2012 IEEE 17th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)

A Proof-of-Concept Testbed for Cooperative and Self-Growing, Energy Aware Networks

Panagis Magdalinos, Gerasimos Stamatelatos, Nancy Alonistioti

Department of Informatics and Telecommunications, National and Kapodistrian University of Athens
Athens, Greece

[1]panagis@di.uoa.gr

[2]makiss@di.uoa.gr

[3]nancy@di.uoa.gr

***Abstract*— Cooperating heterogeneous wireless elements provide advanced problem solving capabilities and improved services. At the same time, low energy solutions create attractive business case offering significant benefits in terms of products dependability, and operation costs. CONSERN project addresses those challenges through the Self-growing paradigm, combining autonomic and collaborative capacities towards serving energy efficiency in heterogeneous networking environments. This paper describes the CONSERN Self-growing Functional Architecture, realised by a set of mechanisms and enablers which have been deployed in a real life proof-of-concept testbed.**

***Keywords*—** heterogeneous wireless elements, cooperative networking, autonomic control, SON, Self-growing.

I. Introduction

In future distributed systems, mechanisms achieving dependable behaviour for large-scale networking will realise an important evolution step. Key challenges lie in the efficient cooperation of heterogeneous elements providing advanced problem solving capabilities and services. Furthermore, low energy solutions, combating climate change, create an attractive business case by offering significant benefits in terms of operational cost, long-term product reliability, sustainability, and increased lifetime of network elements.

In order to provide tangible solutions for the presented challenges, the ICT-CONSERN project [7] focuses on two research directions: i) Solutions for optimised energy and power consumption in a small-scale, purpose-driven network through balancing autonomic and cooperative approaches, and ii) Enablers for the self-growing of the network/system, towards a large scale, multi-purpose network/system.

In this paper, we present an excerpt of this research effort and its deployment in a real-life, proof-of-concept, testbed. Due to space constraints we omit the presentation of related work. For an extensive overview of related work in the areas of energy efficiency and self-growing the reader may refer to [5], [3], [8] and [9]. Section II, presents the CONSERN Functional Architecture; Section III presents the incorporation of energy efficiency and self-growing enablers in the architecture. Section IV presents the testbed and Section V concludes the paper and sketches future research directions.

II. The CONSERN Functional Architecture

The CONSERN Functional Architecture (FA) targets the realization of the self-growing concept: heterogeneous networks' capabilities (e.g. WSN) are dynamically exploited and combined, optimizing network's autonomic operations whilst the principles of research efforts in the area of autonomic network management [1], [2] are adopted and incorporated as well. Self-growing Cognitive Manager (SCM) is the main function and is a software entity embedded in IP nodes realising self-x or SON actions (i.e. optimization, configuration, and growing). CONSERN-enabled network integrates SCMs communicating to each-other, to other devices featuring networking capabilities (e.g. sensors) or even legacy devices. Such network is capable of communicating with adjacent CONSERN-enabled networks thus highlighting the full extent of self-growing application.

Each SCM realizes a control loop realising (a) monitoring, (b) decision making, (c) execution of decided actions, and (d) learning from prior decisions and executions. Each control loop can be executed in a distributed fashion, among different SCMs providing a subset of the overall functionality. This way, CONSERN FA envisages the execution of multiple coordinated cognition loops across different SCMs operating even in different time scales; thus enabling network identifying and reacting to events caused by fast and slow changing network dynamics. The overall CONSERN FA described below consists of the fully functional SCM, and the interfaces for the network administrator side.

A. The Self-Growing Cognitive Manager

A fully functional SCM is composed of i) the CONSERN Cognitive Engine (CCE), ii) the Communication Services, and iii) the Translation (TRA) function. In this paper we will focus on the CCE. Further details can be found at [3][4].

