# Taxonomy of Dynamic Power Saving Techniques in Fixed Broadband Networks

Ralph Schlenk[(1)], Christoph Lange[(2)], Heiko Lehmann[(2)], Ron Vleugel[(1)]

[(1)]Alcatel-Lucent Deutschland AG, Thurn-und-Taxis-Straße 10, 90411 Nürnberg
[(2)]Deutsche Telekom AG, Telekom Innovation Laboratories, Winterfeldtstraße 21, 10781 Berlin

{ralph.schlenk; ron.vleugel}@alcatel-lucent.com; {christoph.lange; h-lehmann}@telekom.de

## Abstract

Energy awareness in wired networks has become a major trend, triggered by increasing energy prices and environmental concern. However, power management solutions can result in degraded network performance if power saving mode exit latencies are too high. This paper reviews the relevant technologies and their temporal behavior on subsystem-, system-, and network levels. To this end, approaches are categorized into adaptive link rate, sleeping, and energy-aware data/traffic control techniques. In a second step, corresponding switching time vs. capacity characteristics are developed. Different network element switching strategies are considered, namely serial and parallel switching. The derived models are validated by an in-depth analysis of the activation behavior of a current generation network element.

## 1 Introduction

In light of higher energy costs and the drive towards energy-efficient networks the concept of load-adaptive networking has gained increased popularity in the past few years. Within the project DESI („Durchgängig Energie-Sensible IKT-Produktion“[1]) systems and service providers, a systems integrator, and optimization experts investigate potential and constraints of load-adaptive operation of fixed telecommunication networks and their coupling to the smart energy grid.

In [1] the authors presented a general network element model which connects the power consumption to the capacity provided. Based on this framework a refinement of existing analytical power consumption models [2] is possible. Additionally, simulations for optimization of load-adaptive networks can be simplified.

It should be noted that load-adaptive network operation comprises power management functionality on several architectural levels [3]. While the corresponding technologies on component- and subsystem-level (e.g. microprocessors) are already employed, developments on network element and on network levels are still at an early stage. Essential to the, per se, *dynamic* load-adaptive mode is a temporal behavior that keeps performance degradations within the limits defined by service level agreements. For example, a long wake-up time from sleep mode may result in non-acceptable traffic outages.

In this paper, both a survey of relevant load-adaptive power-saving technologies and an analysis of their dynamic behavior are presented in Section 2. In Section 3 the analytical model of [1] is extended to account for the dynamics of load-adaptive operation. The model is compared with measured characteristics in Section 4. Results are summarized in Section 5.

## 2 Dynamic aspects of green networking

In general there is a trade-off between energy savings and network performance (bandwidth, delay, loss, resilience etc). A more detailed analysis shows that degradation of network performance depends both on the timescale of the power management actions and on the architectural level at which these actions are taken. For example, microprocessor sleep modes in the order of 10–100 nanoseconds, on the one hand, can be handled within the corresponding subsystem and thus have limited effect on the network. Receiver sleep modes, on the other hand, with wake-up times in the order of seconds, require coordination between several systems to mitigate their disadvantageous effects.

In this section we will classify the major load-adaptive power management technologies according to their temporal properties. In doing so, we limit our investigations to fixed broadband networks. Our findings will be applied to the models derived in the following section.

[1] “Pervasively Energy-Efficient ICT Production”

## 2.1 Classification criteria

Various power management solutions for transmission networks have been proposed and employed so far. In order to study their coordinated operation we selected following criteria for a categorization:

- Architectural level [3]:
  - Subsystem (devices, components, memory)
  - System (network elements)
  - Network
- Timescale [4]:
  - Intra-packet (less than 10 µsec)
  - Intra-flow (10 µsec – 10 sec)
  - Inter-flow (10 sec to several min)
- Approach:
  - Rate adaptation
  - Sleeping
  - Energy-aware data/traffic control

While power management techniques on subsystem level are confined to the network element itself due to their shorter timescales (as shown in Section 2.2), actions on system level might require local coordination between adjacent network elements. Network-level technologies, however, call for complex network-wide energy management control functions.

