2014 IEEE 13th International Symposium on Network Computing and Applications

Orchestration of Energy Efficiency Functionalities for a Sustainable Network Management

Ana C. Riekstin*, Guilherme C. Januário*, Bruno B. Rodrigues*, Viviane T. Nascimento*,
M. Roxana Pirlea*, Tereza C. M. B. Carvalho * and Catalin Meirosu†
*University of São Paulo, São Paulo, Brazil. Email: {carolina.riekstin, gcjanuario, roxana.pirlea, tereza.carvalho}@usp.br,
{brodrigues, vianetn}@larc.usp.br
†Ericsson Research, Stockholm, Sweden. Email: catalin.meirosu@ericsson.com

***Abstract*—The energy demand for operating Information and Communication Technologies is growing, implying in high operational costs and consequent carbon emissions. Both in datacenters and in telecom infrastructures, the networks represent a significant amount of energy spending. This implies in an increased demand for energy efficiency, and several functionalities to save energy are being proposed. However, there is no proposal on how to coordinate or combine such energy efficiency functionalities in the same network, or choosing the best one for a given scenario. There is neither a way to do this using business directives, telling, for instance, the period of the day to apply such functionalities or the operational conditions (e.g. if the network usage is low). In this regard, we propose a method able to orchestrate different energy efficiency functionalities considering the possible combinations and conflicts among them, as well as the best option for a given workload and network characteristics, such as topology and power profiles of devices. The business policies are refined down to the network level in order to bring high-level directives into the operation. Our method operation is validated with an example which achieves near 50% of savings for the given scenario by combining two different energy efficiency functionalities.**

Keywords*—*Sustainability, Energy Efficiency, Policy Refinement, Network Management

I. INTRODUCTION

The energy demand for operating Information and Communication Technologies (ICT) is growing. The ICT total electricity consumption is forecasted to increase almost 60% until 2020 to almost 1,100 TWh [1]. Besides incurring in high operational costs, this significant energy consumption leads to greenhouse gases (GHG) emissions. According to GeSI (The Global e-Sustainability Initiative) [2], ICT is responsible in average for 2% of the carbon emissions worldwide. This amount is expected to grow to 2.3% in 2020 [2].The datacenter energy demand is the fastest growing part (including servers, networking, cooling). From 2012 to 2013, its power demand grew 7%, reaching 40GW (near 350TWh). It is expected to grow 81% by 2020 [3]. Although networking accounts for 10 to 20% of energy consumption in datacenters, current efforts on servers and cooling, responsible for roughly 70% of datacenters' consumption, will make the networking share much higher, having the potential to raise up to 50% [4].

For operators, the network infrastructure scenario is even more challenging. According to an Ericsson report [1], energy costs are among the most significant that network operators have to absorb. Verizon, for instance, reports that the electricity to run its networks surpassed 92% of their carbon emissions in 2013 [5]. According to [6], the consumption of telecom operators networks (only the networks) was 260 TWh in 2012. Moreover, their energy consumption growth rate is higher than the world's growth rate. Bolla et al. [7] assert that relying on novel low-consuming silicon technology is not enough to cope with such a demand, and sheds light upon novel management paradigms. They suggest adapting the network requirements and management to take advantage of the periods when the traffic load is not as high as in rush hours, a common situation, making the network more energy efficient.

Several functionalities and protocols have been proposed to cope with energy efficiency in networks, such as Adaptive Link Rate (ALR) [8], Synchronized Coalescing [9], energy-aware routing (a green OSPF) [10], green traffic engineering (GreenTE) [11], ElasticTree [12], and the Sustainability-Oriented Network Management System (SustNMS) [13]. However, to the best of the authors' knowledge, there is no proposal on how to coordinate or combine them in the same network, or choose the best functionality for a given scenario. There is neither a way to do this using business policies.

Policy-Based Network Management (PBNM) uses policies to manage systems, providing a centralized solution to reduce the complexity of the management task. Policies can be expressed in different abstraction levels, and the translation between them is called Policy Refinement. The use of PBNM with automated refinement makes the network management more efficient and less error prone. To manage energy efficiency functionalities or provide green services for users, one can use sustainability-oriented policies. Through these policies, it is possible to offer green Service-Level Agreements (SLAs) to users interested in saving energy and reducing emissions, and to manage the network in an energy efficient manner using operational policies. However, how to refine and use business, high-level sustainability-oriented policies is not obvious, neither in legacy networks nor in more modern software defined networks (SDN). Several requirements must be addressed in this regard. The problem of writing high-level policies and translating to command execution policies is well known [14] and has been studied in the policy refinement community for a while, but no work has become a standard.