The CONSERN Cognitive Engine (CCE) provides the intelligence enabling SCM to realize the self-growing paradigm, the cooperation and energy-efficient mechanisms developed in the context of the project. The CCE is composed by lower layer functions responsible for executing specific parts of the cognition loop. These are as follows (Figure 1):

Information Base stores information related to the device status. Format is common across network devices (e.g. terminal, access point). **Communication Services** implement the communication among all involved network entities. **Monitoring** undertakes the task of monitoring the operational environment of the device. **Decision Making** undertakes the tasks of collaborative decision making, accommodation of novel network elements, reconfiguration and adaptation according to network conditions. **Execution's** main task is the actual execution of high level instructions/directives on the underlying hardware. **Self-Growing** checks the local topology in order to identify available self-growing capabilities. **Learning** enhances the system with learning capabilities in order to build knowledge for improving future decision-making and enable system's proactive operation. **Translation** provides the translation between abstract configurations commands generated by the CCE into vendor/hardware specific configurations.

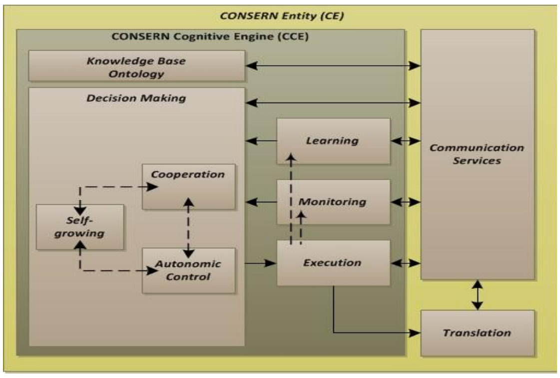

Figure 1: CONSERN Cognitive Engine.

III. CONSERN Mechanisms

In the context of this paragraph, four major mechanisms, comprising research outcomes of CONSERN are presented and their incorporation in the presented architectural framework will be identified.

A. System Idle Time Estimation

System Idle Time Estimation (SITE [5]) addresses important issue in future energy-aware systems i.e. how to accurately estimate the idle time of a system in order to enter low-energy mode and maximize energy gains. SITE is based on periodic monitoring for detecting whether the system is active or not and react accordingly whilst the key assumption is that events occur in batches. SITE's event-driven algorithmic solution developed on stochastic model is aiming at (i) maximizing the successful identification of events and (ii) simultaneously minimizing energy consumption.

We consider a model in which each node can operate in a certain space of states [1, *K*]. In each state of operation, the node scans the environment at a given frequency. In *state-1* the frequency has its minimum value and the node idle period is the longest, while in *state-K* the node scans with the maximum frequency and the node idle period is the shortest. A node can identify the existence of an event based on thresholds that are relevant to the case under consideration. Each state *i* is characterized by (i) a maximum number of failed sense attempts that a node can perform before transiting to a previous state, and (ii) the scanning frequency. If a node detects an event, it transits to a higher state (towards *state-K*, i.e. more frequent scanning). If the node does not detect an event, it transits to a lower state (towards *state-1*, i.e. sparser scanning). Thus, in *state-1* the node consumes least energy, but the penalty for this is high probability of missing an event. On the other hand, in *state-K* the node can detect the occurrence of an event with the highest probability, but the penalty is higher energy consumption. In other words, as the node scanning attempts are getting more frequent, the probability of a successful identification of an event is getting higher but the energy consumption is increased as well. The SITE mechanism is integrated in the Monitoring module of the proof-of-concept environment.

B. Access Points Clustering

Clusters of network nodes are defined as dynamic medium-term (or in some occasions opportunistic) federations of Control Loops (CLs) enabling network devices collaboration for distributed monitoring and decision making. The formation of clusters among network devices (e.g., APs in this work) is promoted for synergistic tackling of various management problems, by extending individual CL monitoring, situation awareness capability and decision making capacity (e.g., APs de-activation, Tx Power adjustment) through CLs collaboration. Clustering is in effect a mechanism for creating communities.