Another important criterion is the temporal behavior of a power management technique, in particular the exit latency which influences network performance. Following the rationale of [4], we chose three different timescales related to the traffic flow.

Finally, we classified the solutions to the employed approach. We distinguish three basic methodologies: rate adaptation, sleeping, and energy-aware data/traffic control. While the two schemes rate adaptation and sleeping are contrary to each other and have been compared in different publications, e.g. [5], we found solutions pertaining to data/traffic control to be more diverse.

## 2.2 Taxonomy of dynamic power management techniques

**Table 1** gives an overview of the main energy-saving strategies for fixed transmission networks. The different contributions are listed in rows, color coded in green/yellow/orange with their respective power mode exit latencies, i.e. reconfiguration or wake-up times. Data in this table has been obtained from publicly available sources and own measurements.

It stands out that the *subsystem level* column is well populated for all categories. This is because energy management solutions on this level clearly profit from work done for personal computers and battery-driven devices in the past years. Exit latencies are mostly below 10 µs, thus data can be buffered on the device and actions are confined to the subsystem or a single system only.

Control becomes more complex for power management on *system level*. Network interface re-configuration (rate change, sleep mode etc.) needs to be aligned with the far end interface through appropriate protocols.

System power reduction efforts currently focus on access network elements as these consume the largest part of the network power. Solutions have been developed for DSL (reduction of transmit power and rate in L2 mode, sleep in L3 mode) [13], PON (dozing with transmitter asleep; cyclic sleeping with both transmitter and receiver asleep – both at the ONT/ONU side) [16], and Ethernet (rapid PHY selection – not standardized; low power idle in Energy Efficient Ethernet) [14][15]. While minimum low-power mode exit times are constrained by hardware capabilities, often additional parameters/policies can be configured to balance temporal performance and energy efficiency, taking into account traffic properties and service classes [13][16].

Further power savings can be achieved by measures on *network level*. In contrast to opportunistic sleeping on subsystem and system levels, those techniques require network wide coordination and management: Energy-aware routing, for example, aims to preserve minimum connectivity and service levels while putting unused network resources to sleep – or powering them down completely. Wake-up times, however, are significantly longer than for the other two categories. This is not only due to topology / protocol convergence times and transmission delay variations (caused by bandwidth changes). In particular, re-activation of powered down network resources that do not (yet) have special power management features contribute to the long transition times. It is the goal of this paper to characterize this behavior in more detail in order to facilitate further optimizations of energy-aware routing algorithms.

## 2.3 Improving activation times

For many classes of network elements, speeding up activation times of interfaces and line cards or startup times of the entire chassis have not been design goals so far.

However, decreasing wake-up / power-on times are now becoming relevant research and engineering topics. Design optimization principles include:

- HW Recovery
  e.g. faster resynchronization by using a local oscillator as input to CDR circuits during sleep
- SW Recovery
  e.g. OS boot time reduction by removing unused functionality
- Parallelization
  e.g. parallel download/activation of components
- Fetch-ahead
  e.g. by first performing only essential configurations
- Local Persistency
  e.g. configuration storage in local non-volatile flash memory for faster activation

Knowledge of these mechanisms is a prerequisite for detailed analyzes of activation times. In the following, we

**Table 1:** Overview of dynamic power management techniques for fixed transmission networks

| Approach | Subsystem | | | System | | Network | | |
|---|---|---|---|---|---|---|---|---|
| Rate Adaptation | Dynamic voltage and frequency scaling | CPU P-states [6] | 10-100 µs | DSL L2 mode [13] | <1 s[e] | | | |
| | | Memory DVFS [7] | <100 µs[a] | Ethernet RPS [14] | 20 ms | | | |
| Sleeping | CPU C-states (C1, C2) [6] | | 10-100 ns | DSL L3 mode [13] | <3 s[e] | | | |
| | CPU C-states (C3) [6] | | 50 µs | PON dozing [16] | 1 µs[f] | | | |
| | DRAM power-down [8] | | <10 ns | PON fast sleeping [16] | <200 µs[f] | | | |
| | DRAM self refresh [8] | | <1 µs | PON deep sleeping [16] | –[g] | | | |
| | CMOS clock gating [9] | | 10 ns[b] | EEE low-power idle [15] | 1-25 ms[h] | | | |
| | CMOS clock and power gating [9] | | 5 µs | | | | | |
| | FPGA suspend techniques [10] | | <100 µs | | | | | |
| Energy-aware data/traffic control | CPU idle core "hotplug" [11] | | 5 ms[c] | Ethernet background traffic buffering [17] | 5-20 µs[i] | Energy-aware routing | Interface shutdown | >10 s[j] |
| | Power-aware virtual memory [12] | | <1 µs[d] | | | | Line card shutdown | > 30 s[j] |
| | | | | | | | Chassis shutdown | minutes[j] |
| | | | | | | | Protocol conver-gence | n/a[k] |