This paper presents a method that overcomes the challenges in orchestrating energy efficiency functionalities considering sustainability-oriented policies refinement, thus enabling more energy efficient and automated infrastructure. We analyzed the existing solutions for energy efficiency and sustainability-oriented policy refinement. Our main contributions are the

978-1-4799-5393-6/14 $31.00 © 2014 IEEE
DOI 10.1109/NCA.2014.30

method to orchestrate energy efficiency functionalities and to refine sustainability-oriented policies from business level down to network level. After describing the energy efficiency related works in Section 2, and sustainability-oriented policies refinement aspects and requirements in Section 3, we propose our method in Section 4, showing how to make the network operation more energy efficient. We discuss a preliminary validation of our solution in Section 5. Final considerations are given on Section 6.

II. ENERGY EFFICIENCY TECHNIQUES AND MECHANISMS

There are many different solutions to manage a network focusing on energy efficiency, ranging from local chip-level enablers for more power-efficient nodes to routing protocols. Most of the current approaches use functionalities and protocols that are, in general, based on standard techniques or mechanisms that are already partially available in computing systems. Techniques and mechanisms can be classified in the following categories [15]–[17]: traffic/resource consolidation, selective connectedness, proportional computing, and virtualization. Schlenk et al. [18] proposed a taxonomy that, rather than just the technique (rate adaption, sleeping, and energy-aware routing), takes the scope of the functionality into account: Subsystem (components, memory), such as ALR [8] and ACPI [19]; System (network elements), such as Synchronized Coalescing [9] and IEEE 802.3az [20]; and Network, such as GreenTE [11], SustNMS [13], and ElasticTree [12].

From the aforementioned energy efficiency techniques and functionalities, one can see that they can be conflicting. For instance, if ElasticTree is being applied at the network level and Synchronized Coalescing is running on the network nodes, one functionality could request a router to sleep while the other tells it to keep running. On the other hand, applying more than one functionality in the same network can bring higher gains. For instance, at a certain workload situation, ALR can be applied after SustNMS. SustNMS performs green traffic engineering and puts unused devices to sleep; ALR is applied after, in the nodes that remain powered on, thus enabling higher energy savings. Therefore, a method is necessary, able to handle such conflicts and, at the same time, maximize savings, considering the network conditions.

III. SUSTAINABILITY-ORIENTED POLICIES REFINEMENT

In order to bring business policies down to the network, PBNM and Policy Refinement are key approaches. Policies can be represented in different levels of abstraction, and then refined to more machine-oriented formats. The employment of policies at different abstraction levels is also a way to cope with the difficulty of network management, as it provides a high-level way to perform network configuration and thus makes management significantly easier and less error-prone. The number of layers may be arbitrary and application specific. Strassner [21] proposes five levels: business, system, network, device, and instance. This is the Policy Continuum, of which each level relates to different aspects of network management.

Policy Refinement, the translation between the different continuum levels, involves more than just translating the policy. Based on previous policy refinement works, we identified seven requirements for a sustainability-oriented refinement method. They are: (a) ways to express sustainability-oriented policies in a standardized way; (b) translation of high-level policies into enforceable policies; (c) validation and verification if the refined policy is in accordance to the high-level policy; (d) determination of the resources that are required for the policy execution; (e) policy conflicts detection and resolution; (f) addressing policies dynamicity that is, policies able to handle dynamic time and scenario changes; and, (g) energy efficiency functionalities management, as explained in the previous section. We analyzed the following existing refinement methods: [14], [22]–[31]. None of them complies with all the requirements for refinement of sustainability-oriented policies. In the next section, we propose an architecture that addresses all the requirements.

IV. PROPOSED ARCHITECTURE

Among the requirements, we focus on those that change more with sustainability-oriented policies: (a) ways to express sustainability-oriented policies in a standardized fashion; (b) translation of high-level policies into enforceable policies; (f) addressing policies dynamicity; and (g) energy efficiency functionalities management. The requirements (c) validation and verification if the refined policy is in accordance to the high-level policy and (e) policy conflicts detection and resolution are supported by the table lookup approach we are using, but we let their adaption to this domain as a future work. Figure 1 depicts the proposed architecture, which considers an SDN environment, taking advantage on the centralized approach, also supporting requirement (d) topology discovery.