The proposed clusters of CLs are non-overlapping and consist of two types of nodes: a) the Simple Member CL, and b) the Head CL managing the cluster. The Head CL, which is unique per cluster, collects and correlates monitoring data provided by simple cluster member nodes. Based on this information the Head builds a wider view of the status of the cluster area and thus is able to make global deductions which cannot be handled individually by Simple Member CLs.

The scheme for clusters formation is based on the topological characteristics of the network area (i.e. APs degree, diameter of the network area) where clusters are created by a process of "preferential attachment", according to which members prefer to connect to the most "popular" existing members. The application of the proposed scheme leads to the election of the Head CLs and the specification of the borders of the clusters by allocating the member CLs to the elected Heads. Each Head is aware of the member nodes constituting its cluster, their distance in hops, and its degree, while each member is aware about the Head id (e.g., MAC) that is assigned to and their distance. The proposed distributed clustering scheme takes into account the low or zero mobility levels of the APs. Further details about this algorithm can be found in [4]. The AP Clustering algorithm is implemented in the context of the Self Growing module while exploits functionalities of Monitoring, Execution and Decision Making.

C. Access Points Switch On/Off

Following the formation of AP Clusters we introduce an algorithmic model for wireless networks coverage optimization addressed by the dynamic deactivation or reactivation of a group of APs in a network area, according to the existing capacity requirements, a typical coverage and capacity management issues.

The goal of the coverage management is to provide network connectivity at all desired locations, while capacity management undertakes to provide sufficient bandwidth to satisfy clients' needs. Towards this end we introduce two metrics, (i) the Capacity Usage Ratio (CUR) of a network area consisting of *n* APs which indicates the percentage of the used capacity that the APs provide and, (ii) the Overlapping Factor (OF) which takes into account overlappings in the transmission range of the *n* APs.

The association of the CUR and OF in the network allows the more effective interpretation of the information that CUR provides by taking into account the overlapping level of the offered bandwidth in the corresponding network area. For this reason we introduce Coverage Optimization Opportunity Coefficient (COOP), which is given by $COOP = CUR^{OF}$. The COOP metric is useful for the identification of optimization opportunities for a low load (i.e. less capacity is needed) as well for a high load situation (i.e. more capacity is required). Low COOP value means that too much capacity is provided in a very dense area (i.e. switch AP off), while a high COOP value indicates an overloaded network area, so a need for more resources (i.e. switch AP on).

The On/Off algorithm is integrated in the Decision Making function while further information can be found in [4].

D. Cooperative Power Control

The final –and directly related to the coverage and capacity optimization case outlined previously- step is the proper adjustment of APs Tx Power so as to minimize overlaps and save energy in case AP needs to switch on for coverage requirements. The proposed approach is based on information exchange scheme identifying the appropriate Tx power levels. The algorithm incorporates a fuzzy reasoner for handling uncertainties related to message exchange.

The algorithm consists of three steps, (i) the initialization, (ii) the power update, and, (iii) the interference price update. The former is related to the introduction of initial valid transmission power and interference price values. The second concerns the transmission power update based on the interference prices received from neighbor nodes. Finally, the latter captures the communication of its interference prices to the neighborhood, by every network node. Steps (ii) and (iii) are asynchronously repeated until the algorithm reaches a steady state (i.e. a state where every network element has the same transmission power for two consecutive time iterations).

The algorithm is integrated in the Decision Making function while also exploits functionalities of Execution, Monitoring and Communication Services. Further information related to this approach can be retrieved from [6].

IV. Deployment

In this section, the actual implementation of the previously described architecture followed by the realization and real life deployment of a self-growing test case is presented.

A. Testbed Deployment

At first we provide the details of the implementation effort for the deployment of the self-growing architecture. Thus, this paragraph builds on the theoretic outcomes of the previous sections and establishes the validity and viability of the proposed solution. We commence, by providing an overview of the hardware and software platform and proceed with the description of the realization of the architecture on the testbed.