[a] same as CPU DVFS
[b] applicable on a cycle-by-cycle basis
[c] after optimization
[d] linked to memory wake-up times
[e] according to standard
[f] assuming burst-mode transceivers
[g] not applicable – configurable timer-based
[h] fixed timer interval, PHY dependent
[i] Ethernet rx wake-up time; PHY dependent
[j] assuming no special sleep-mode optimizations; network element dependent
[k] not evaluated in this paper

will concentrate on modeling the most important ones, namely recovery and parallelization techniques, to derive our characteristics.

# 3 Network reconfiguration time characteristics

In [1] the characteristics of network elements (NE) with respect to their power consumption vs. activated capacity have been modeled and studied. Besides this pure ability of the network elements of switching their capacity flexibly according to the fluctuating traffic demands it is important for optimizing the load-adaptive switching regimes itself and for quality-of-service continuity in networks how this switching affects the delay as another important network parameter: For this reason, the temporal behavior of those capacity switching regimes has to be studied.

In this respect, in principle two different switching times can be distinguished on the network level (cf. the right column of **Table 1**):

- Network element switching times: Caused by switching parts of network elements on and off with respect to the required capacity, e.g. by device-level delays within network elements. This characteristic time is modeled to sum up the system und subsystem processes discussed above for empirical use by the systems integrator.
- Network-wide switching times: Caused by network organization requirements, e.g. by the distribution of routing information throughout a network. By introducing such a time scale, we wish to model inter-element processes which are not captured by the separable local processes as displayed in **Table 1**. The existence of such a time presupposes a convergent switching regime in which case it would be the governing time in an exponential convergence. In Section 3.3, this regime is discussed in detail.

## 3.1 Network element reconfiguration time modeling

When considering the gross switching times per network element, for study purposes two different switching regimes are investigated:

- Serial switching: The components of a network element can be only activated and de-activated in a strictly serial manner (i.e. the individual components can be switched on/off one after the other).
- Parallel switching: The components of a network element can be – also – activated and de-activated in parallel, if the targeted capacity change allows for this (i.e. it is large enough that several components have to be switched on/off).

In doing so, extreme cases can be obtained and conclusions for requirements regarding the respective network element design can be drawn.

Throughout this article the network element model and the nomenclature introduced in [1] is adopted where the subscripts *P*, *PG*, *LC* and *C* stand for *port*, *port group*, *line card* and *chassis*, respectively. Similarly, the superscripts *S* and *P* denote *serial* and *parallel* (switching), respectively.

*Serial switching*

The time for switching a network element from a lower initial capacity $C_{\text{in}}$ to a higher final capacity $C_{\text{fi}}$ is obtained as the sum of the times needed for activating the required components of the network element

$$T^{\text{S}}_{\text{NE ON}}(C_{\text{in}}, C_{\text{fi}}) = T_{\text{C}_{\text{ON}}} + \sum_{i=1}^{n_{\text{LC}_{\text{ON}}}} T_{\text{LC}_{\text{ON}\,i}} + \sum_{j=1}^{n_{\text{PG}_{\text{ON}}}} T_{\text{PG}_{\text{ON}\,j}} + \sum_{l=1}^{n_{\text{P}_{\text{ON}}}} T_{\text{P}_{\text{ON}\,j}}, \quad (1)$$

where $T_{\text{LC ON}}$, $T_{\text{PG ON}}$, $T_{\text{P ON}}$ denote the times a line card, a port group and a port takes to be activated, respectively, and $n$ represents the total number of each of the respective components that have to be active to achieve the final capacity $C_{\text{fi}}$. The term $T_{\text{C ON}}$ represents the time the chassis of a network element – or the basic node – takes to be switched on without configuring any additional component. The notion of *initial* and *final* capacity is to be seen with respect to a single switching process regarding a change of the network element capacity.