Considering the existing proposals, the most automated solution that starts the refinement at the business level is based on Table Lookup [32], as the one from [29]. The proposal addresses the translation requirement and supports the validation and verification of policies, the resources determination, and policy conflicts detection. The use of Table Lookup technique is not new, but the way sustainability is going to be represented is novel. To model the policies at each level of the Policy Continuum, we used the Policy Core Information Model Extensions (PCIMe) RFC [33] as basis, and extended it to comprise sustainability-oriented aspects. The Sustainability Information Models are specified using UML and comprise what is needed for a policy to express sustainability issues in each Policy Continuum level. The business-level policies are translated to system level policies by module 1 using Table Lookup, satisfying requirement (b), modeled as defined by the information models, fulfilling requirement (a). The same module identifies the parameters required at the network level. The period of the day (time condition), or the network usage (environment condition) are examples of identified parameters.

The module for management of energy efficient functionalities (module 2 in Figure 1) assigns the best functionality or combination of energy efficiency functionalities to be applied to the network to save energy, given a workload. Table I shows the additional aspects the information models should provide in order for this module to identify the conflicts and possible combinations considering four energy efficiency functionalities as example: ALR, Synchronized Coalescing, ElasticTree (topology-aware heuristics), and SustNMS. Using this table, and the following policies, the module determines the allowed combinations: (i) two network level functionalities

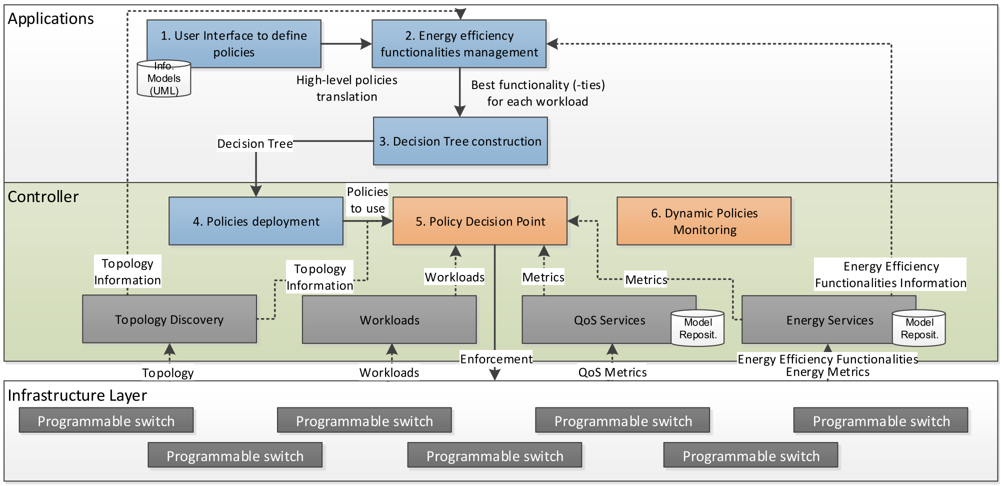


Fig. 1. Architecture for Energy Efficiency Functionalities Management and Business Level Sustainability-Oriented Policy Refinement

performing similar actions cannot be used at the same time; (ii) two or more functionalities to put devices to sleep cannot be applied at the same device, at the same time; (iii) two functionalities to do link rate cannot be applied at the same port, at the same time.

To satisfy the requirement (g), this module performs the following steps: (1) it checks the available functionalities in the network; (2) it uses Table I to select and combine the functionalities in a way that avoids conflicts; (3) considering a set of randomly generated workloads (according to the topology), it combines them with all possible functionalities; (4) given the set of possible combinations of active functionalities that support each workload, it selects the best configuration according to a utility function. We propose a utility function that combines energy savings and QoS, assuring a minimum level of service while trying to save energy, as described in Equation 1, for an n-router topology. For another example on using utility functions, we refer the reader to [34].

$$UF = pl * \frac{1}{\frac{\sum_{k=0}^{n} EnergyAfterSavings_{RouterK}}{\sum_{k=0}^{n} EnergyBaseline_{RouterK}}} \qquad (1)$$

The term pl refers to packet loss, clusterized in categories. For instance, if $0\% < PacketLoss < 0.1\%$, $pl = 1$; if $0.1\% < PacketLoss < 1\%$, $pl = 0.9$; until $pl = 0$ when the losses are bigger than 50%. To calculate the expected savings and losses used in the utility function, we developed analytical solvers for the energy efficiency functionalities. Giving the training workloads, the method can assert what each functionality would do on the network. After determining which devices are going to be affected by the energy efficiency functionalities, an associated algorithm to calculate the packet loss (if any) in the network is required. This is important because a minimum level of service must be specified, because, otherwise, in order to maximize energy savings, we could turn everything off. This analytical solution is proposed because simulating would be expensive in this step.

