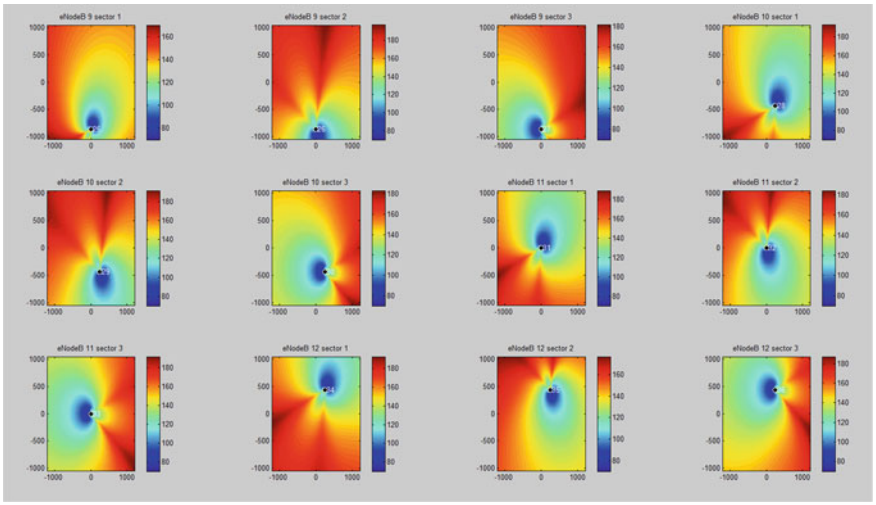

Fig. 3.17 Antenna diversity *eNodeB* and *Wi-Fi* combined sectorial *BLER* transmission contours (Access Points 9–12). We see that for the same amount of emitted energy, the area of coverage has doubled in the joint *Wi-Fi–LTE* scenario

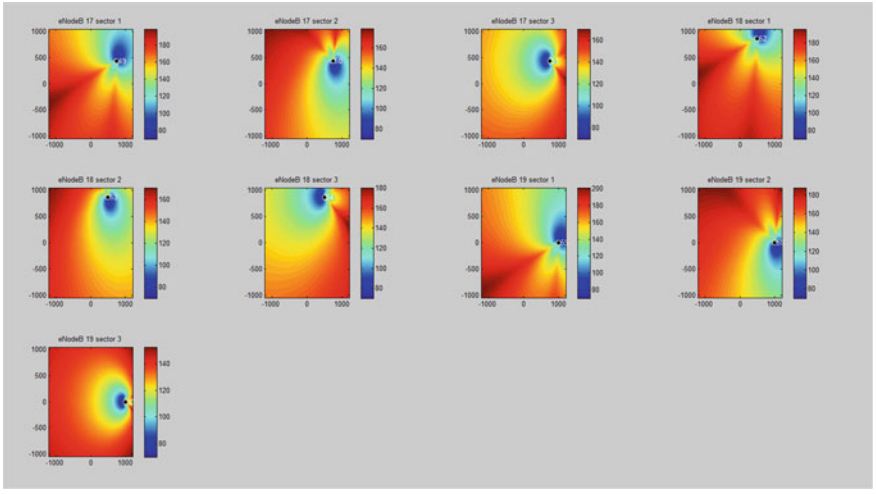

Fig. 3.18 Antenna diversity *eNodeB* and *Wi-Fi* combined sectorial *BLER* transmission contours (Access Points 17–19). We see that for the same amount of emitted energy, the area of coverage has doubled in the joint *Wi-Fi–LTE* scenario

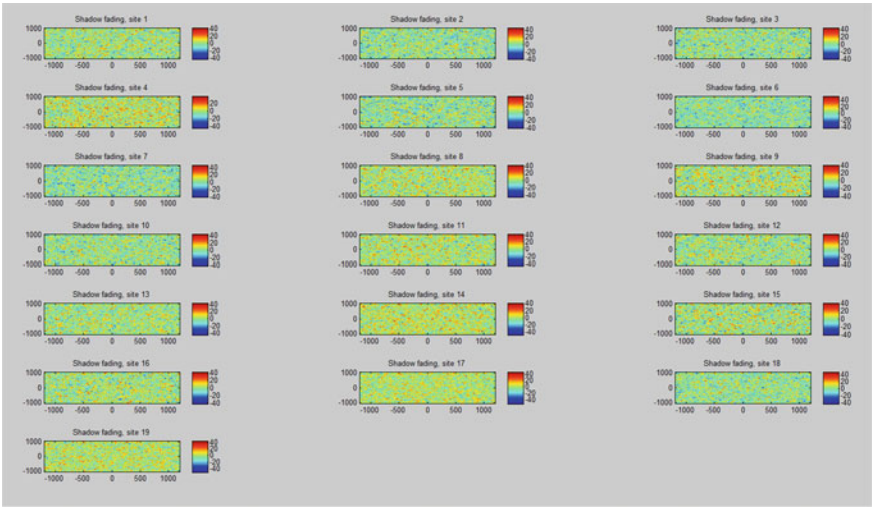

Fig. 3.19 Slow fading patterns at *eNodeB*'s and *Wi-Fi* Access Points in the joint operational scenario

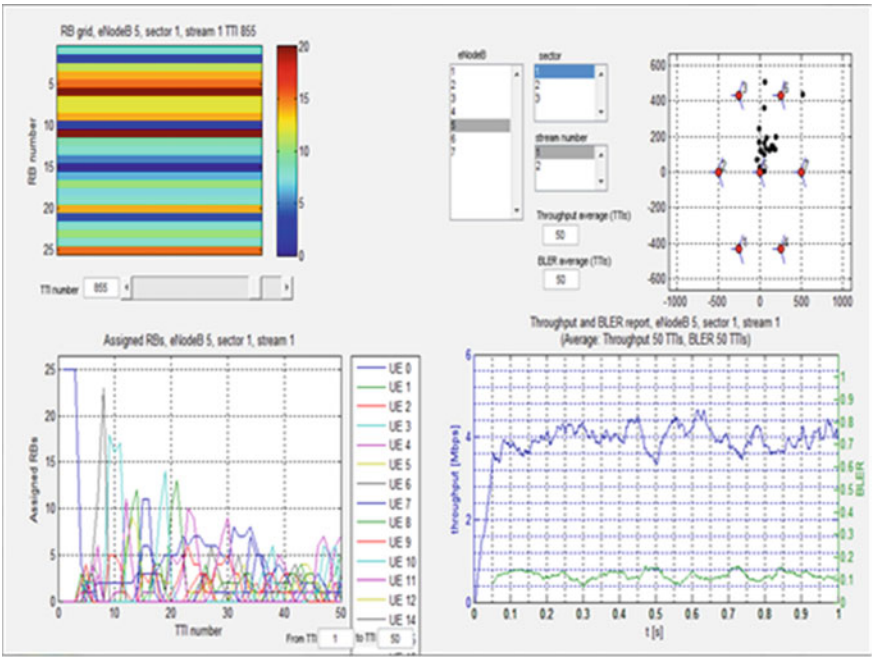

Fig. 3.20 Throughput obtained in the joint scenario is a summation of the individual throughputs of the *LTE eNodeB*'s and *Wi-Fi* Access Points

Table 3.1 Average test parameters for *LTE eNodeB* only scenario

Frequency: 2 GHz	Bandwidth: 5 MHz
*UE*s in target sector only	Simulated over 1000 TTIs
ISD (eNodeB): 500 m × 500 m	T_x Power: 43 dBm
Max *UE* per *eNodeB*: 20 × 3	UE speed: 5 km/h (pedestrian)
Path loss models simulated: Urban-macro, Urban-micro (rural not published)	

Table 3.2 Average test parameters for *LTE eNodeB* and *Wi-Fi* mixed scenario with same T_x power and doubled *ISD*

Frequency: 2 and 2.4 GHz	Bandwidth: 5 and 160 MHz
*UE*s in target sector only	Simulated over 500 TTIs
ISD (*eNodeB* and *Wi-Fi* AP's): 1000 m^2	T_x Power: 43 dBm
Max *UE* per *eNodeB*: 20 × 3	UE speed: 5 km/h (pedestrian)
Path loss models simulated: Urban-macro, Urban-micro (rural not published)	

3.3.2 Experimental Results

The proposed *GSCC* paradigm is validated from the capacity and energy consumption perspective. We consider a scenario where the prototype consists of communication between two *UE*s. The prototype setup, as shown in Figs. 3.21, 3.22 and 3.23, consists of two laptops, two servers and two smartphones acting as *UE*s. Each *UE* has access to three *CL*s-two *WLAN* (*IEEE 802.11n Wi-Fi*) and one wired (*CAT6/Fiber Optic*) *LAN*. A virtualized cloud is set up as a communication server via which both the *UE*s communicate. The *UE*s and *CL*s are positioned in space using a Poisson point distribution of intensity n, yielding the test area as a square $\{\sqrt{n} \times \sqrt{n}\}$, as shown in Fig. 3.21. The variable n is varied in the range 15–25 m in random steps. Experimentation is carried out in both indoor and outdoor environment. The throughput obtained on each *CL* is in the range of 2–4 Mbps, over the test period. Random test sets comprising of audio, video and data, of sizes ranging from 2 to 6 GB, are transmitted between the two *UE*s. The *TCP/IP* packet size is randomly varied from 10 bytes to 1500 bytes. The addressing scheme used in *IPv6* and socket virtualization is embedded in applications of the *UE*. Around 900 experiments in different environments were conducted and the average data is presented in Fig. 3.21.

In concurrence with the proposed conjecture in Sect. 3.1, the average throughput shown in Fig. 3.24, obtained over all the test runs, approaches the sum of individual throughputs of each *CL*. Due to the relative increase in throughput and simultaneous use of all the *CM*s, the active period of each *CL* decreases substantially. The major

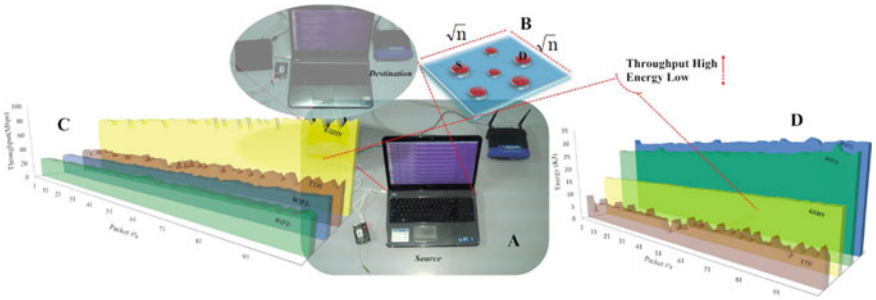

Fig. 3.21 Two *UE*s communicating with each other via multiple *CM*s is seen in A. The *CL*s comprising of two *WLAN* ports and one Ethernet wired connection at each *UE* are distributed randomly using Poisson distribution in the range of 15–250 m as shown in B. Conducting 150 runs over random sample data sets, with size ranging from 2 to 6 GB and averaging the results in C, we obtain a linear increase in throughput. The energy consumption in *GSHN* is significantly lower than other *CM*s as seen in D

Fig. 3.22 Transmitting end experimental setup of the *GSCC* paradigm

energy consumption of any wireless medium, which is during its activity period, is thus minimized in the *GSHN* architecture. The averaged energy consumption over all the test runs, as shown in Fig. 3.25, indicates that *GSCC* consumes lesser energy than *WLAN CL*s for the same data transmission, presenting a win-win situation for the *UE*s.

The task performed is to connect to the remote server and retrieve a specific file of interest. The size of the file is taken to be large (approximately 500 MB). A comparative experiment was performed to find the throughput of the system and its performance using normal *FTP* on each individual link to the server as to when the task performance of the modified *FTP* using all the links at a time. Ethernet possesses

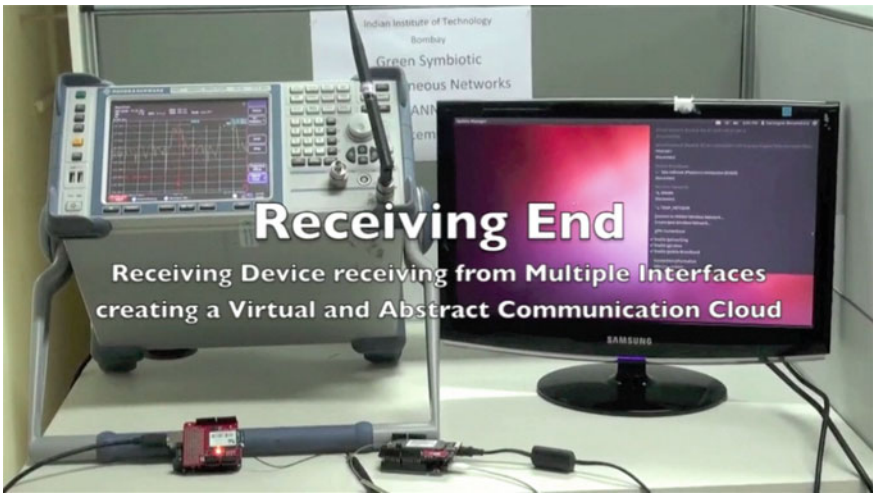

Fig. 3.23 Receiving end of the *GSCC* paradigm

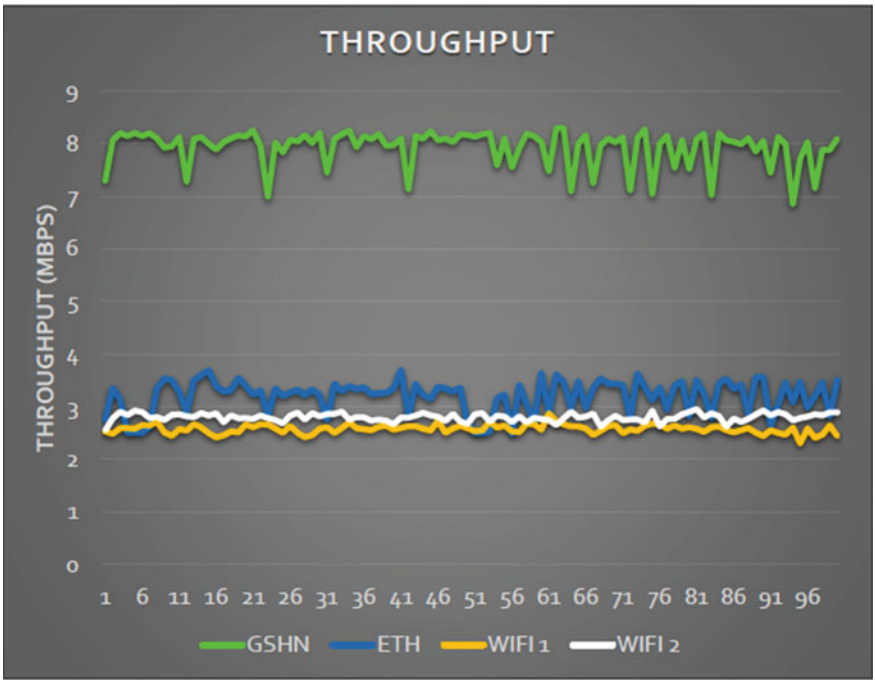

Fig. 3.24 Throughput analysis of the *GSCC* paradigm averaged over 300 trials with different topologies and environments. *GSCC* demonstratively gives the highest throughput in video streaming