Hardware Infrastructure

The experimentation facilities consist of a variety of equipment including hardware and software components all deployed in the SCAN Lab premises [11]. Figure 2 depicts the available network elements. Several interconnected routers and switches provide access to the Internet. The core network supports different access technologies (both wireless and wireline). A number of IEEE 802.11 (Wi-Fi) access points are also available providing connectivity to a set of Wi-Fi-enabled mobile terminals. Overall, the testbed has the ability to monitor energy consumption in the wireless access points.

The core network consists of 4 Linux based routers and several multiport switches. The Wi-Fi access points are Soekris devices net5501 [10] running Linux (kernel version 2.6.33). These devices are fully programmable, thus enabling effective and efficient prototyping.

The "terminals" side hosts numerous commodity laptops, PDAs and smart–phones all operated by SCAN personnel during the course of a working day. To put it simply, researchers employ the deployed infrastructure in order to access the internet while the four CONSERN algorithms run in the background and manage the device.

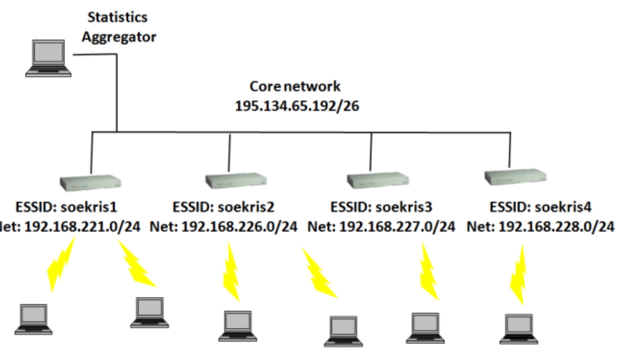

Figure 2: Actual testbed used in the context of the experimental assessment

Software Infrastructure

Web Services have been used for the implementation of the selected architectural concepts and techniques. Web Services developed in Java are used for managing the network devices and monitoring resources e.g., energy, signal strength, radio frequencies. Software bundles are also deployed on the terminals in order to access specific characteristics of the devices (energy consumption, wireless channels, CPU levels, memory usage and battery levels).

The internal software architecture of each device is depicted in Figure 3. The **Monitoring** module gathers information after scanning the operational environment. Scanning is performed according to the System Idle Time Estimation approach, thus time interval between subsequent scans is dynamically adjusted to event occurrence.

The **Channel Selection** module, although not developed in the context of CONSERN, is used in order to enable the AP to select the transmission channel so as to minimize interference. **Cooperative Power Control** runs in the background and dynamically adjusts the device Tx power according to the signal strength of nearby operating access points.

All algorithms are orchestrated by the **Cognitive Control Engine** which evaluates the measurements retrieved by the **Monitoring** module and assesses the occurrence of problematic situations, namely high interference or high load. In the first case, it triggers the **Channel Selection** in order to perform channel re-allocation while in the second one the **AP On/Off** module in order to switch on an existing AP.

In order to facilitate information exchange among devices and avoid information flooding, we introduced an indexing entity in the core network. The latter is a simple Java application which retains the core network IPs of the APs and their corresponding ESSIDs. In this way, if an AP wishes to directly communicate with a neighboring AP it queries the indexing software and uses the core IP of the neighbor.

The access point clustering algorithm is triggered manually. Its task is to initialize the process of clustering so that participating APs elect cluster head to undertake all organizational issues. The cluster head is the only AP which actually exploits the functionality of the AP On/Off algorithm in order to resolve load balancing issues. It has to be stressed out at this point that in case of network resources waste (overall capacity is larger than actual requirements) then APs are directed to switch off.

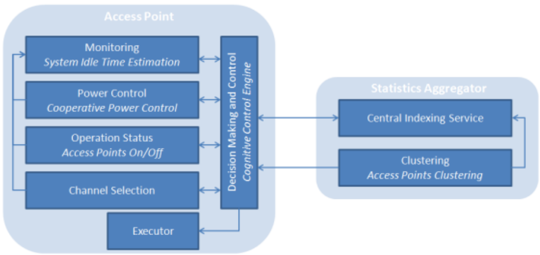

Figure 3: Internal Software Deployment

A typical initialization scenario follows; at first the Monitoring module is initialized which constantly scans the environment. Part of the derived information is forwarded to the indexing through the Cognitive Control Engine. Afterwards, channel selection takes place and the optimal deployment channel is forwarded to the CCE. The latter enforces the decision on the underlying hardware by exploiting the capabilities of the translation engine (not appearing in this figure). As soon as a channel is selected the device initiates the power adjustment procedure. The latter exploits information available in nearby devices and adjusts its transmission power in order to minimize interference and energy consumption.