After combining all generated workloads and the functionalities, the utility function will take the analytical solvers results for expected savings and losses and will determine the best option for each workload. After determining the best energy efficiency functionality for each workload, we use the Scikit tool [35] to build a decision tree able to, given a workload, determine which path in the tree to follow (module 3). Each path in the tree corresponds to a workload whereas the leaves encode the best combination of energy efficiency functionalities for the corresponding workload. As the network workload changes, the system selects another path in the tree to define which functionalities to apply in the new scenario. After the tree is created, it is deployed inside the controller. From this point on, policies are ready to be used in the switches, with the controller acting as a decision point using its services to gather information about QoS and energy (module 5).

The requirement (f) for policy dynamicity is satisfied by two mechanisms: by the controller, that differentiates time, as a special type of a simple policy condition, and by module 6 in Figure 1 that dynamically detects changes in the network and asks for changes in the policies, if necessary. This module and the policy decision point (module 5) are the only modules

TABLE I. ENERGY EFFICIENCY FUNCTIONALITIES ADDITIONAL INFORMATION

	Functionality		Scope			Topology		
	Rating	Sleeping	Port	Device	Network	SoHo/LAN	WAN	Datacenter fat tree
ALR	√		√			√	√	√
Synchronized Coalescing		√		√		√	√	√
SustNMS		√			√	√	√	
ElasticTree		√			√			√

that operate online. That is, in case the system needs to deploy the policies for a new topology or new functionalities, it is not required to cease network operation while the new decision tree is being generated. Regarding scalability, we are studying the use of heuristics to compute each node of the decision tree, in order to minimize the impacts of large amounts of network nodes or energy efficiency functionalities.

V. PRELIMINARY VALIDATION

To illustrate our proposal, we present a use case in Figure 2. The numbers 1-6 represent the architecture module being used. An operational policy is translated, giving the network level information such as the environment conditions (will only save energy if the network load is less than 50%), the period of the day the energy efficiency functionalities will take place (during the night), and the specific network or subnetwork in which they are going to be applied. We are assuming a fat tree topology with 1Gbps edge and aggregation nodes, and 10Gbps core nodes. The power profiles of the nodes must consider the fixed and variable power spent for the handled workload as the one described by [36]. We generate a set of random workloads to train our method. These workloads are combined with the set of possible energy efficiency functionalities.

In the example, the possible functionalities considering the fat tree topology are ALR, Synchronized Coalescing, and ElasticTree. Let us suppose all these functionalities are available in the studied network. The possible combinations, considering the aforementioned policies are: only ALR in all nodes; only Synchronized Coalescing in all nodes; combination of ALR and Synchronized Coalescing in all nodes; ElasticTree alone; ElasticTree followed by ALR. For instance, ElasticTree and Synchronized Coalescing are not expected to work together because both put nodes to sleep. This could lead to conflicts during operation if one tries to put a node to sleep while the other is expecting this same node to be fully operational.

Considering a workload of 10%, a common scenario during low-usage periods, ALR would reduce interfaces speed from 1Gbps to 100Mbps. According to [36], ALR can save up to 21% in the studied equipment. It is also interesting to use after other functionality, in those nodes that remain powered on. Considering the expected savings, in our example, ALR would save approximately 20% of the original power demanded. Synchronized Coalescing, considering the experiments in [9] for the percentage of time the switch stays powered on as a function of load, the ON time for a load of 10% is circa 20%.