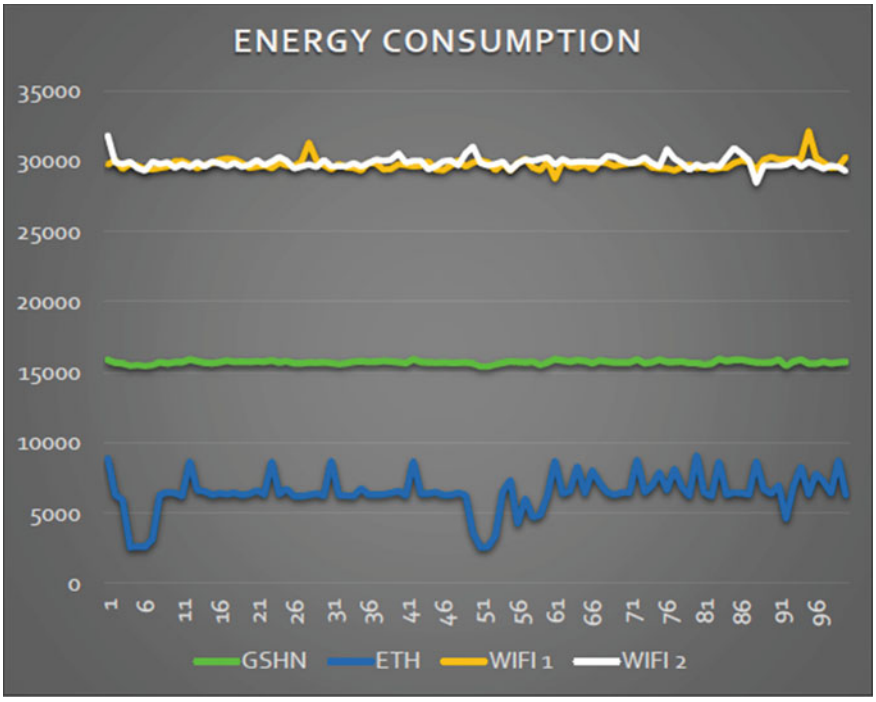

Fig. 3.25 Energy consumption analysis of the *GSCC* paradigm averaged over 300 trials. *GSCC* demonstratively gives optimum energy consumption in live streaming applications, higher than Ethernet but lower than *Wi-Fi*

the highest throughput and the least power consumption, whereas *3G* has the lowest transmission rate.

The decision function is dependent on 3 main parameters, viz. throughput, cost of the link and power consumed. In this experiment, we try to maximize throughput, neglecting the other two parameters which are the cost and the power consumption, enabling use of all the three links simultaneously to transfer the data.

The experiment is intended to show that with the improvement in throughput, there will be a minimal increase in energy usage in this scenario. We have captured the parameters of the 500 MB, 1 GB, 2 GB file transfers through different links which are as below. The graphs in Figs. 3.26, 3.27, 3.28, 3.29, 3.30 and 3.31 are plotted for the below mentioned parameters during the transfer of the file.

- Average throughput versus time
- Power consumption versus time

This power consumed is calculated depending on the maximum transmission power utilized by the individual links. The outputs are tabulated in and quantified in Table 3.3.

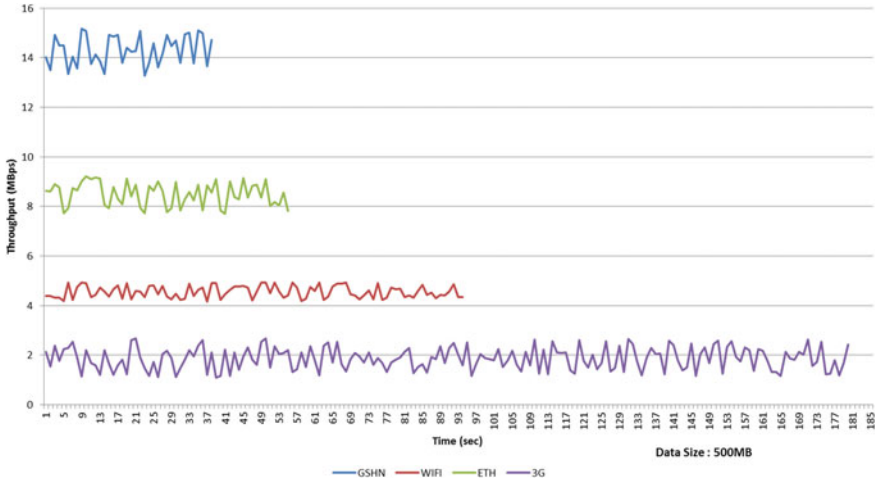

Fig. 3.26 Throughput analysis of the *GSCC* paradigm for small-sized data (500 MB) averaged over 300 trials. *GSCC* demonstratively gives optimum energy consumption in downloading and browsing applications

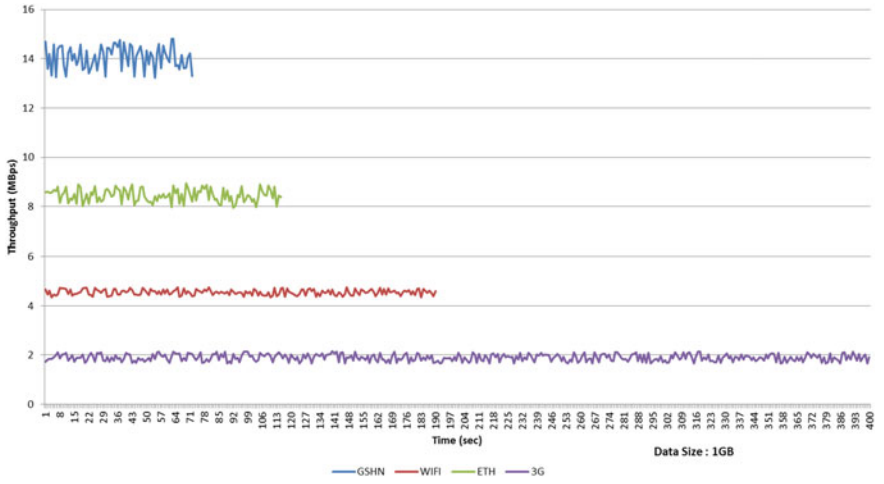

Fig. 3.27 Throughput analysis of multiple networks for medium-sized data (1 GB) averaged over 300 trials. *GSCC* demonstratively gives optimum energy consumption in downloading and browsing applications

From the results obtained, we can see that there has been an improvement in the overall throughput of the system. We have also seen an overall increase in the power consumption in the proposed protocol as shown in Fig. 3.29. We can conclude that the power consumed has increased but only for a limited duration. This power consumption takes place during the transfer of data, so it is equivalent to the power

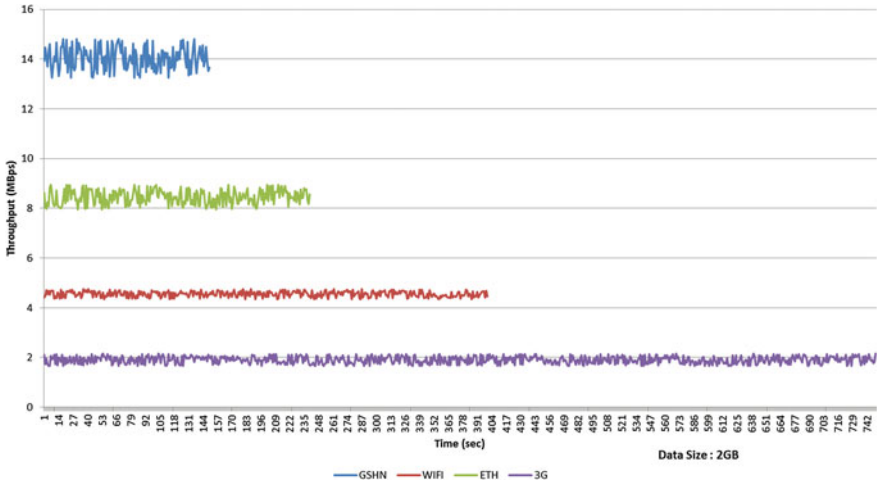

Fig. 3.28 Throughput analysis of multiple networks in big-sized data (2 GB) averaged over 300 trials. *GSCC* demonstratively gives optimum energy consumption in downloading and browsing applications

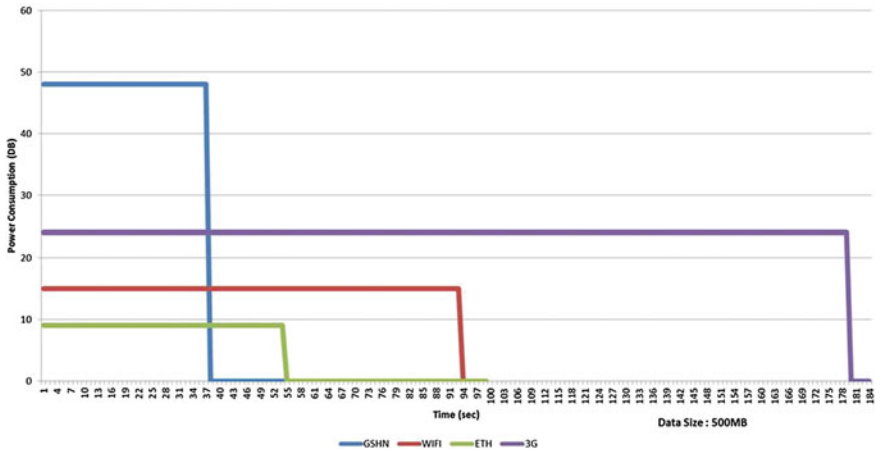

Fig. 3.29 Energy consumption analysis of the *GSCC* papradigm averaged over 900 trials of big, medium and small data. *GSCC* demonstratively gives optimum energy consumption in downloading and browsing applications

consumed by only one communication link. This overall increase in power is compensated by the reduction in time required for overall transmission, implying that the energy required for the overall task is constant. This constant energy output has been augmented by the increased throughput.

We further experiment with the implementation of sending high definition (*HD*) voice data using both circuit switching and packet switching simultaneously. As we

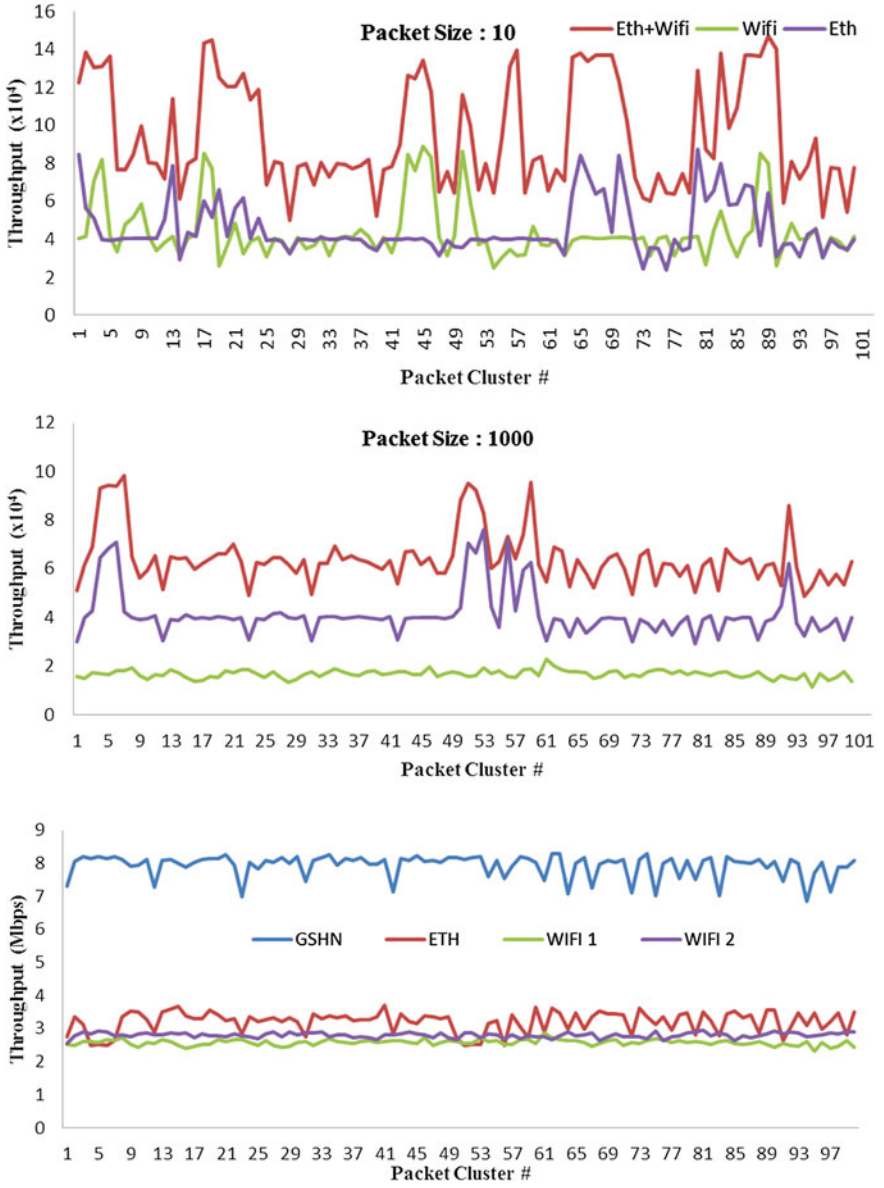

Fig. 3.30 The average throughput shown over 150 test runs, wherein A has a smaller packet size of 10 MB and B has a higher packet size of 1000 MB. The increase in throughput is consistent proportionately with the increase in packet size. Overall throughput of *GSCC* as shown in C shows a linear summation of throughputs as compared to the original mediums

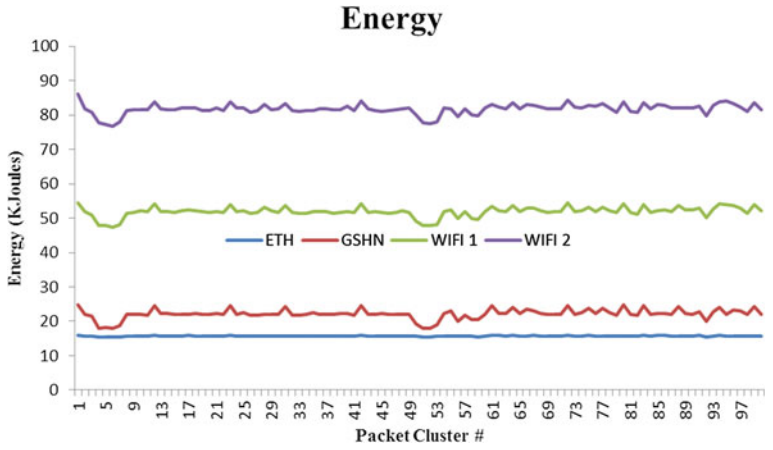

Fig. 3.31 The average power consumption showed over 150 test runs, for transmitting per packet. We see that for overall transmission, the power consumed by *GSHN* is slightly higher than the wired Ethernet line, but significantly lower than its wireless counterparts

Table 3.3 Experimental data comparison of different standalones with GSCC paradigm

Experimental results	Ethernet	*Wi-Fi*	*3G*	GSCC
Maximum throughput (Mbps)	9.2	4.6	2.4	15
Average throughput (Mbps)	8.5	4.25	2.1	14.1
Power for transmission (dBm (mW))	9 (8.8)	25 (292)	24 (251)	15 (32)
Time taken (s)	57	96	181	39

know we cant increase the data speed on a voice call using circuit switching, we use packet switching running parallel to the established call and transmit the voice data through it. We create a scenario where we transfer voice data using both packet switching (e.g. Skype call) and circuit switching (cellular call) and transmit voice data using both the links where on the other side we will combine both the receptions and make it appear as *HD* version. We are sending higher bits of voice data from circuit switching in order to assemble and perform normal operation in case of errors in packet switching.