Similarly, the time needed for the de-activation of components of a network element in serial manner is formulated as

$$T^{\text{S}}_{\text{NE OFF}}(C_{\text{in}}, C_{\text{fi}}) = T_{\text{C}_{\text{OFF}}} + \sum_{i=1}^{n_{\text{LC}_{\text{OFF}}}} T_{\text{LC}_{\text{OFF}\,i}} + \sum_{j=1}^{n_{\text{PG}_{\text{OFF}}}} T_{\text{PG}_{\text{OFF}\,j}} + \sum_{l=1}^{n_{\text{P}_{\text{OFF}}}} T_{\text{P}_{\text{OFF}\,j}}. \quad (2)$$

*Parallel switching*

In parallel switching, to switch from a lower capacity $C_{\text{in}}$ to a higher capacity $C_{\text{fi}}$, all of the network element components necessary to achieve the final capacity will be activated simultaneously. To account for additional organization efforts in parallel processing an additional processing time $\Delta T(n_{\text{LC}}, n_{\text{PG}}, n_{\text{P}})$ is introduced in the model that depends on the number of components to be activated or de-activated – this can be e.g. the execution time of an appropriate software. Then the time it takes to switch a network element from a lower initial capacity $C_{\text{in}}$ to a higher final capacity $C_{\text{fi}}$ consists of the time it takes to activate/de-activate a single component plus the additional processing organization time:

$$T^{\text{P}}_{\text{NE ON}}(C_{\text{in}}, C_{\text{fi}}) = T_{\text{C}_{\text{ON}}} + T_{\text{LC}_{\text{ON}}} + T_{\text{PG}_{\text{ON}}} + T_{\text{P}_{\text{ON}}} + \ldots \\ \ldots + \Delta T_{\text{ON}}(n_{\text{LC}} + n_{\text{PG}} + n_{\text{P}}). \quad (3)$$

Similarly, the time required for the de-activation of components of a network element in parallel manner is described as

$$T^{\text{P}}_{\text{NE OFF}}(C_{\text{in}}, C_{\text{fi}}) = T_{\text{C}_{\text{OFF}}} + T_{\text{LC}_{\text{OFF}}} + T_{\text{PG}_{\text{OFF}}} + T_{\text{P}_{\text{OFF}}} + \ldots \\ \ldots + \Delta T_{\text{OFF}}(n_{\text{LC}} + n_{\text{PG}} + n_{\text{P}}). \quad (4)$$

The functional structure of $\Delta T$ used in the formulae above may be argued on theoretical grounds. Since, however, no exact data are available we employ this simple shape to represent the discussed effect. Later work may dwell on more refined modeling.

## 3.2 Exemplary numerical analysis and results

For an exemplary numerical evaluation a fictive, but realistic, network element is assumed: The network element has 10 slots each supporting one 40G line card while each line card supports a maximum of two 20G port groups. The line card is configured with 2-port 10GE port groups and the capacity is switched in steps of 5 % of the maximum network element capacity. Further numerical assumptions are summarized in **Table 2**.

**Table 2:** Network element components' switching times

| Component | Activation time (in min) | Deactivation time (in min) |
|---|---|---|
| basic node (chassis) | 3 | 2 |
| line card | 1 | 0.8 |
| port group | 1 | 0.8 |
| port[2] | 0 | 0 |
| $\Delta T$ | 0.1–1 | 0.01 |

**Figure 1** depicts the exemplary result for serial network element switching regimes: High switching times are obtained when differences between initial and final capacity are high, whereas low switching are needed when the respective difference is small – as expected.

[2] Activation/de-activation time for the ports is considered together with the port groups.