Executing ElasticTree in a 20% occupancy network scenario, the authors on [12] reported 38% of energy savings (minimum spanning tree topology to ensure connectivity). This value would be similar for a 10% workload since the minimum spanning tree topology should be respected. The savings in this case are expected to be bigger than with Synchronized Coalescing because the latter does not comprise a traffic engineering functionality while ElasticTree does. ElasticTree will relocate the traffic to allow more switches to sleep, thus saving more energy. Besides, by coordinating ElasticTree energy saving capabilities with ALR, we can potentially increase the savings ratio (reducing the link rate to 100Mbps). Therefore, the best option for this scenario is ElasticTree plus ALR. Dividing the possible savings with ALR for the whole network by the total number of nodes and then multiplying the result with the number of nodes that remained powered on after ElasticTree, the savings can reach 50% for a 10% load scenario and the given topology.

VI. FINAL CONSIDERATIONS AND FUTURE WORKS

Energy efficiency functionalities help operators and service providers to reduce operational costs and GHG emissions in their ICT infrastructures. Sustainability-oriented network management policies can help by bringing business directives into the network, turning the management more automated and less error prone, also reducing operational costs. In this regard, we proposed a method able to orchestrate energy efficiency functionalities considering sustainability-oriented policies refinement, enabling a more energy efficient and automated infrastructure. The architecture presented addresses the seven requirements for the complete refinement of sustainability-oriented policies presented in Section 3, being able to manage different and conflicting energy efficiency functionalities. To the best of our knowledge, this is the first work that comprises the complete refinement of such policies including the orchestration of energy efficiency functionalities. At this moment, we are concluding the implementation to fully automate the calculations, as well as the validation of additional scenarios. The next topology we are going to test was inspired by the core part in the RNP network (*Rede Nacional de Ensino e Pesquisa*, the Brazilian national research and education network).

REFERENCES

[1] Ericsson, “Ericsson Energy and Carbon Report - On the Impact of the Networked Society,” Ericsson, Tech. Rep., July 2013. [Online]. Available: http://www.ericsson.com/res/docs/2013/ericsson-energy-and-carbon-report.pdf

[2] GeSI, “SMARTer 2020: The Role of ICT in Driving a Sustainable Future,” The Climate Group on behalf of the Global eSustainability Initiative (GeSI), Tech. Rep., 2012. [Online]. Available: http://gesi.org/SMARTer2020

[3] G. Cook, T. Dowdall, D. Pomerantz, and Y. Wang, “Clicking Clean: How Companies are Creating the Green Internet,” Greenpeace, Tech. Rep., 2014. [Online]. Available: http://www.greenpeace.org/usa/Global/usa/planet3/PDFs/clickingclean.pdf

[4] D. Abts, M. R. Marty, P. M. Wells, P. Klausler, and H. Liu, “Energy proportional datacenter networks,” in *Proceedings of the 37th Annual International Symposium on Computer Architecture*, ser. ISCA ’10. New York, NY, USA: ACM, 2010, pp. 338–347.

[5] Verizon, “2013 Corporate Responsibility Supplement,” Verizon, Tech. Rep., 2013. [Online]. Available: http://responsibility.verizon.com/assets/docs/cr-report-supplement-2013.pdf

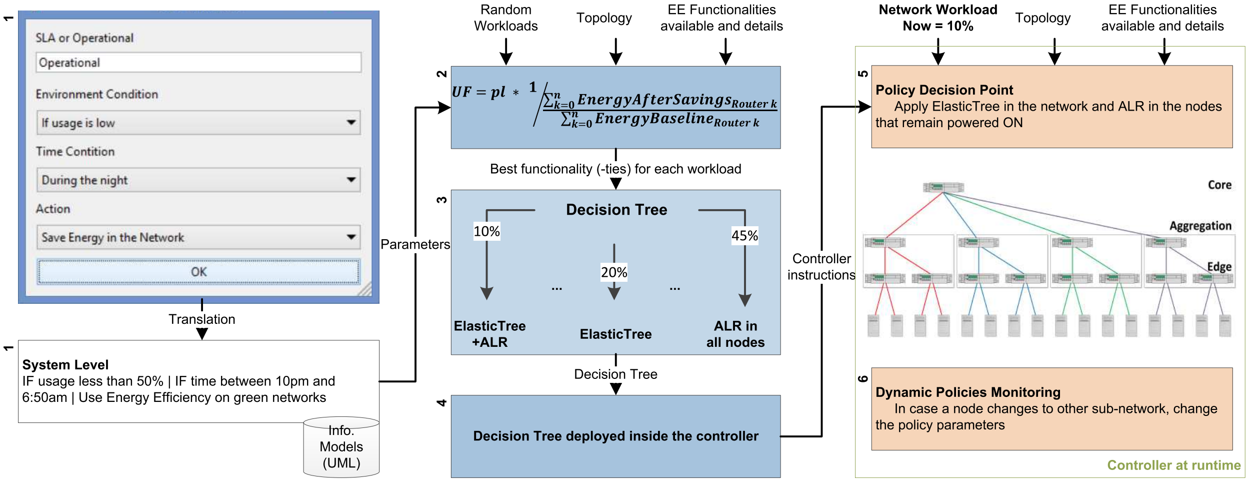
[6] S. Lambert, W. V. Heddeghem, W. Vereecken, B. Lannoo, D. Colle, and M. Pickavet, “Worldwide electricity consumption of communication networks,” *Opt. Express*, vol. 20, no. 26, pp. B513–B524, Dec 2012.