The experimental setup is designed to store voice data in buffer which is sampled at a higher sampling frequency. We split the data into two parts of which one part will be sent through circuit switching and the other half is sent through packet. Packet switching is by using a Skype call and circuit switching is by a normal cellular call. The socket programming virtualizing the communication ports for establishing the *GSCC* paradigm is available at SPANN Lab (https://www.ee.iitb.ac.in/~spann/projects.html). The code is divided into two parts: one for cognitive splitting and transmitting, and the other for the reassembling at the receiving end.

We were able to establish transmission of data using both circuit switching and packet switching. Generally, we transfer 8 kHz sampled data using circuit switching. By implementing this technique, we can establish 16 kHz sampling or more, without affecting the circuit switching data link with the data rate of circuit switching remaining unaltered. So, with this implementation, we can establish a *HD* link on voice data, which will increase clarity, and a reliable network as it is using all the available communication mediums, an archetype in published literature.

A basic advantage offered by the *GSCC* paradigm is the unified usage of all available *CM*'s. To this effect, the next stage development calls for a unified controller, which enables effective implementation of the proposed paradigm. In the next chapter we introduce the design and implementation of a cognitive software-defined Universal Communication Modem (*UCM*), which effectively handles multiple *CM*'s in one device.

References

1. Cheng Wang, Xiang-Yang Li, ChangJun Jiang, Shaojie Tang, and Yunhao Liu. Multicast throughput for hybrid wireless networks under gaussian channel model. IEEE Transactions on Mobile Computing, 10(6):13, 2011.
2. Cheng Wang, Xiang-Yang Yang Li, Chang-Jun J. Jiang, Shao-Jie Tang, Yunhao Liu, and Jizhong Zhao. Scaling laws on multicast capacity of large scale wireless networks. In *INFOCOM 2009, IEEE*, page 8, 2009.

Chapter 4
GSCC Universal Modem: Unifying Communications

Abstract Unification of varied signal transmission through a cloud structure of communications is one of the major postulates of the *GSCC* paradigm. Unification yields new possibilities in architecture evolution wherein a single access point can manage communication demands of multiple standards. In this chapter, the authors propose a novel access point (modem design), which allows for one modem to control all types of Communication Mediums (LTE, *3G*, *Wi-Fi*, etc.) effectively delineating a universal communication modem (*UCM*). Three innovative modules are proposed, which when amalgamated together, result in *UCM*. Firstly, when different types of signals are sent and received on different mediums, the receiver needs intelligent channel estimation for coupling and decoupling these signals. An intelligent blind channel estimation scheme is proposed for achieving this. Secondly, the received signals of different or the same mediums need to be classified and routed to the appropriate devices. To accomplish, this we propose a novel Optimized Coulomb Energy Neural Network (OCENN) for classification of incoming signals. Finally, an innovative Internet Protocol (IP) based addressing schematic is proposed, which allows bidirectional communications.

Unification of varied signal transmission through a cloud structure of communications is the paramount advantage offered by the *GSCC* paradigm. This yields new possibilities in architecture evolution wherein a single access point can manage transmissions of multiple standards. In this chapter, we propose a novel access point (modem design), which allows for one modem to control all types of Communication Mediums (*3G*, *Wi-Fi*, etc.) effectively delineating a universal communication modem (*UCM*). 3 innovative modules are proposed, which when amalgamated together, result in *UCM*. Firstly, when different types of signals are sent and received on different mediums, the receiver needs intelligent channel estimation for coupling and decoupling these signals. An intelligent blind channel estimation scheme is proposed for achieving this. Secondly, the received signals of different or the same mediums need to be classified and routed to the appropriate devices. To accomplish, this we propose a novel Optimized Coulomb Energy Neural Network (OCENN) for classification of incoming signals. Finally, an innovative Internet Protocol (IP) addressing schematic is proposed, which allows bidirectional communications. Experimental

H.D. Mustafa et al., *Green Symbiotic Cloud Communications*,
DOI 10.1007/978-981-10-3512-8_4

results over a real-time *GSCC* paradigm implemented over low voltage power line networks shows that *UCM* achieves a throughput of 75, 63 and 56 mbps at a SNR of 30 dB for the data, audio and video transmissions respectively. Comparative analysis of UCM with existing state-of-the-art OFDM and FMT modems is presented, which shows *UCM* as a viable prospective option.

In this chapter, we propose 3 modules, namely channel estimation, signal classification and bidirectional communication, which expansively combine to form *UCM*. The proposed modules are standalone and hence can be applied to all the existing communication systems. We however select the Powerline Communications (*PLC*) [1, 2] system not only because it is the most densely spread but also because the primary purpose of power distribution cannot be performed by other existing systems. *PLC* systems allow the possibilities of utilizing the existing most pervasive and dense infrastructure and also amalgamates the entire communication archetype into a singular paradigm, which is in alignment with the design postulates promoted by the *GSCC* architecture [3–5]. An essential component of any communication architecture is the Access Point, and *PLC*'s are no exception. It characterizes an integral element in determination of costs, ease of use and practicability of the architecture. Proposed *PLC* modems in literature are very application- and device-specific, which prevents unification of this revolutionary system [2, 4, 6, 7]. In [1, 7–10] power line modem structures were proposed, which work at the device level. Furthermore, a true universal modem in tune with the *GSCC* architecture allows portability of operations in conjunction with the existing communication architecture. We realize these requirements by proposing a Universal Communications Modem (*UCM*), addressing the need for a comprehensive architecture over the power lines architecture.

4.1 Architecture of *UCM*

The block diagram of UCM is shown in Fig. 4.1. To classify the modem as universal, it should be able to effectively handle communication between multiple devices

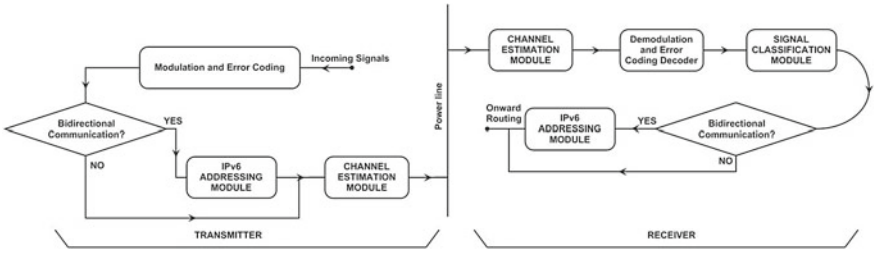

Fig. 4.1 Process flow/block diagram of the Universal Communication Modem (*UCM*) designed on the postulates of the *GSCC* architecture

and users connected to it [3, 7]. We identify the following issues which restrict the universal compatibility of the modems:

(a) A universal modem should be able to receive varied types of signals in real time simultaneously.
(b) The received signals need to be routed to the specific device that it associates with.
(c) The devices connected to a universal modem will have bidirectional communications and unique identification requirements.

These identifications are resolved in the subsequent subsections with the design of *UCM*.

4.1.1 Transmitter Architecture

On the T_x end of *UCM*, incoming signals from the assortment of devices are forward error-coded followed by modulation. The error-coding process entails scrambling, encoding, puncturing and interleaving [3, 5, 7]. The modulation scheme involves mapping, transform, attachment of preamble and prefixes, and RC shaping [6]. If only one type of device was connected to this modem, the above two processes would suffice. However, in *UCM*, different types of devices will be connected and some of them will even require bidirectional communication. To handle this situation, we propose an innovative *IP* addressing module which assigns temporary dynamic internet protocol address in line with the *GSCC* architecture, for allowing two way communications between such devices. The signals are then inductively coupled to the power line after performing intelligent channel estimation.

4.1.2 Receiver Architecture

In this section we describe the signal flow through the R_x architecture. This involves channel estimation, demodulation and error code decoding, signal classification and bidirectional communication module. The receiver measures the average power level on the power line in real time and compares it to a predetermined initial gain during the idle state. As identified earlier, UCM receives various signals where every archetype signal has its specified energy threshold. To overcome this limitation an intelligent blind channel estimation module is proposed. The signal is further demodulated and error decoded through an inverse procedure to the one in the transmitter [5, 6, 11, 12]. *UCM* furthermore needs to confirm the identity of the received signal and for this, we propose a novel software-based signal classification module [13, 14]. Upon identification of the signal, we further check if the signal pertains to the device requiring bidirectional communication, in which case it is passed to the virtualized *IP* addressing module. The signal is then routed to the connected device.

The introduction of the above three modules, achieves our rationale of embedding universal connectivity and functionality in the modem, delineating a comprehensive *UCM* in its true sense.

4.2 Channel Estimation of UCM

The channel estimation module forms an essential component of both the transmitter and the receiver circuit of *UCM*. For the transmission module, we have knowledge of the various parameters of the signal; but at the receiving end, we have no such knowledge. Further, the coupling circuit is also unaware of what type of signal it has to extract from the power line when multiple signals are being transmitted over the same medium. This demands for a blind and intelligent detection of the channel conditions to infer the type of signal received. An efficient way of channel estimation is through predicting the energy contained in the transmitted signal over the power line [5, 12]. Frequency-based classification will not work in the unified *GSCC* paradigm because multiple communication links (e.g. multiple *Wi-Fi* signals of the same frequency band) might be present at the same time. We propose an intelligent coupling energy mixture model for channel estimation in *UCM*. When we receive the signal $x_i \in \mathcal{X} = \{x_1, x_2, \ldots, x_N\}$, we assign the individual energy components of the signal at various instances of time $t_i \in \mathcal{T} = \{t_1, t_2, \ldots, t_N\}$ to G neurons $\mathcal{R} = \{r_1, r_2, \ldots, r_G\}$. The network is assigned a coupling function h_{kl} which defines the potency of orthogonal interaction between two neurons r_k and r_l for $k, l \in \{1, 2, \ldots, G\}$. During the training session, each neuron r_k is associated with a reference model θ_k that represents probability of that particular kind of signal in the data space. The learning procedure is devised with an intention of having maximum correlations between the neurons and its neighborhood for a given input signal, i.e. identifying and grouping the energy components contained in a particular signal. We further define the local attraction energy between r_k and its neighbors as 4.1 [10, 12]

$$E_{(x_i|k)} = \sum_{i=1}^{G} h_{kl} r_k(x_i; \theta_k) r_l(x_i; \theta_l) = r_k(x_i; \theta_k) \sum_{i=1}^{G} h_{kl} r_l(x_i; \theta_l) \tag{4.1}$$

where $r_k(x_i; \theta_k)$ denotes the response of neuron r_k to x_i, which can be modeled by an isotropic Gaussian density in analogy to the noise contained over power lines. The energy of the entire signal over the network for x_i is defined as 4.2

$$E_{x_i} = \sum_{k=1}^{G} E_{x_i|k} \tag{4.2}$$

and the energy function [12] to be maximized is 4.3

$$C = \sum_{i=1}^{N} \log E_{x_i} \tag{4.3}$$

We make an important observation here that the term $\sum_{i=1}^{G} h_{kl} r_l(x_i; \theta_l)$ can be considered as the energy response of r_k when comparison between the energy potency of r_k and r_l is performed. Further, the neuron response component $r_l(x_i; \theta_l)$ of the energy response can be modeled as a multivariate Gaussian distribution as shown in 4.4

$$r_l(x_i; \theta_l) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_l|^{\frac{1}{2}}} e^{-\frac{1}{2}(x_i - \mu_l)^T \Sigma_l^{-1} (x_i - \mu_l)} \tag{4.4}$$

for $l = 1, 2, \ldots, G$. The energy response of r_k can thus be written as shown in 4.5

$$\prod_{l \neq k} r_l(x_i; \theta_l)^{h_{k,l}} \tag{4.5}$$

Thus, for a given x_i, we can define the local energy response between r_k and its neighbors in the same class as 4.6

$$p_s(x_i|k; \Theta, h) = r_k(x_i; \theta_k)^{h_{kk}} \prod_{l \neq k} r_l(x_i; \theta_l)^{h_{k,l}} = \prod_{l=1}^{G} r_l(x_i; \theta_l)^{h_{k,l}} = e^{\sum_{l=1}^{G} h_{kl} \log r_l(x_i; \theta_l)} \tag{4.6}$$

where Θ is the set of reference models θ_k and denotes the cumulative coupling function. We can thus summarize the global energy response over the entire network as 4.7

$$p_s(x_i; \Theta, h) = \sum_{k=1}^{G} w_s(k) p_s(x_i|k; \Theta, h) \tag{4.7}$$

where $w_s(k)$, the learning weights of the network for $(k = 1, 2, \ldots, G)$ are fixed at $1/G$. When the local energy response $p_s(x_i k; \Theta, h)$ is maximized for each neuron r_k, the spatial ordering between neuron r_k and its class for a given signal sample x_i is learned. Owing to this fact, we use equal weights $w_s(k) = 1/G$ for the global energy response $p_s(x_i; \Theta, h)$ in order to account for the spatial structure learned by the local class neurons. This type of channel estimation also avoids the learning domination by a few high energy information signals giving *UCM* a wider operating range. A diagrammatic representation of the proposed model is shown in Fig. 4.2. Having established an intelligent mechanism of extracting various types of signals sent over the power lines, which can be easily extended to any communication medium (wireless, fiber optics, wired, etc.), we now address the issue of classifying and routing them onwards to the multiple devices connected to UCM.