After these first initialization steps, the clustering procedure is initiated. In an environment with 4 operating APs, one cluster is generated. However the resulting cluster head initialized and periodically executes the AP On/Off algorithm. The latter attempts to optimize network resources consumption by switching on and off APs based on network wide traffic conditions. In parallel, all APs execute the Cooperative Power Control algorithm.

B. Experimental Results

NKUA SCAN Lab members ([11]) used these access points for 12 consecutive hours (from 11:00 CET until 23:00 CET on May 28th 2012) in order to access the internet and perform all normal, working-day, activities. Overall traffic throughout during the day ranged from 1 to 10 Mbps while APs were configured to serve clients at 5.5Mbps.

	AP1	AP2	AP3	AP4
ECP	47%	36%	36%	37%
SDP	91%	96%	98%	97%

Table 1: Results of SITE algorithm on all 4 APs

The SITE algorithm was evaluated with respect to the Energy Consumption Percentage (ECP) and the Success Detection Percentage (SDP) indicators. ECP is calculated as the ratio of monitoring loops performed by SITE to the number of loops performed by a fixed sleep-interval thread. Similarly, SDP is calculated as the ratio of events identified by SITE to the number of events identified by a fixed sleep-interval thread. Table 1 depicts the corresponding results; SITE achieves a reduction of more than 50% in monitoring loops in conjunction with an event detection level higher than 90%.

In order to trigger the On/Off algorithm we deliberately set extremely low values for On and Off triggers, 0.4 and 0.1 respectively. Due to the limited area for experimentation, OF (Overlapping Factor) was always 1, signifying that all devices could scan each other. Thus, an access point would be directed to switch off if the global traffic fell below 10% of the maximum available capacity. Similarly, an AP would be directed to switch on if global traffic exceeded 40% of the maximum available capacity. The results obtained by the experiment appear in Figure 4.

The de-activation of an access point's RF reduces its energy consumption from 9.73 to 7.79 watts thus induces a 20% reduction per access point. Given the fact that 1 watt = 1 Joule / second, the aggregated gain because of the 3 RF de-activated from 18:00 until 23:00 is 104KJoules (4 fully functional APs from 18:00 to 23:00 require 700.56 KJoules, while 1 fully functional AP and 3 with their RFs suspended require 595.80 KJoules). Alternatively, a network operating with CONSERN enabled Access Points requires 15% less energy than the same network operating with non-CONSERN devices.

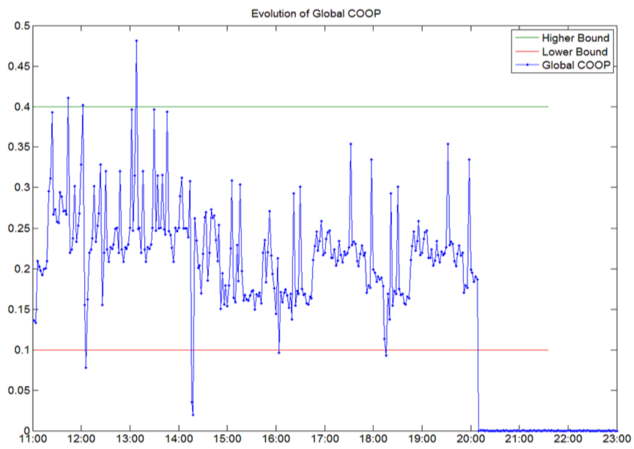

Figure 4: Results of the APs On/Off algorithm

In parallel, all devices perform some kind of adjustment which results in reducing the power requirements of the wireless card. AP4, which is the cluster head, operates most of the time on significantly lower TxPower (Figure 5). Indeed, its TxPower ranges from 65%-85% of the highest attainable value (27dbms) while after 20:00 CET, when everybody had left the office, the TxPower was adjusted to 75% and continued to operate on this level until the end of the experiment. These experiments demonstrate that CPC can reduce the power consumed by an AP from 0.5 to 0.06 Watts (10 times less energy), thus inducing significant energy gains.