[7] R. Bolla, F. Davoli, R. Bruschi, K. Christensen, F. Cucchietti, and S. Singh, “The potential impact of green technologies in next-generation wireline networks: Is there room for energy saving optimization?” *IEEE Communications Magazine*, vol. 49, no. 8, pp. 80–86, Aug 2011.

[8] C. Gunaratne, K. Christensen, B. Nordman, and S. Suen, “Reducing the Energy Consumption of Ethernet with Adaptive Link Rate (ALR),” *Computers, IEEE Transactions on*, vol. 57, no. 4, pp. 448–461, 2008.

[9] M. Mostowfi and K. Christensen, “Saving energy in LAN switches: New methods of packet coalescing for Energy Efficient Ethernet,” in *IGCC’11*, 2011, pp. 1–8.

[10] A. Cianfrani, V. Eramo, M. Listanti, M. Marazza, and E. Vittorini, “An Energy Saving Routing Algorithm for a Green OSPF Protocol,” in *IEEE Conference on Computer Communications Workshops*, 2010, pp. 1–5.



SLA or Operational: Operational

Environment Condition: If usage is low

Time Contition: During the night

Action: Save Energy in the Network

OK

System Level
IF usage less than 50% | IF time between 10pm and 6:50am | Use Energy Efficiency on green networks

$$UF = pl * \frac{1}{\frac{\sum_{k=0}^{n} EnergyAfterSavings_{Router\,k}}{\sum_{k=0}^{n} EnergyBaseline_{Router\,k}}}$$

Decision Tree deployed inside the controller

Policy Decision Point
Apply ElasticTree in the network and ALR in the nodes that remain powered ON

Dynamic Policies Monitoring
In case a node changes to other sub-network, change the policy parameters

Fig. 2. Use case

[11] M. Zhang, C. Yi, B. Liu, and B. Zhang, “GreenTE: Power-aware traffic engineering,” in *Network Protocols (ICNP), 2010 18th IEEE International Conference on*. IEEE, 2010, pp. 21–30.

[12] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown, “Elastictree: Saving energy in data center networks,” in *NSDI’10*, 2010, pp. 17–17.

[13] C. H. Costa, M. C. Amaral, G. C. Januario, T. C. Carvalho, and C. Meirosu, “SustNMS: Towards service oriented policy-based network management for energy-efficiency,” in *SustainIT’12*, 2012, pp. 1–5.

[14] T. C. M. B. Carvalho, A. C. Riekstin, M. Amaral, C. H. A. Costa, G. C. Januario, C. K. Dominicini, and C. Meirosu, “Towards Sustainable Networks - Energy Efficiency Policy from Business to Device Instance Levels,” in *ICEIS*. SciTePress, 2012, pp. 238–243.

[15] K. Bilal, S. Khan, and A. Zomaya, “Green data center networks: Challenges and opportunities,” in *Frontiers of Information Technology (FIT), 2013 11th International Conference on*, Dec 2013, pp. 229–234.

[16] A. Bianzino, C. Chaudet, D. Rossi, and J. Rougier, “A survey of green networking research,” *Communications Surveys Tutorials, IEEE*, vol. 14, no. 1, pp. 3–20, First 2012.

[17] S. K. Garg and R. Buyya, “Green cloud computing and environmental sustainability,” *Harnessing Green IT: Principles and Practices*, pp. 315–340, 2012.

[18] R. Schlenk, C. Lange, H. Lehmann, and R. Vleugel, “Taxonomy of dynamic power saving techniques in fixed broadband networks,” in *Photonic Networks, 14. 2013 ITG Symposium.*, May 2013, pp. 1–8.