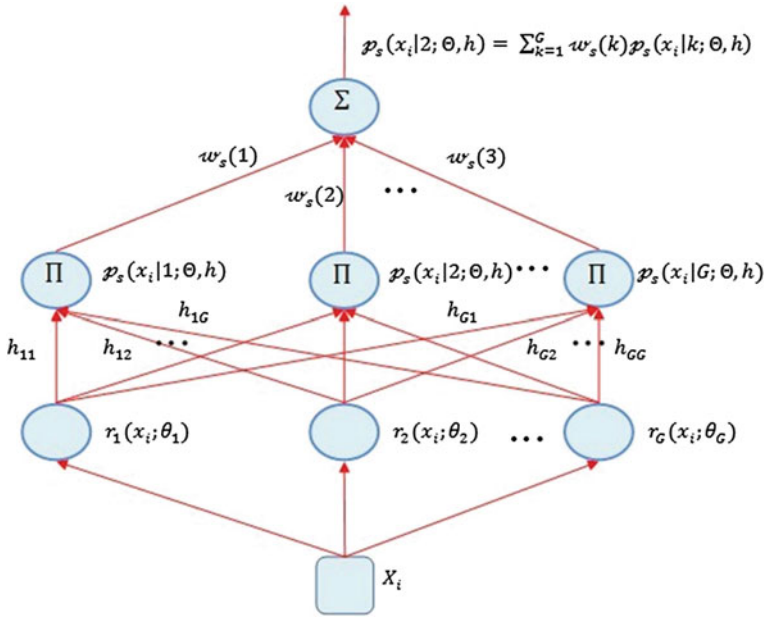

Fig. 4.2 Blind Kohonen Neural Network channel estimation module for energy-based classification of multiple signals received simultaneously at the *UCM*

4.3 Signal Processing and Classification

A significant characteristic advantage of *PLC* or any other communication medium, which we aim to exploit from the perspective of unification, is its ability to allow transmission of all types of information signals. This infers that *UCM* should have the ability to route its received signals correctly, efficiently and economically to the multiple devices connected to it. In the existing scenario, a central modem receives the signals and sends them to all the connected devices. At the individual devices, another hardware-based filter or modem is connected which filters the associated signal for that device. However, this hardware-based approach significantly increases the cost and makes *PLC* practically unfeasible [1, 9, 11]. We divert ourselves from the hardware-based approach and develop a novel software-based approach termed as Optimized Coulomb Energy Neural Network (*OCENN*) for classification of signals in *UCM*.

A basic advantage of Coulomb Energy Networks is the presence of repulsive-attractive Coulomb potential between the target classes [13–15]. We take advantage of this property in our formulation, where direct attention is paid to the internal representations. This is because we are not only concerned with the discovery of a mapping that fulfils local constraints (fitting the samples) but we also target aggregate properties of the mapping as a whole. Another distinct difference in our *OCENN* is

that our approach considers pair-wise interactions between vectors where neither vector in the pair is designated as a target. This nature of the network, not only reduces the dimensionality of the processed data, but also guarantees convergence of the incoming signals, which is important from the *UCM* perspective that it does not get jammed. By guaranteed convergence we imply that an incoming signal will be classified, regardless of the decision being right or wrong, thus preventing jamming at the modem.

In the proposed *OCENN*, each input signal $\mathcal{X}$ carries with respect to each pair of target classes, two charges of attraction, denoted by Q_i and Q_j respectively. The Coulomb energy [13–15] ψ is calculated as 4.8

$$\psi = \frac{1}{2L} \sum_{i=1}^{M} \sum_{j=1}^{M} Q_i Q_j |x_i - x_j|^{-L} \tag{4.8}$$

and the corresponding logistics function L is defined as 4.9

$$L_n \left(\sum_{m=1}^{K} w_{nm} f_{mi} \right) = \left(1 + e^{-\frac{1}{kT}} \left(\sum_{m=1}^{K} w_{nm} f_{mi} + \Theta \right) \right)^{-1} \tag{4.9}$$

where L_n denotes the activity of the nth neuron, w_{nm} is the matrix of weights, f_{mi} is the afferent activity in the space $\mathbb{R}^K$. Our target is to minimize the Coulomb energy ψ by computing the gradient of a vector defined by G_{mn} (mth component of vector G_n) with respect to the weights w_{mn}. The vector G can be physically interpreted as the potential energy of the network or the network output. We assign the set of vectors G_{mn} in the population as 4.10

$$G = \sum_{mn} \sum_{i} \sum_{i+1}^{j} \frac{h_{mn}}{|x_i - x_j|^2} \tag{4.10}$$

where

$$h_{mn} = \begin{cases} 1 & \text{for Class of } x_i \neq \text{Class of } x_j \\ -1 & \text{for Class of } x_i = \text{Class of } x_j \end{cases}$$

We now need to devise a procedure for the computation of the weights from the input to the hidden layers and from the hidden layers to the outputs. We adapt the gradient of the G vector to calculate the change in weights as $\Delta w = -\alpha \nabla_w G$ which further from 4.10 yields

$$\nabla_w G = \frac{\partial G}{\partial w} = \sum_{mn} h_{mn} \left(\frac{\partial \left(\sum_i \sum_{i+1}^{j} \left(\frac{1}{|x_i - x_j|^2} \right) \right)}{\partial w} \right) \tag{4.11}$$

We further make an important observation which infers (4.12)

$$\frac{\partial\left(\frac{1}{|x_i-x_j|^2}\right)}{\partial w}=-\frac{\left(\frac{\partial\left(|x_i-x_j|^2\right)}{\partial w}\right)}{|x_i-x_j|^4}=-2\frac{\left(\frac{\partial\left(|x_i-x_j|\right)}{\partial w}\right)}{|x_i-x_j|^3} \tag{4.12}$$

We also take $|x_i-x_j|\equiv\sqrt{\sum_r\left(x_{ir}-x_{jr}\right)^2}$, where x_{ir} is the output that the rth neuron produces when the ith input vector is supplied. Substituting this in (4.12), we have (4.13) and (4.14)

$$\frac{\partial|x_i-x_j|}{\partial w}=\frac{\partial\sqrt{\sum_r\left(x_{ir}-x_{jr}\right)^2}}{\partial w}=\frac{\frac{1}{2}\left(\frac{\partial\sum_r\left(x_{ir}-x_{jr}\right)^2}{\partial w}\right)}{\sqrt{\sum_r\left(x_{ir}-x_{jr}\right)^2}} \tag{4.13}$$

$$\frac{\partial\sum_r\left(x_{ir}-x_{jr}\right)^2}{\partial w}=2\sum_r\left(x_{ir}-x_{jr}\right)\frac{\partial\sum_r\left(x_{ir}-x_{jr}\right)}{\partial w} \tag{4.14}$$

Combining (4.11), (4.12), (4.13) and (4.14) gives (4.15)

$$\frac{\partial G}{\partial w}=-2\sum_{mn}\left(h_{mn}\sum_i\sum_{i+1}^{j}\left(\sum_r\frac{\left(\left(x_{ir}-x_{jr}\right)\left(\frac{\partial\left(x_{ir}-x_{jr}\right)}{\partial w}\right)\right)}{|x_i-x_j|^3\left(\sqrt{\sum_r\left(x_{ir}-x_{jr}\right)^2}\right)}\right)\right) \tag{4.15}$$

If we take the partial update from the intermediate to the output neuron $\frac{\partial x_{ir}}{\partial w_{kr}}=x_{ir}\left(1-x_{ir}\right)z_{ik}$, where z_{ik} is the output of the kth intermediate neuron when the ith vector is the input, we can obtain the weight updating rule from intermediate to output layers as (4.16)

$$\frac{\partial G}{w_{kr}}=-\frac{2\sum_{mn}h_{mn}\sum_i\sum_{i+1}^{j}\left(\sum_r\left(x_{ir}-x_{jr}\right)\left(x_{ir}\left(1-x_{ir}\right)z_{ik}-\left(x_{jr}\left(1-x_{jr}\right)z_{jk}\right)\right)\right)}{\sum_i\sum_{i+1}^{j}|x_i-x_j|^3\left(\sqrt{\left(x_{ir}-x_{jr}\right)^2}\right)} \tag{4.16}$$

Similarly the weight updates of the input to the intermediate neuron can be classified as (4.17)

$$\frac{\partial G}{w_{mk}}=-2\sum_{mn}h_{mn}\sum_i\sum_{i+1}^{j}\frac{\sum_r\left(x_{ir}-x_{jr}\right)\left(x_{ir}\left(1-x_{ir}\right)z_{ik}\left(1-z_{ik}\right)\right)w_{kr}x_{im}-\left(x_{jr}\left(1-x_{jr}\right)z_{jk}\left(1-z_{jk}\right)\right)w_{kr}x_{jm}}{|x_i-x_j|^3\sqrt{\sum_r\left(x_{ir}-x_{jr}\right)^2}} \tag{4.17}$$

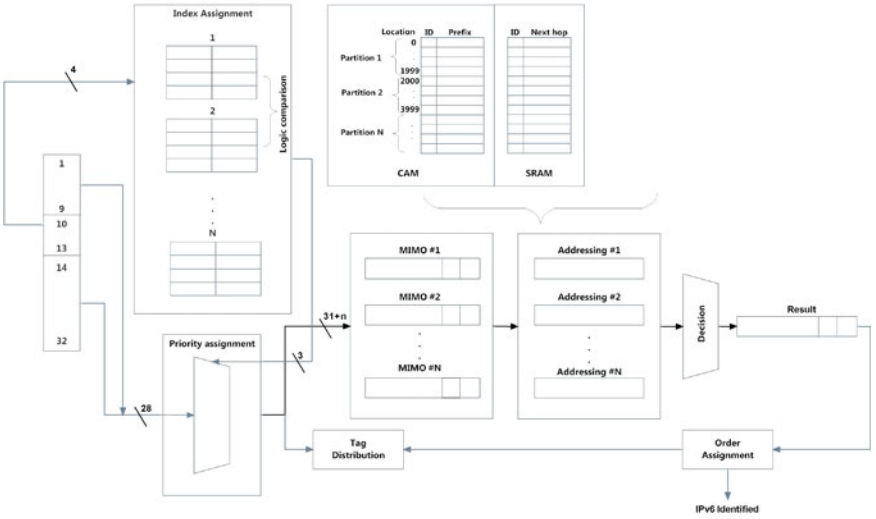

Fig. 4.3 Duplex communication is facilitated in UCM with the help of a virtualized IP scheme

The *OCENN* is trained using the algorithm shown in Fig. 4.3a. During the training, the algorithm selectively stores a set of prototypical inputs called prototypes (training patterns) that are obtained from the training data. It also assigns to the field of influence (hypersphere) of each prototype, a radius called lambda (λ) or threshold distance. The hypersphere maximizes its radius but in such a way that patterns from two different classes do not overlap. This is done to reduce the overlap region by enforcing the boundary condition r_x. For each prototype, a specific class is assigned. The network modifies its weights while training as mentioned in the previous sections. Cross validation is used to avoid overtraining of the network.

Once the network is trained, it classifies the new incoming data or pattern based on the prototypes. When a new pattern arrives, the algorithm finds all hyperspheres in which this new pattern falls and classifies it according to what it has learnt from the training examples.

4.4 Duplex Communication Module

The judiciously designed *OCENN* helps in efficient and economical classification of signals related to different devices connected to *UCM*. However, *OCENN* will not be able to handle cases where multiple devices of the same type are connected. Let us illustrate this with an example. Consider one such *UCM* connecting devices in a house to the power line. There may be multiple computers/telephones in the house that will be connected to this modem. In the event that two or more similar devices want to communicate through the power line, the modem is likely to receive

Algorithm 3: *OCENN* training

begin initialize $j \leftarrow 0$, $n \leftarrow$ *num patterns*, $\epsilon \leftarrow$ *small param*, $\lambda_m \leftarrow$ *max radius*
  repeat
    $j \leftarrow j + 1$;
    $w_{ij} \leftarrow x_i$ (train weight);
    $\hat{x} \leftarrow \arg\min_x \omega_i D(x, x') - r_x$ (find nearest hypersphere's centre not in ω_i);
    $\lambda_j \leftarrow \min [D(\hat{x}, x') - r_j, \lambda_m]$ (set radius);
    if $x \in \omega_k$ **then**
      $\alpha_{jk} \leftarrow 1$;
    end
  until $j = n$;
end

Algorithm 4: *OCENN* testing

begin initialize $j \leftarrow 0$, $k \leftarrow 0$, $\mathbf{x} \leftarrow$ test pattern
  repeat
    if $D(\mathbf{x}, \mathbf{x}'_j) < \lambda_j$ **then**
      return *label of* $\mathbf{x}'_j$;
    end
  until $j = n$;
  $\hat{x} \leftarrow \arg\min_x (D(x, x') - r_x)$ (find nearest hypersphere's centre);
  return *label of* $\hat{x}$;
end

similar class of signals from the power line simultaneously. In such a case, even if the modem is capable of classifying signals based on their type like data, telephone, television, etc., in the absence of a unique identification code for each device, this bidirectional communication system will fail. Without a unique code, it is impossible for the modem to distinguish between similar types of signals directed towards similar devices. An identification scheme is also required, because if we look at the bigger picture, such universal access points will integrate existing communication models which widely use various addressing schemes for effective routing and duplex communication. It is only natural to assign a similar addressing scheme to devices connected through the power line network, which does not exist in the current state-of-the-art.