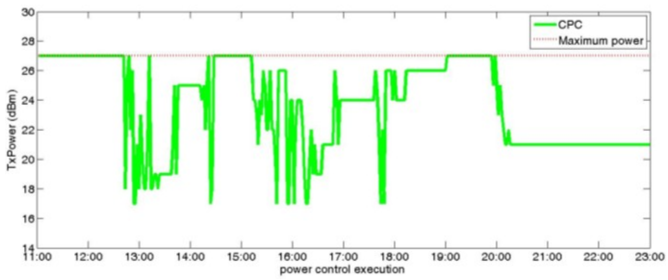

Figure 5: Power Adjustment with CPC on the cluster head

Table 6 provides a synopsis of the derived experimental results. A full detailed presentation of this experimentation activity is available in [12].

Algorithm	Synopsis
System Idle Time Estimation	A reduction of more than 50% in monitoring loops in conjunction with an event detection level higher than 90%
Cooperative Power Control	15% to 35% reduction in the TxPower of participating APs
AP Switch On/Off	15% reduction in the energy required by the network.

Table 6: Synopsis of derived experimental results

V. Conclusions

In this paper we presented the key outcomes of the CONSERN project at Architectural, mechanisms and deployment level. CONSERN Functional Architecture was overviewed as providing the reference framework for embedding networking devices with cooperative energy efficiency solutions under the Self-growing umbrella. Moreover, a set of promising mechanisms was presented – after a pre-evaluation at simulation level-as realizing self-growing and cooperative capacities towards advanced problem solving. Finally, the plan, the layout and the actual deployment of the integrated CONSERN testbed was presented based on the mentioned mechanisms and enablers. Extended experimentation is currently on-going and will be reported in following papers.

Acknowledgment

This work has been performed in the framework of the ICT-257542 CONSERN project funded by the European Union.

References

[1] Jennings, B., Van der Meer, S., Balasubramaniam, et al. "Towards Autonomic Management of Communications Networks", IEEE Communications Magazine 45(10), pp. 112–121, 2007.

[2] R. Chaparadza, S. Papavassiliou, T. Kastrinogiannis, M. Vigoureux, et al. Creating a viable Evolution Path towards Self-Managing Future Internet via a Standardizable Reference Model for Autonomic Network Engineering. FIA Prague 2009 Conference, published in the FI Book produced by FIA, 2009.

[3] CONSERN Deliverable D4.1, "Initial Description of Self-growing Scenarios, Properties, Requirements and Envisaged Framework".

[4] CONSERN Deliverable D4.2, "Distributed Self-growing Architecture and Interface Description".

[5] CONSERN Deliverable D2.1, "System Level Energy Optimization Solutions"

[6] CONSERN Deliverable D3.3, "Energy-Efficient Cooperative Decision and Control Schemes" .

[7] ICT CONSERN, COoperative aNd Self growing Energy awaRe Networks, https://www.ict-consern.eu

[8] CONSERN Deliverable D3.1 "Enablers for Energy - Aware Cooperative Decision and Control"

[9] CONSERN Deliverable D2.2 "Terminal Level Energy Optimisation Solutions"

[10] Soekris communication devices http://www.soekris.com/

[11] Self-Evolving Cognitive and Autonomic Networking Lab, http://scan.di.uoa.gr/, National and Kapodistrian University of Athens

[12] A day in the SCAN Lab facilities http://www.youtube.com/watch?v=efcANFYLzwI