[19] R. Bolla, R. Bruschi, and A. Ranieri, “Performance and power consumption modeling for green cots software router,” in *Proceedings of the First International Conference on COMmunication Systems And NETworks*, ser. COMSNETS’09, 2009, pp. 420–427.

[20] K. Christensen, P. Reviriego, B. Nordman, M. Bennett, M. Mostowfi, and J. Maestro, “IEEE 802.3az: the road to energy efficient ethernet,” *Communications Magazine, IEEE*, vol. 48, no. 11, pp. 50–56, Nov 2010.

[21] J. Strassner, *Policy-Based Network Management: Solutions for the Next Generation*, 1st ed. Morgan Kaufmann, Sep. 2003.

[22] M. S. Beigi, S. Calo, and D. Verma, “Policy transformation techniques in policy-based systems management,” in *Policies for Distributed Systems and Networks, 2004. POLICY 2004. Proceedings. Fifth IEEE International Workshop on*. IEEE, 2004, pp. 13–22.

[23] Y. B. Udupi, A. Sahai, and S. Singhal, “A classification-based approach to policy refinement,” in *Integrated Network Management, 2007. IM’07. 10th IFIP/IEEE International Symposium on*, 2007, pp. 785–788.

[24] J. Rubio-Loyola, *Towards the Policy Refinement Problem in Policy-based Management Systems: A synthesis study*. VDM Publishing, 2008. [Online]. Available: http://books.google.com/books?id=hO0HNQAACAAJ

[25] K. Twidle, E. Lupu, N. Dulay, and M. Sloman, “Ponder2-a policy environment for autonomous pervasive systems,” in *Policies for Distributed Systems and Networks. IEEE Workshop on*. IEEE, 2008, pp. 245–246.

[26] A. Uszok, J. M. Bradshaw, J. Lott, M. Breedy, L. Bunch, P. Feltovich, M. Johnson, and H. Jung, “New developments in ontology-based policy management: Increasing the practicality and comprehensiveness of kaos,” in *Policies for Distributed Systems and Networks, 2008. POLICY 2008. IEEE Workshop on*. IEEE, 2008, pp. 145–152.

[27] R. Craven, J. Lobo, E. Lupu, A. Russo, and M. Sloman, “Policy refinement: decomposition and operationalization for dynamic domains,” in *Network and Service Management (CNSM), 2011 7th International Conference on*. IEEE, 2011, pp. 1–9.

[28] B.-S. Liao and J. Gao, “An automatic policy refinement mechanism for policy-driven grid service systems,” in *Grid and Cooperative Computing-GCC 2005*. Springer, 2005, pp. 166–171.

[29] D. C. Verma, *Policy-based networking: architecture and algorithms*. New Riders Publishing, 2000.

[30] A. Voellmy, H. Kim, and N. Feamster, “Procera: a language for high-level reactive network control,” in *Proceedings of the first workshop on Hot topics in software defined networks*, ser. HotSDN ’12. New York, NY, USA: ACM, 2012, pp. 43–48.

[31] C. Monsanto, N. Foster, R. Harrison, and D. Walker, “A Compiler and Run-time System for Network Programming Languages,” *SIGPLAN Not.*, vol. 47, no. 1, pp. 217–230, Jan. 2012.

[32] A. K. Bandara, E. C. Lupu, and A. Russo, “Using event calculus to formalise policy specification and analysis,” in *Policies for Distributed Systems and Networks, 2003. Proceedings. POLICY 2003. IEEE 4th International Workshop on*. IEEE, 2003, pp. 26–39.

[33] Moore, B., “RFC3640 - Policy Core Information Model (PCIM) Extensions,” United States, Tech. Rep., 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3460.txt

[34] G. Jung, K. R. Joshi, M. A. Hiltunen, R. D. Schlichting, and C. Pu, “Generating Adaptation Policies for Multi-tier Applications in Consolidated Server Environments,” in *Proceedings of the 2008 International Conference on Autonomic Computing*, ser. ICAC ’08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 23–32.

[35] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[36] M. Ricca, A. Francini, S. Fortune, and T. Klein, “An assessment of power-load proportionality in network systems,” in *Sustainable Internet and ICT for Sustainability (SustainIT), 2013*, Oct 2013, pp. 1–8.