Of all the existing communication networks, the one that comes closest to power line networks in pervasiveness and usage is the Internet [16]. Thus, if we are to come up with an addressing scheme for power line networks, care must be taken to see that this addressing scheme is as compatible as possible to the Internet addressing scheme to ensure a smooth transition. Seemingly, the easiest way to achieve this is to borrow the *IP* address allocation scheme over to the power line networks. *IP* addressing scheme, especially the newly formed *IPv6* scheme, is capable of accommodating large number of nodes and seems a logical first choice to act as an addressing scheme for *GSCC*-based *UCM*s.

4.4.1 Addressing Schematic

We propose an innovative, low processing requirement, high throughput *IPv6* (*IPv4* compatible) addressing module for *UCM*. The module allows for parallel processing of the addressing operations by inhibiting embedded multiple memory modules, which is an essential element for multiple types of different devices to communicate through the same communication medium. The module also allows for even allocation of the *IPv6* addresses to the multiple parallel memory modules ensuring optimum memory use. It also balances the addressing traffic load among parallel processors, to achieve maximum throughput. The block diagram of the duplex communication module is shown in Fig. 4.3.

In a situation where bidirectional communication is required, a dynamic *IPv6* address is assigned to the respective device. To classify the *IPv6*, we extract the 10th–13th bits as its *ID* and pass it to the Index Assignment (*IA*) block. The *IA* block consists of parallel processors, which are assigned the group of prefixes of the *IPv6* address. It outputs a 3 bit Addressing Number (*AN*), indicating the processor and which block inside each processor will have the matching *IPv6* address. The Priority Assignment (*PA*) block receives the *AN*, and from it selects the processor with the shortest processing load. It further sends the assigned address to that particular processor and also assigns it an *n* bit tag for tracking. The tagged *IPv6* address is then passed to an Aligning Block (*AB*). As we have multiple bidirectional communication devices, the incoming *IPv6* address will leave the output block in a different order from its input. The *AB* will ensure that the address exits in the same sequence as it was inputted (*MIMO*), by using the attached tags to align the output sequence.

Illustrating the above algorithm with an example, let us assume a Dynamic *IPv6* address is assigned as 220.227.216.144. The corresponding ID is generated as "1100" (12) and sent to the *IA* block, which further compares it with different indexes on parallel processors and assigns respective *AN*s. Suppose we have 4 parallel processors and the *AN*s returned are "010" (2), "'001" (1), "011" (3) and "110" (6). This implies that the first 3 processors have *AN*s corresponding to the *ID* while the 4th does not. They are further sent to the *PA* block where let us assume that the counters are 14, 06 and 09 in processors 1, 2 and 3 respectively. We select the processor 3 for its lowest count value. The *IPv6* address and its tag are then sent into the *MIMO* list corresponding to processor 3. When required for addressing, the *IPv6* address is pulled out from the list. The *AB* sends out the final result in the original order as per the tag attached.

4.4.2 Throughput Evaluation

We further constitute the model for evaluating the efficiency of the Bidirectional Communication module by assuming the allocation rate as λ and a constant service rate μ with service time $T_s = 1/\mu$. The buffer is loaded with traffic load $D(t)$, which

yields the traffic intensity as $\rho_t = \frac{\lambda \times D(t)}{\mu}, t = 1, \ldots, k$. We assume that the number of *IP* addresses in queue at ith time is $\{Q_i\}_{i=1}$ and we define the loss probability γ, throughput of the modem β_t and throughput of the system S as (4.18–4.21).

$$\alpha_j(\rho_t) = \sum_{i+m=j-2} \frac{e^{\rho_t(i=1)}(-1)^m \rho_t^m (i+1)^m}{m!} (j > 2) \tag{4.18}$$

$$\gamma_t = \frac{1 + (\rho_t - 1)\alpha_n \rho_t}{1 + \rho_t \alpha_n \rho_t} \tag{4.19}$$

$$\beta_t = \lambda_t(1 - \gamma_t) = \frac{\lambda_t \alpha_n(\rho_t)}{1 + \rho_t \alpha_n(\rho_t)} = \frac{\lambda \times D(t) \times \mu \times \alpha_n \left(\lambda \times \frac{D(t)}{\mu}\right)}{\mu + \lambda \times D(t) \times \alpha_n \left(\lambda \times \frac{D(t)}{\mu}\right)} \tag{4.20}$$

$$S = \sum_{t=1,\ldots,k} \frac{\lambda \times D(t) \times \mu \times \alpha_n \left(\lambda \times \frac{D(t)}{\mu}\right)}{\mu + \lambda \times D(t) \times \alpha_n \left(\lambda \times \frac{D(t)}{\mu}\right)} \tag{4.21}$$

We can hence evaluate the best performance of the proposed scheme by evaluating the worst instance scenario S_w as in (4.22) where we have $D(t) = \frac{1}{K} \pm \frac{1}{2^p}$.

$$\begin{aligned} S_w = & \frac{\lambda \times \left(\frac{1}{K} + \frac{1}{2^p}\right) \times \mu \times \alpha_n \left(\lambda \times \frac{\left(\frac{1}{K} + \frac{1}{2^p}\right)}{\mu}\right) \times K/2}{\mu + \lambda \times \left(\frac{1}{K} + \frac{1}{2^p}\right) \times \alpha_n \left(\lambda \times \frac{\left(\frac{1}{K} + \frac{1}{2^p}\right)}{\mu}\right)} \\ & + \frac{\lambda \times \left(\frac{1}{K} - \frac{1}{2^p}\right) \times \mu \times \alpha_n \left(\lambda \times \frac{\left(\frac{1}{K} - \frac{1}{2^p}\right)}{\mu}\right) \times K/2}{\mu + \lambda \times \left(\frac{1}{K} - \frac{1}{2^p}\right) \times \alpha_n \left(\lambda \times \frac{\left(\frac{1}{K} - \frac{1}{2^p}\right)}{\mu}\right)} \end{aligned} \tag{4.22}$$

Illustrating the above with an example, if we assume the *UCM* to have 133 Mhz memory addressing capacity, ($K = 4$, $\mu = 133$ Mpps) and supply the system with a heavy load ($\lambda = 4$) with *ID* bits ($P = 4$), the worst case scenario (S_w) is given by Eq. (4.22) as $S_w = 3.49 \mid \mu = 464$ Mpps.

4.5 Experimental Analysis: *UCM*

UCM is tested in real time and comparison analysis is performed with existing power line *OFDM* and *FMT* modems used for broadband transmission as described in [6–8, 11]. The modulation parameters are defined in Table 5.1, where we adapt the *OFDM* modulation scheme. The experimental setup consists of two *UCM*'s connected on a power line of 240 v AC current, with a distance of 500 m between them.

The devices connected to each *UCM* are 2 desktop computers, 2 telephones and one television. Other appliances connected to the power line were an air conditioning unit, microwave oven and a refrigerator to induce impulsive noise on the power line. Test datasets of about 5 GB each of telephonic audio, television video and broadband data were randomly transmitted through *UCM* over the power line. While creating the dataset, care was taken to cover the entire frequency spectrum spanned by the various communication standards. The experiments are conducted for 3 scenarios, namely measurement of throughput for various frame lengths, overload of the network at 110% capacity and then reducing the load to 70%.

First, we test the performance of the proposed modules individually. The channel estimation and signal processing and classification modules are tested for the collected data and classification results are shown in Figs. 4.4, 4.5 and 4.6 and enumerated in Table 4.2. Audio, video and data classes are represented as red, blue and green respectively. The average accuracy of classification obtained is about 97% for the proposed approach.

Further we test the bidirectional communication module for the throughput it supports with respect to the traffic intensity as shown in Fig. 4.7. Throughput is well within the acceptable range at 94 and 97% for the telephonic and data bidirectional communications at 70% load conditions. Finally we compare the performance of UCM under various situations and also with the existing state-of-the-art [3, 6, 7, 17] as shown in Fig. 4.8. Performance results are enumerated at 30 db E_s/N_o in Tables 4.1, 4.2 and 4.3.

In this chapter, we define a first-of-its-kind *UCM*, designed on the postulates of *GSCC* paradigm aiming towards unification of the cloud communications structure. The *UCM* allows multiple devices and multiuser communications over a single channel evolving a true unified communication cloud. We present three judicious designs for channel estimation, signal classification and bidirectional communication

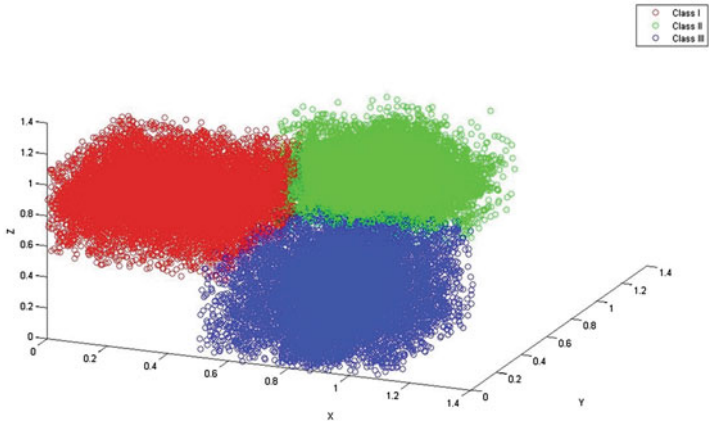

Fig. 4.4 Test results of data set I comprising of three different classes, through *OCENN* with average train error = 0.0146 and average test error = 0.0195

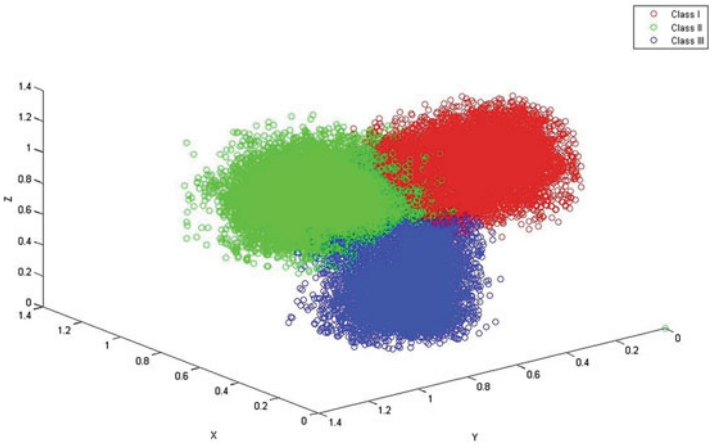

Fig. 4.5 Test results of data set II comprising of three different classes, through *OCENN* with average train error = 0.018 and average test error = 0.028

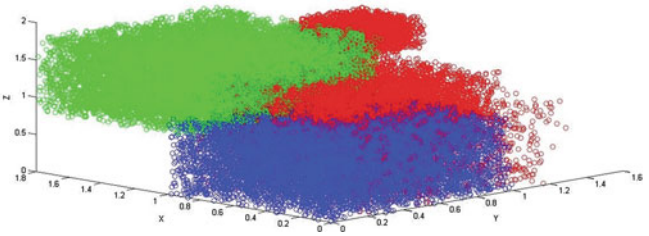

Fig. 4.6 Test results of data set III comprising of three different classes, through *OCENN* with average train error = 0.015 and average test error = 0.032

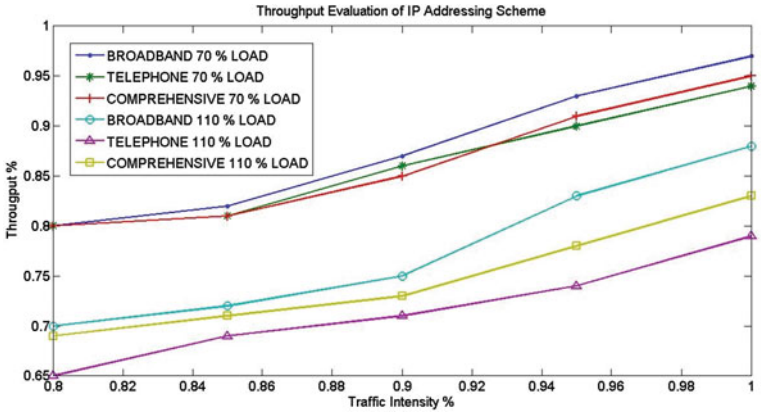

Fig. 4.7 Throughput analysis of the duplex communication facilitated through *UCM*

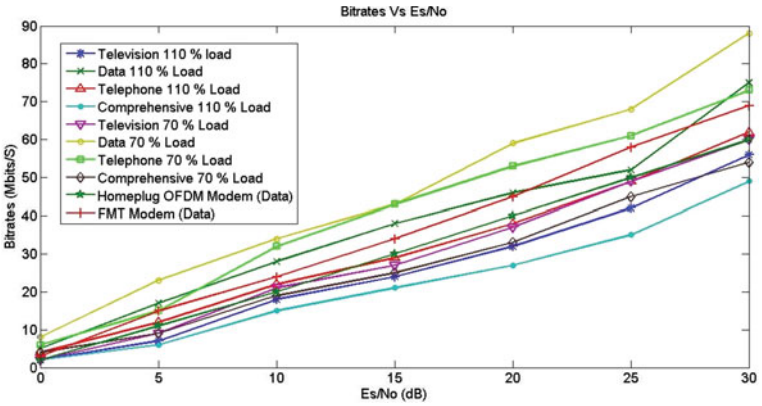

Fig. 4.8 *SINR* and bitrate analysis with varying load profiles for different types of signals sent through the *UCM*

Table 4.1 Main modulation parameters of *UCM* tested over a low voltage power line acting as a common medium for transmission of different categories of signals

Sampling period (T_s)	0.05 μs
Number of sub-carriers (N)	2048
Guard interval duration	15 μs $= 300T_s$
DFT block duration	102, 4 μs $= 2048T_s$
OFDM symbol duration T_{OFDM}	117, 4 μs $= 2348T_s$

Table 4.2 Classification of communication signals after processing through the *OCENN* module

#	Channel estimation and signal classification module (%)		
	Accuracy	False negatives	Class overlap region
1	97.15	10.76	9.83
2	97.62	10.22	8.69

Table 4.3 Comparative analysis of *UCM* with existing state-of-the-art modems capable of low voltage power line transmission. 4 different datasets each of 1 GB size of various signals at 110 and 70% load profiles are tested on *UCM*

$E_s/N_o = 30$ dB	110% power load on CMs				70% power load on *CM*s				OFDM data modem	FMT data modem
	A	B	C	D	A	B	C	D		
Bitrate (Mbps)	56	75	62	49	60	88	73	54	60	69

together constituting the architecture of *UCM*. Performance results and comparison with existing state-of-the-art shows *UCM* as a viable prospective archetype. Currently, *UCM* is only tested on a low voltage scenario, and future directional efforts would include its implementation on the higher voltage power lines. Limiting not only to power lines, to the best of our knowledge, such a universal modem does not exist on any of the existing communication channels and can be migrated to any standard without loss of generality.

The next step entails detecting or depicting the behaviour of communication signals while traversing collectively and simultaneously in a singular medium. To this effect, the next chapter establishes a channel model specifically designed for *UCM* and the *GSCC* paradigm operating over the power lines infrastructure. Since the proposed channel model aims to cover all multiple communication standards over the same infrastructure, its design should be devoid of dependency on the topology of the architecture and use universal parameters like voltage and impedance. We aim to overcome the deficiencies of the existing channel model with the proposed formulation.

References

1. A.M. Tonello and F. Pecile. A filtered multitone modulation modem for multiuser power line communications with an efficient implementation. In *Power Line Communications and Its Applications, 2007. ISPLC '07. IEEE International Symposium on*, pages 155–160, 2007.
2. Xingyuan Fang and Cheng Tao. Design and implementation of broadband plc modem. In *Signal Processing, 2006 8th International Conference on*, volume 4, pages –, 2006.
3. Yi-Fu Chen and Tzi-Dar Chiueh. A 100-kbps power-line modem for household applications. In *VLSI Technology, Systems, and Applications, 1999. International Symposium on*, pages 179–182, 1999.
4. S. Ramseler, M. Arzberger, and A. Hauser. Mv and lv powerline communications: new proposed iec standards. In *Transmission and Distribution Conference, 1999 IEEE*, volume 1, pages 235–239 vol.1, 1999.
5. J. Yazdani, H. Wegert, P. Brown, and B. Honary. Effect of powerline transmissions on hf channels. In *Frequency Selection and Management Techniques for HF Communications (Ref. No. 1999/017), IEE Colloquium on*, pages 7/1–7/7, 1999.
6. S. Gault, P. Ciblat, and W. Hachem. An ofdma based modem for powerline communications over the low voltage distribution network. In *Power Line Communications and Its Applications, 2005 International Symposium on*, pages 42–46, 2005.
7. K.L. Heo, S.M. Cho, J.W. Lee, M.H. Sunwoo, and Seong Keun Oh. Design of a high speed ofdm modem system for powerline communications. In *Signal Processing Systems, 2002. (SIPS '02). IEEE Workshop on*, pages 264–269, 2002.
8. S. Souissi, A. Ben Dhia, F. Tlili, and C. Rebai. Ofdm modem design and implementation for narrowband powerline communication. In *Design and Technology of Integrated Systems in Nanoscale Era (DTIS), 2010 5th International Conference on*, pages 1–4, 2010.
9. F.J. Cortes, E.M. Rubio, and A. Valdovinos. Embedded powerline dsp modem for domotic snmp networking in european countries. *Consumer Electronics, IEEE Transactions on*, 48(4):854–862, 2002.
10. B. Jensen. Experimental studies of the noise recovery ability of in-house powerline equipment. In *Power Line Communications and Its Applications, 2008. ISPLC 2008. IEEE International Symposium on*, pages 269–273, 2008.

11. A.M. Tonello. Gen01-1: A wide band modem based on impulse modulation and frequency domain signal processing for powerline communication. In *Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE*, pages 1–6, 2006.
12. M. Tucci, M. Raugi, A. Musolino, and S. Barmada. Blind channel estimation for power-line communications by a kohonen neural network. In *Power Line Communications and Its Applications, 2007. ISPLC '07. IEEE International Symposium on*, pages 35–40, 2007.
13. H.D. Mustafa, C.S. Rao, S.N. Merchant, and U.B. Desai. A novel algorithm for multiple signal classification with optimized coulomb energy neural networks for power line communications. In *Power Line Communications and Its Applications (ISPLC), 2010 IEEE International Symposium on*, pages 90–95, 2010.
14. C.L. Scofield. Learning internal representations in the coulomb energy network. In *Neural Networks, 1988., IEEE International Conference on*, pages 271–276 vol.1, 1988.
15. T. Olmez, E. Yazgan, and O.K. Ersoy. Modified restricted coulomb energy neural network. *Electronics Letters*, 29(22):1963–1965, 1993.
16. P. Gupta, S. Lin, and N. McKeown. Routing lookups in hardware at memory access speeds. In *INFOCOM '98. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1240–1247 vol.3, 1998.
17. R. Araneo, S. Celozzi, and G. Lovat. Design of impedance matching couplers for power line communications. In *Electromagnetic Compatibility, 2009. EMC 2009. IEEE International Symposium on*, pages 64–69, 2009.

Chapter 5
GSCC Channel Characterization and Modelling

Abstract The aim of *GSCC* architecture is to facilitate a universal medium for varied signal transmission creating a cloud of communications. A power line communication *(PLC)* system coupled with the *UCM* as presented in the previous chapter allows the possibilities of not only utilizing the existing most pervasive and dense infrastructure, the electric power lines, but also amalgamates the entire communication archetype into a singular paradigm. An essential constituent of this universal communication system is the estimation of its channel characteristics and how this channel could be effectively used with the GSCC Paradigm. With renewed interest over the last decade in *PLC*, quite a few channel models have been reported in literature. However, these models are highly topology specific and suffer from various drawbacks like the use of voltage as a parameter, requirement of the a priori knowledge of the network and knowledge of load impedances. In this chapter, the authors propose a robust, non-parametric, magnetic field intensity-based channel model for non-invasive, multipath channel estimation for operations of GSCC Paradigm over power lines. The novel model accounts for the parameters of the signal like quality, strength and attenuation by simplistic inference from the measured magnetic field intensity. The true essence of ubiquity of power lines is amalgamated with the GSCC architecture evolving the next generation cloud communications infrastructure.

The aim of *GSCC* architecture is to facilitate a universal medium for varied signal transmission. A power line communication (*PLC*) system coupled with the *UCM* as presented in the previous chapter allows the possibilities of not only utilizing the existing most pervasive and dense infrastructure but also amalgamates the entire communication archetype into a singular paradigm [1, 2]. An essential constituent of this universal communication system is the estimation of its channel characteristics. It characterizes an integral element in determination of costs, ease of use and practicability of the architecture. Further, the power line channel owing to its variable link topologies, uncanny wiring practices and its aptness only for transmission of low frequency high voltage signals, demands for a robust channel modeling. With renewed interest over the last decade in *PLC*, quite a few channel models have been reported in literature. Majority of the models provide the description of the channel by a selected few parameters, viz. voltage and current [3–5]. These models assemble

H.D. Mustafa et al., *Green Symbiotic Cloud Communications*,
DOI 10.1007/978-981-10-3512-8_5

matrices which comprehend properties of the associated parameters. Generally, such requirement stresses on the a priori knowledge of network topology, which cannot be practically accumulated for the dense power line infrastructure. Some models are based on statistical categorizations which do not allow physical understanding, lack pragmatic rationalization and are very difficult to reproduce or generalize [6–10]. In modulation specific systems like *OFDM*, *DMT*, etc., channel estimation is inferred by measuring a distorted channel frequency response, or by sending pilot symbols, which in turn is based on several assumptions [11–14]. The time domain channel models require measurement of actual signal propagation paths individually, which for a power line network is an unconceivable task [12, 15]. In frequency domain models, computationally tedious transfer functions are obtained yielding complex-valued parameters for attenuation, noise and channel delay [3, 5, 9, 11, 14].

In our efforts towards this attainment, we propose a robust, non-parametric, magnetic field intensity-based channel model for non-invasive, multipath channel estimation for operations of *UCM* over low voltage power lines. The novel model accounts for the parameters of the signal like quality, strength and attenuation by simplistic inference from the measured magnetic field intensity. To the best of our knowledge, it is a first-of-its-kind non-parametric field intensity-based channel model, which characterises the channel conditions for communication signals from the differential magnetic field intensity over the power line. The drawbacks of the existing power line channel models for characterizing high frequency communication signals sent over them are enlisted in the subsequent section which forms the basis for our proposed model.

5.1 Existing Channel Models: Drawbacks

5.1.1 Voltage as a Parameter

All the current available models use voltage as a parameter in defining their channel characteristics [3–5, 9, 11, 14]. The methods thus depend on measurement of voltages over the power line for establishing the channel model. Now, the very basic requirement for measuring a voltage over any kind of line is physical contact between the conductor and the measuring instrument. Thus, if we require to measure voltage over a line, we need to remove the outer covering or shield of the line in order to make physical contact between the conductor and the measuring instrument (in this case, a voltmeter). Since most of the lines used in low voltage have one or two protective insulation coverings over the conductor, removing these coverings may cause a temporary or a permanent fault in the line. Thus, except for the locations at the source and the load, the task of measuring voltage along the power line is quite cumbersome and tedious.

5.1.2 Knowledge of the Network

Secondly, if we notice the transfer functions of the existing models, we observe that the models are dependent on the network topology [9–11, 16]. Network topology in this case would mean the knowledge of the number of junctions/nodes in the network, number of branches coming out from each junction/node, length of each branch and finally, the amount of loads connected to each branch. For a typical power line network, this becomes quite problematic, since the network may keep on changing from one instance to another.

To further emphasize on this drawback, let us consider a scenario in which an in-house power line network is considered. Say at time '$t = 0$' the loads connected on the line are AC, refrigerator, television and 2 fans. Assuming that at '$t = 0$', we are aware of the entire network topology. Now, let us say at an instance '$t = t_1$', a person in the house switches on the microwave and a mixer grinder. Now, because of this change, the line connecting the microwave and the mixer grinder to the main line has become active and is now to be included in the set of active branches of the network. Thus, the network topology has changed and we need to update our knowledge of the network.

5.1.3 Load Impedance

The existing channel models over power lines depend on the knowledge of load impedances which are connected to the network [5, 9, 16, 17]. Computation of the model requires the prior knowledge of impedances of the loads. Since the number of loads over the *PLC* network changes from one instance to another, keeping information about the number of loads connected at a particular time is near to impossible. This condition again presents itself as a major roadblock for channel model computation over an ever-changing *PLC* network.

The significant drawbacks of the existing models that we overcome are:

1. Existing models consider the propagating current over the power lines as background noise. This makes the existing models unpractical as it colossally decreases the SNR, giving inaccurate channel conditions. The proposed model considers the propagating current as itself, thus overcoming this major drawback.
2. The existing methods entail invasion of the power lines for measurement of parameters for channel estimation, rendering the models useful only at line ends. The proposed model is non-invasive, enabling channel estimation at all points over the power line.
3. The proposed channel model accommodates the impulsive noise from the transients (cables, joints, connected devices, etc.) on the network. In the existing models, back-tracing of measurement results to channel characteristics is not possible.

4. The existing models either neglect multipath effects or consider each path individually. While the former results in inaccurate assumptions, the latter gives computationally complex and different transfer functions for each path. The proposed model is independent of multipath effects and thus vouches for its strong candidature.

5.2 Theoretical Modelling: Channel Model

We begin with the basic transmission line equations for voltage and current waveforms [9]. The voltage and current waveforms over the power line could be represented in terms of second differential equations:

$$\frac{\partial^2 V(z)}{\partial z^2} = \gamma_1^2 V(z) \tag{5.1}$$

$$\frac{\partial^2 I(z)}{\partial z^2} = \gamma_1^2 I(z) \tag{5.2}$$

Here, $V(z)$ and $I(z)$ represent voltage and current waveforms respectively, varying with distance in the direction of propagation 'z'. $\gamma_1 (= \alpha_1 + j\beta_1)$ is the propagation constant, α_1 and β_1 are the attenuation and the phase constants respectively over the line. The propagation constant, γ_1, is dependent on the line parameters as shown by the following equation:

$$\gamma_1 = \sqrt{(R + j2\pi f L)(G + j2\pi f C)} \tag{5.3}$$

where, R, G, L and C are the resistance per unit length, conductance per unit length, inductance per unit length and capacitance per unit length respectively over the power line. Solving Eqs. (5.1) and (5.2) leads to Eqs. (5.4) and (5.5), which represent voltage and current waveforms in terms of second order homogenous equations:

$$V(z) = V^+ e^{-\gamma_1 z} + V^- e^{+\gamma_1 z} \tag{5.4}$$

$$I(z) = \frac{V^+}{Z_o} e^{-\gamma_1 z} - \frac{V^-}{Z_o} e^{+\gamma_1 z} \tag{5.5}$$

where, Z_o is the characteristic impedance.

From Eqs. (5.4) and (5.5) it is clear that the voltage and current waveforms are dependent on distance 'z'. The two terms $V^+ e^{-\gamma_1 z}$ and $V^- e^{+\gamma_1 z}$ represent a forward travelling and a backward travelling wave respectively. The dependence on time could be incorporated by multiplying the equations by $e^{j\omega t}$, giving (5.6) and (5.7):

$$V(z, t) = V^+ e^{-\alpha_1 z} e^{j(\omega t - \beta_1 z)} + V^- e^{+\alpha_1 z} e^{j(\omega t + \beta_1 z)} \tag{5.6}$$

$$I(z,t) = \frac{V^+}{Z_o} e^{-\alpha_1 z} e^{j(\omega t - \beta_1 z)} - \frac{V^-}{Z_o} e^{+\alpha_1 z} e^{j(\omega t + \beta_1 z)} \tag{5.7}$$

The behaviour of electric and magnetic field in a medium is portrayed by Maxwell's equations [7, 13]. These equations can be represented in the form of second order differential equations as follows:

$$\frac{\partial^2 E_x(z)}{\partial z^2} = \gamma_2^2 E_x(z) \tag{5.8}$$

$$\frac{\partial^2 H_y(z)}{\partial z^2} = \gamma_2^2 H_y(z) \tag{5.9}$$

Equations (5.8) and (5.9) represent a transverse electromagnetic wave (*TEM* wave), where the directions of the electric field and the magnetic field, and the direction of propagation are mutually perpendicular to each other. The electric field is assumed to be in the 'x' direction, the magnetic field in the 'y' direction, while the direction of propagation is considered to be in the 'z' direction. γ_2 represents the propagation constant which is dependent on the medium parameters ω, μ_o, μ_r, ϵ_o, and ϵ_r. Here ω, μ_o, μ_r, ϵ_o, and ϵ_r represent the frequency of the wave, permeability of free space, relative permeability of the medium, permittivity of free space and relative permittivity of the medium respectively. Solving Eqs. (5.8) and (5.9), one obtains homogeneous equations, (5.10) and (5.11)

$$E_x(z) = E_x^+ e^{-\gamma_2 z} + E_x^- e^{\gamma_2 z} \tag{5.10}$$

$$H_y(z) = \frac{\gamma_2}{\omega\mu} \left[E_x^+ e^{-\gamma_2 z} - E_x^- e^{\gamma_2 z} \right] \tag{5.11}$$

In a two conductor PLC system, the potential difference (or voltage) is the amount of work done by the electric field. Thus, voltage and electric field are related to each other. This relationship between voltage and electric field is given by Eq. (5.12)

$$V(z) = \int_{x=0}^{a} E_x(z) dx \tag{5.12}$$

Here, electric field is assumed to be in the 'x' direction, while the direction of propagation is along the 'z' direction. 'a' is the distance between the two conductors. On solving (5.12), we get (5.13)

$$V(z) = a E_x(z) \tag{5.13}$$

Now, using Eqs. (5.4) and (5.10), we get (5.14)

$$V^+ e^{-\gamma_1 z} + V^- e^{+\gamma_1 z} = a \left(E_x^+ e^{-\gamma_2 z} + E_x^- e^{+\gamma_2 z} \right) \tag{5.14}$$

Observing Eq. (5.14), we find that both the quantities on the left hand as well as the right hand side represent the voltage waveform as a combination of a forward travelling wave and a backward travelling wave. Equating the forward and the backward travelling parts of the waveforms on both sides, we get Eqs. (5.15) and (5.16)

$$E_x^+ = \frac{V^+ e^{(\gamma_2 - \gamma_1)z}}{a} \tag{5.15}$$

$$E_x^- = \frac{V^- e^{(\gamma_1 - \gamma_2)z}}{a} \tag{5.16}$$

Dividing Eqs. (5.15) by (5.16), we get Eq. (5.17)

$$\frac{E_x^+}{E_x^-} = \frac{V^+}{V^-} e^{2(\gamma_2 - \gamma_1)z} \tag{5.17}$$

In Eq. (5.4), we represented the voltage waveform as a combination of a forward and a backward travelling wave. The formation of the forward travelling wave is self-explanatory; it is the wave which is generated by the voltage source connected at the point '$z = 0$' over the power line. The origin of the backward travelling may not be quite clear at this stage. One can then argue that the backward travelling wave must be because of the reflection of the forward wave from the load point. Thus, a signal over the power line undergoes multiple reflections depending on the loads and junctions in the network. At this point, we define a parameter which gives the relative amplitudes of the two waves, forward and backward, at any point on the line.

The parameter that relates the forward and the backward travelling wave at any point over the power line is known as the reflection coefficient, denoted by the symbol $\Gamma(z)$ [5]. The reflection coefficient at any point 'z' over the power line is defined by Eq. (5.18)

$$\Gamma(z) = \frac{\text{Reflected signal at } 'z'}{\text{Incident signal at } 'z'} = \frac{V^- e^{\gamma z}}{V^+ e^{-\gamma z}} = \frac{V^-}{V^+} e^{2\gamma_1 z} \tag{5.18}$$

Here $V^- e^{\gamma z}$ represents the reflected wave and $V^+ e^{-\gamma z}$ represents the incident wave. We use the reflection coefficient to further advance on our theory for establishing the channel model. From Eq. (5.17), we notice that the ratio $\frac{E_x^+}{E_x^-}$ is related to the ratio $\frac{V^-}{V^+}$ which in turn is related to the reflection coefficient. Thus, using Eqs. (5.17) and (5.18), we get (5.19)

$$\frac{E_x^+}{E_x^-} = \frac{1}{\Gamma(z)} e^{2\gamma_2 z} \text{ or } E_x^- = \Gamma(z) E_x^+ e^{-2\gamma_2 z} \tag{5.19}$$

Using this relation between E_x^- and E_x^+ in the equation for the magnetic field (5.11), we get (5.20)

$$H_y(z) = \frac{\gamma_2}{\omega\mu} E_x^+ e^{-\gamma_2 z}[1 - \Gamma(z)] \tag{5.20}$$

Consider the transmission line scenario, where a voltage source V_s is connected to the line, ending into a load Z_L. Let us assume that L is the length of the line and point 'z' is any point of interest for us over the line. Let us denote impedance over the power line at the point 'z' as $Z(z)$, $I(z)$ is the current flowing over the line at the point 'z' and $V(z)$ is the voltage at the point 'z'. Using simple Kirchoff's law equations, we get the current travelling over the line at the point 'z' by Eq. (5.21)

$$I(z) = \frac{V_s}{Z_s + Z(z)} \tag{5.21}$$

and the voltage at point 'z' is given by (5.22)

$$V(z) = Z(z)I(z) = \frac{Z(z)V_s}{Z_s + Z(z)} \tag{5.22}$$

Equation (5.18) gives us the relation between V^- and V^+ in terms of the reflection coefficient $\Gamma(z)$, which could be further modified to obtain (5.23)

$$V^- = V^+\Gamma(z)e^{-2\gamma_1 z} \tag{5.23}$$

Using this relation in the equation for $V(z)$, we get Eq. (5.24) of $V(z)$, which is only in terms of V^+

$$V(z) = V^+e^{-\gamma_1 z}\left[1 + \Gamma(z)\right] \tag{5.24}$$

Equations (5.22) and (5.24) both represent voltage over the transmission line at the point 'z'. The next natural step would be to equate these two equations and obtain an expression for V^+. Thus, equating (5.22) and (5.24), we get (5.25)

$$V^+ = \frac{Z(z)V_s e^{\gamma_1 z}}{\left[Z_s + Z(z)\right]\left[1 + \Gamma(z)\right]} \tag{5.25}$$

Next we use the equation for V^+ to find an expression for E_x^+. We use Eq. (5.15), which relates E_x^+ and V^+ to arrive at (5.26)

$$E_x^+ = \frac{Z(z)V_s e^{\gamma_2 z}}{a\left[Z_s + Z(z)\right]\left[1 + \Gamma(z)\right]} \tag{5.26}$$

Using the Eq. (5.26) for E_x^+ in Eq. (5.20) for the magnetic field, we get a new expression for the magnetic field at point 'z'. This expression is given by Eq. (5.27)

$$H_y(z) = \frac{\gamma_2 Z(z)V_s\left[1 - \Gamma(z)\right]}{a\omega\mu\left[Z_s + Z(z)\right]\left[1 + \Gamma(z)\right]} \tag{5.27}$$

Impedance at a point 'z', $Z(z)$, could be represented in terms of the reflection coefficient at 'z' by the following Eq. (5.28)

$$Z(z) = Z_o \left[\frac{1 + \Gamma(z)}{1 - \Gamma(z)} \right] \tag{5.28}$$

Using Eq. (5.28) in Eq. (5.27) for the magnetic field, we get Eq. (5.29), representing the magnetic field at a distance 'z' over the power line

$$H_y(z) = \frac{Z_o V_s \gamma_2 [1 - \Gamma(z)]}{a\omega\mu [Z_s(1 - \Gamma(z)) + Z_o(1 + \Gamma(z))]} \tag{5.29}$$

In Eq. (5.29), we observe that the quantities Z_o, V_s, γ_2, a, ω and μ are the terms which are independent of the distance 'z', hence we club them into one coefficient which is denoted by $A(f)$. $A(f)$ is given by Eq. (5.30)

$$A(f) = \frac{\gamma_2 Z_o V_s}{a\omega\mu} \tag{5.30}$$

And the expression for the magnetic field thus at point 'z' is given by Eq. (5.31)

$$H_y(z) = \frac{A(f)[1 - \Gamma(z)]}{[Z_s(1 - \Gamma(z)) + Z_o(1 + \Gamma(z))]} \tag{5.31}$$

In order to find an expression for the channel transfer function, we relate the magnetic field intensity at the point 'z' with the magnetic field intensity at the point '$z = 0$'. Thus, now we find an expression for the magnetic field intensity at the point '$z = 0$', i.e. at the source end. Putting '$z = 0$' in the expression for the magnetic field intensity, Eq. (5.31), we get (5.32)

$$H_y(0) = \frac{A(f)[1 - \Gamma(0)]}{[Z_s(1 - \Gamma(0)) + Z_o(1 + \Gamma(0))]} \tag{5.32}$$

At the source end, we assume that the entire voltage is being transmitted and no part is reflected back to the source (i.e., matching condition holds true). Hence, the value of the reflection coefficient is '0' at the point '$z = 0$', and we get Eq. (5.33)

$$H_y(0) = \frac{A(f)}{[Z_s + Z_o]} \tag{5.33}$$

We observe that the magnetic field intensity at the point '$z = 0$' is dependent only on the $A(f)$ coefficient, the source impedance and the characteristic impedance. We obtain the transfer function (5.34) using the expressions for $H_y(z)$ (Eq. 5.31) and $H_y(0)$ (Eq. 5.33).

$$h(f) = \frac{H_y(z)}{H_y(0)} = \frac{(Z_s + Z_o)[1 - \Gamma(z)]}{[Z_s(1 - \Gamma(z)) + Z_o(1 + \Gamma(z))]} \tag{5.34}$$

Equation (5.34) represents the final transfer function for the channel model of *UCM* over a low voltage *PLC* network.

At this point, when we have the knowledge about the transfer function, we relate the transfer function with the reflection coefficient. The receiver would then receive this parameter as an input and would decide about the signal strength/quality. Consider the magnitude of the transfer function. The transfer function is the ratio of the magnetic field intensity at point '$z > 0$' and the magnetic field intensity at point '$z = 0$'. As the signal moves along the line away from the source because of channel attenuation, noise and reflections, it is bound to lose energy. Thus, we could safely state that the signal, upon reaching any point '$z > 0$', would have a magnetic field intensity value which would be less than the magnetic field intensity value at point '$z = 0$'. Thus, the magnitude of the transfer function would always be <1. We take the magnitudes of the transfer function as well as the reflection coefficient term. Since Z_s and Z_o are terms which are already real and greater than '0', we do not need to take their magnitudes. Now, using Eq. (5.34) to find the expression for the reflection coefficient in terms of the transfer function, we get Eq. (5.35)

$$\Gamma(z) = \frac{[(Z_s + Z_o)(1 - h(f))]}{[(Z_o - Z_s)h(f) + (Z_o + Z_s)]} \tag{5.35}$$

Since the transfer function varies between 0 and 1, the reflection coefficient would also vary between 0 and 1. We observe from Eq. (5.35) that the reflection coefficient and the transfer function have an inverse relation with each other. That is to say if we plot the reflection coefficient vs. the transfer function, the graph would have a negative slope. The negative slope would be determined by the values of the source and the characteristic impedances.

The procedure of determining channel characteristics over the power line and signal transmission through *UCM* using the *GSCC* architecture is as follows:

- Measure the magnetic field intensity at the point where the signal strength is to be calculated.
- Measure the magnetic field intensity at the point '$z = 0$', i.e. the source end point of the power line.
- Divide the magnetic field intensity value at point 'z' with the magnetic field intensity at the point '$z = 0$', to obtain the transfer function. Note that the value of this transfer function has to be between 0 and 1, since the magnetic field intensity at any point '$z > 0$' would be less than the magnetic field intensity at the point '$z = 0$' for reasons explained in the previous section.
- Use the above calculated transfer function to calculate the value of the reflection coefficient.
- Calibrate the receiver according to the reflection coefficient value. Set the maximum value for the reflection coefficient, beyond which the signal becomes undetectable.

- If the value of the reflection coefficient at the point 'z' is more than the maximum set value, the signal strength/quality at the point 'z' is not enough for it to be detected by the receiver.
- Based on the measured values, compute the extra amount of energy required by the signal at the source end.
- The location of the repeater at a strategic distance from the source could also be estimated based on the above findings. The repeater would amplify the dying out signal, so that it is able to reach the desired destination point.

5.3 Theoretical Modelling: Channel Model

For verification of the proposed channel model, measurements were evaluated from an experimental bed of a 240 V power line. The length of the power line was fixed at 300 m with multiple loads connected at various points. Parameters of the test bed are enlisted in Table 5.1.

The magnetic field intensity along the wire was measured using a magnetic field meter. The meter is calibrated in the range of 0.1–3000 mG (0.01–300 μT) and has a sensitivity resolution of 0.1 mG (0.01 μT). A multipath network of power lines was considered with different loads connected at each end. The devices connected to the test bed include two desktop computers, two telephones, one television, one air conditioning unit, one microwave oven and one refrigerator. Test datasets of about 5 gigabytes each of telephonic audio, television video and broadband data were randomly transmitted through the power line in line with the distributed architecture of *GSCC*. While collecting the dataset, care was taken to cover the entire frequency spectrum spanned by the various categories. Furthermore, random impulsive noise was generated using the attached appliances. To establish a robust channel model we perform an exhaustive analysis with varied parameters.

Firstly we test the performance of individual parameters. Maintaining the other parameters over the network constant, the frequency profile of the test dataset is shown in Fig. 5.1 and enumerated in Table 5.2.

The proposed channel model accurately predicts the signal condition up to 97.48% accuracy on an average for the test dataset with varied frequency. The average

Table 5.1 Test bed parameters of the power line experimental bed equipped with *UCM* and *GSCC*

Analysis parameters	Analysis values
Resistance (R)	0.1 Ω/m
Inductance (L)	0.2 μH/m
Capacitance (C)	10 pF/m
Conductance (G)	0.02 ℧/m
Characteristic impedance (Z_o)	50 Ω

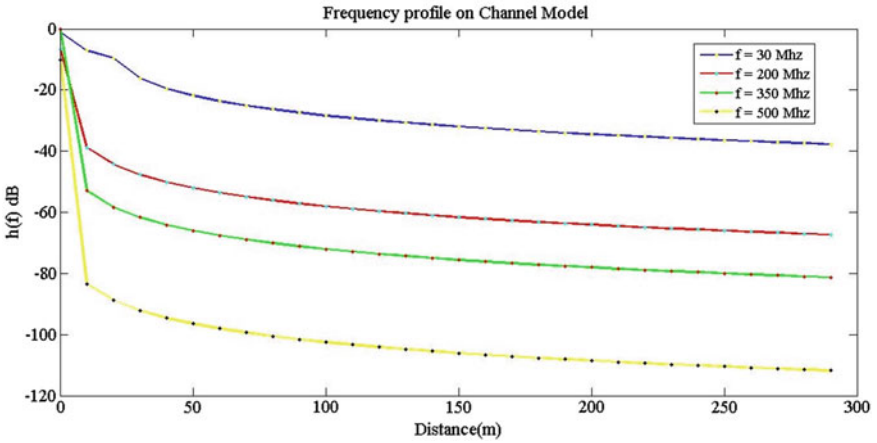

Fig. 5.1 Frequency profile of the channel model tested with signals of frequency from 30 to 500 Mhz

Table 5.2 Frequency profile of the channel model. Signals frequency range: 30–500 Mhz

Voltage = 240 V	Impedance level = medium		z = 150 mts
Frequency (MHz)	$\gamma(z)$		Accuracy (%)
	Theoretical	Measured	
30	0.39	0.40	97.42
200	0.42	0.43	97.61
350	0.43	0.44	97.67
500	0.45	0.46	97.77
Average	97.61		

Table 5.3 Load profiling of the channel model tested with loads varying from very high to low

Voltage = 240 V	Frequency = 200 MHz		z = 150 mts
Impedance	$\Gamma(z)$		Accuracy (%)
	Theoretical	Measured	
Very High	0.57	0.58	98.24
High	0.51	0.54	94.11
Medium	0.42	0.43	97.61
Low	0.38	0.38	100.00
Average	97.49		

frequency of the test dataset being 200 mHz, we measure the effect of varied load at this frequency as shown in Table 5.3. The load is characterized as very high, high, medium and low with a comprehensive profile. The results are documented

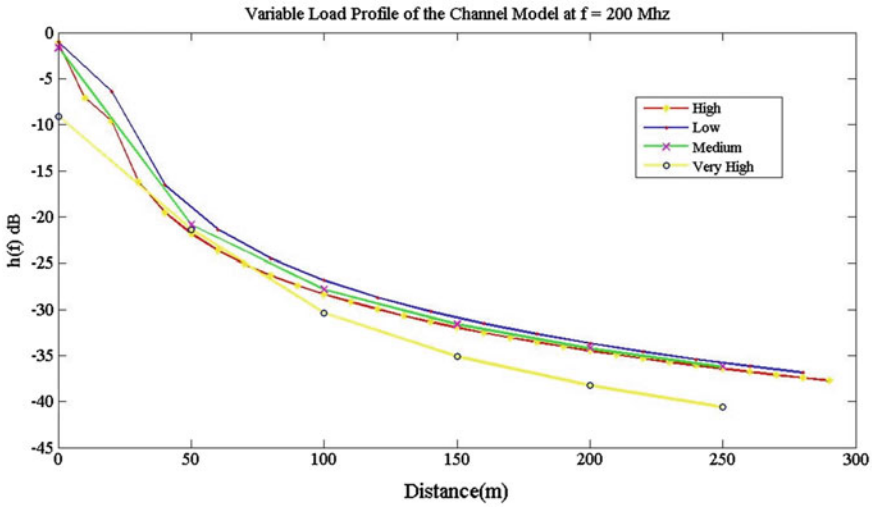

Fig. 5.2 Load profiling of the channel model tested with loads varying from very high to low

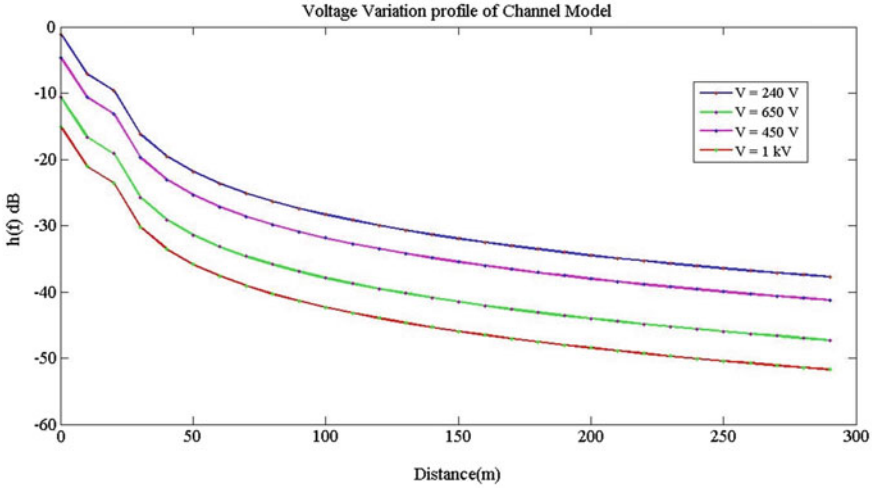

Fig. 5.3 Varying voltage profile over the power lines for channel model verification using *UCM*

graphically in Fig. 5.2 with an accuracy of 98.48%, thus enforcing our claims of a robust, noise-inclusive channel model.

Furthermore the proposed model is tested for a varied voltage profile at the average frequency. The results are shown in Fig. 5.3 and Table 5.4, with an accuracy of 96.42%, confirming universal compatibility of the channel model over the low voltage power line network.

Table 5.4 Varying voltage profile over the power lines for channel model verification using *UCM*

Frequency = 200 MHz	Impedance level = medium		z = 150 mts
Voltage (V)	$\Gamma(z)$		Accuracy (%)
	Theoretical	Measured	
240	0.42	0.43	97.61
450	0.46	0.45	97.82
650	0.49	0.47	97.95
1000	0.52	0.50	98.07
Average	97.86		

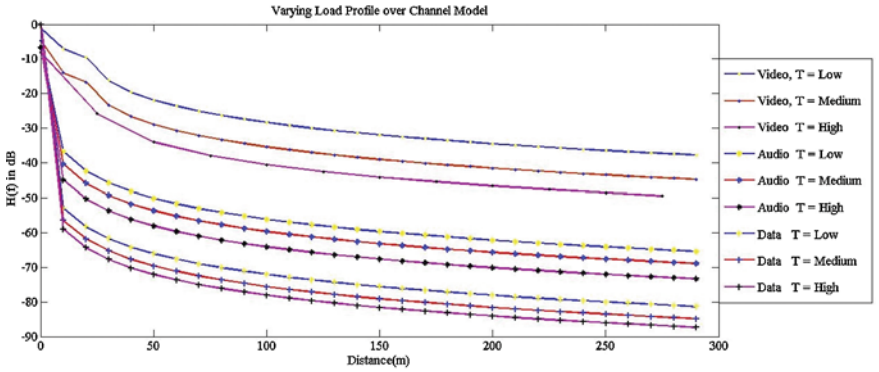

Fig. 5.4 Comprehensive varied parameter profiling on the channel model using the *UCM* for transmitting varied signals in a unified communication medium

Table 5.5 Comparative analysis of UCM GSCC Channel Model with existing power line paradigms

Frequency (MHz)	GT	M_1	M_2	M_3	UCM GSCC
200	0.42	0.37	0.45	0.38	0.43
350	0.43	0.41	0.47	0.42	0.45
500	0.45	0.49	0.46	0.41	0.46
1000	0.47	0.51	0.51	0.42	0.50
Mean error (%)	0.00	3.75	3.00	3.50	2.50

Finally, we perform a comprehensive analysis with varied frequency, voltage and load statistics for video, audio and a dataset of signals of different standards respectively. The results are shown in Fig. 5.4, and the accuracy is evaluated at 95.21%. We compare the results of the proposed model with the existing models, M_1[5], M_2[3] and M_3[10] for different frequencies, with medium level of load at 240 V and verify it against the ground truth (GT) as shown in Table 5.5.

In this chapter, we establish a non-parametric, magnetic field intensity-based channel model supporting the unified handling of multiple communication mediums through a common interface using *UCM*. The elemental criterion of the proposed model is that it uses the magnetic field which has been largely ignored as an unwanted element. The signal quality, strength and coverage analysis and parametric optimizations can be effectively addressed by non-invasively measuring the magnetic field intensity from a distance, thus making the model practically feasible.

5.4 Conclusion and Future Directions

This chapter culminates in the characterization of the architecture of Green Symbiotic Cloud Communications (*GSCC*) as envisioned in the development of technologies/systems of the future based on fundamental design postulates. However, this new vision of development of communication and computing technologies also demands for new materials that support the pervasiveness and ubiquity of the proposed paradigm. The next chapter paves a new dimension in electronics with the invention of a new material *tuPOY*, which changes our perception of developing electronics. Evolving on a relatively underplayed phenomenon of static electricity in scientific exploration and application, *tuPOY* upholds the potential to rival both silicon and metals as electronics of the future. The manufacturing process and the conduction and radiation properties of *tuPOY* are covered in a previously published work by the author [18] and do not fall in the domain of this book. The subsequent chapter concentrates on the applications of *tuPOY*, namely the design of an antenna, a power generation unit and finally a transistor. The described design of the transistor is first-of-its-kind, made of a fully non-metallic single material, paving a new dimension in the manufacture of electronics.

A cloud is often defined as a visible collection of particles of ice and water suspended in the air, usually at an elevation above the surface. It is generally a dim and obscure area in something otherwise clear and transparent. The cloud computing paradigm accepted within the scientific community, however, is far from this geographical definition. This thesis purposes an approach to do justice to the classical definition and form a rational basis for advancement towards the same with the *GSCC* paradigm. A new concept of cloud communications is introduced and concretized which democratizes the way we look at communications.

A new approach towards greener and efficient use of communication resources via *GSCC* has been evolved. An exquisite natural resource, the wireless spectrum, is cognitively utilized providing the end users with an enriched communication experience. The linear capacity increase with minimal energy requirement as shown theoretically and corroborated experimentally unfolds endless advantages for *GSCC*. The architecture is outlined both theoretically and experimentally in a static and a dynamic scenario. Provisioning users to utilize multiple *CM*s concomitantly demands the simultaneous handling of different communication standards. We envision a universal set of defining protocols for *CM*s that will contribute towards ease of programming

and mitigate the excess computational load on the *UE* arising from overlap and redundancy. Communication systems handle volumes of data generated by embedded devices, mobile users, enterprises, contextual information, network protocols, location information and such. It is a vast amount of information. For example, a global IP backbone generates over 20 billion records per day, amounting to over 1 TB per day! Processing and analysing this 'big data' and presenting insights in a timely fashion will become a reality with advanced analytics to understand the environment, to interpret events and to act on them. This work is a positive development that helps unleash the intelligence in communications systems where networks are no longer labelled 'dumb pipes' but highly strategic and smart cognitive networks.

We take a novel step towards laying fundamental guidelines for systems of the future. The paradigm or more appropriately the thought of *GSCC*, aiming to replicate the geographical cloud, is presented. A case scenario of *GSCC* propounded on the concepts of the proposed paradigm is evolved. The architecture is fundamentally laid out with nine novel design postulates as its backbone. The theoretical hypothesis verified by experimental testing demonstrates substantial benefits. Not limited to the arena of communications, these design postulates can be extended in developing the technology of the future in various fields, giving rise to multiple dimensions of thought.

References

1. S. Galli, A. Kurobe, and M. Ohura. The inter-phy protocol (ipp): A simple coexistence protocol for shared media. In *Power Line Communications and Its Applications, 2009. ISPLC 2009. IEEE International Symposium on*, pages 194–200, 2009.
2. R. Annavajjala. On optimum regenerative relaying with imperfect channel knowledge. *Signal Processing, IEEE Transactions on*, 58(3):1928–1934, 2010.
3. J.I. Montojo and L.B. Milstein. Channel estimation for non-ideal ofdm systems. *Communications, IEEE Transactions on*, 58(1):146–156, 2010.
4. M. Tucci, M. Raugi, A. Musolino, and S. Barmada. Blind channel estimation for power-line communications by a kohonen neural network. In *Power Line Communications and Its Applications, 2007. ISPLC '07. IEEE International Symposium on*, pages 35–40, 2007.
5. Xin Ding and J. Meng. Channel estimation and simulation of an indoor power-line network via a recursive time-domain solution. *Power Delivery, IEEE Transactions on*, 24(1):144–152, 2009.
6. T. Santos, J. Karedal, P. Almers, F. Tufvesson, and A.F. Molisch. Modeling the ultra-wideband outdoor channel: Measurements and parameter extraction method. *Wireless Communications, IEEE Transactions on*, 9(1):282–290, 2010.
7. G.A. Franklin. Using modal analysis to estimate received signal levels for a power-line carrier channel on a 500-kv transmission line. *Power Delivery, IEEE Transactions on*, 24(4): 2446–2454, 2009.
8. G. Ndo, P. Siohan, and M. Hamon. Adaptive noise mitigation in impulsive environment: Application to power-line communications. *Power Delivery, IEEE Transactions on*, 25(2):647–656, 2010.
9. Yong-Hwa Kim. Multipath parameter estimation for plc channels using the geese algorithm. *Power Delivery, IEEE Transactions on*, 25(4):2339–2345, 2010.

10. M. Tlich, A. Zeddam, F. Moulin, and F. Gauthier. Indoor power-line communications channel characterization up to 100 mhz x2014;part ii: Time-frequency analysis. *Power Delivery, IEEE Transactions on*, 23(3):1402–1409, 2008.
11. J. Anatory, N. Theethayi, R. Thottappillil, M.M. Kissaka, and N.H. Mvungi. Expressions for current/voltage distribution in broadband power-line communication networks involving branches. *Power Delivery, IEEE Transactions on*, 23(1):188–195, 2008.
12. D. Bueche, P. Corlay, M.G. Gazalet, F. X Coudoux, and M. Slachciak. Pilot symbol assisted modulation for powerline communications. In *Industrial Electronics, 2004 IEEE International Symposium on*, volume 1, pages 717–720 vol. 1, 2004.
13. Teong Chee Chuah. Adaptive robust turbo equalization for power-line communications. *Power Delivery, IEEE Transactions on*, 22(4):2172–2179, 2007.
14. C. Lele, P. Siohan, R. Legouable, and J.-P. Javaudin. Preamble-based channel estimation techniques for ofdm/oqam over the powerline. In *Power Line Communications and Its Applications, 2007. ISPLC '07. IEEE International Symposium on*, pages 59–64, 2007.
15. A.M. Tonello. Gen01-1: A wide band modem based on impulse modulation and frequency domain signal processing for powerline communication. In *Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE*, pages 1–6, 2006.
16. G. Avril, F. Gauthier, F. Moulin, A. Zeddam, and F. Nouvel. Time variation of the powerline channel simultaneously with impulsive noise. In *Electromagnetic Compatibility, 2007. EMC Zurich 2007. 18th International Zurich Symposium on*, pages 433–436, 2007.
17. Young Sun Lim, Jae Sung Park, and Jin Young Kim. Novel frame synchronization of ofdm scheme for high-speed power line communication systems. In *Power Line Communications and Its Applications, 2008. ISPLC 2008. IEEE International Symposium on*, pages 182–186, 2008.
18. S.H. Karamchandani, H.D. Mustafa, S.N. Merchant, and U.B. Desai. tupoy: Epitomizing a new epoch in communications with polymer textiles. *Proceedings of the IEEE*, 100(11):3079–3098, 2012.