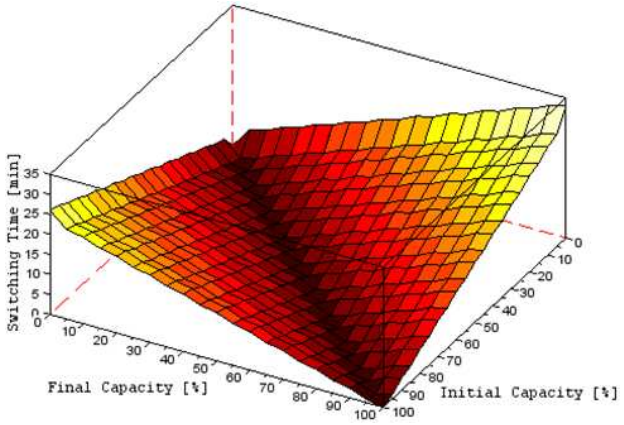


**Figure 1:** Exemplary element-wise serial switching time vs. capacity change (in %)

**Figure 2** shows the dependency of the network element switching time when operating in a parallel switching mode as a function of the difference between initial and final network element capacity and the additional processing organization time. Here, it is clearly visible that the parameter $\Delta T_{ON}$ has a critical impact on the temporal behavior of the network element and – when comparing **Figure 1** and **Figure 2** in the numerical example – serial and parallel switching regimes are in comparable regions under certain circumstances.

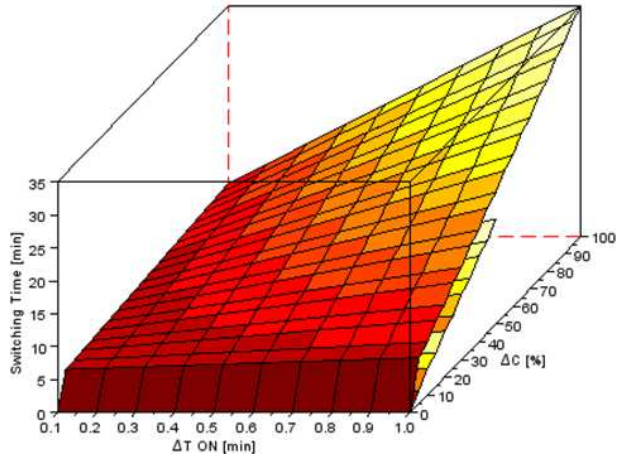


**Figure 2:** Exemplary element-wise parallel switching time vs. capacity difference (in %) and varying $\Delta T$

Now the question arises whether the parallel switching regime is better than the serial switching: **Figure 3** depicts exemplary switching time curves for serial and parallel switching regimes: It becomes obvious that the parameter $\Delta T$ is very critical. In certain cases (regarding the organization time $\Delta T$) the parallel switching requires less time than the serial switching – as expected – but in cases with high $\Delta T$ the parallel switching requires higher switching times – which is contra-intuitive at first sight and a disadvantage for load-adaptive network operation and related switching strategies.

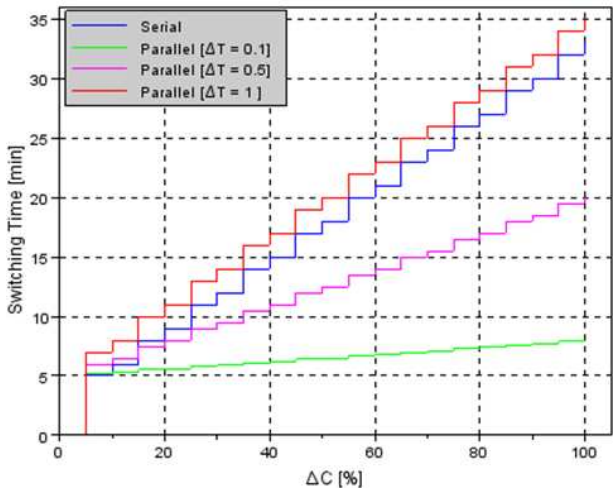


**Figure 3:** Comparison of parallel and serial switching time (parameters: $\Delta T$ = 0.1, 0.5, 1.0; capacity difference: $\Delta C$ = 0 … 100 %

## 3.3 Network-wide switching times

In a network – and in particular in large operator networks – not only the consideration of the network element switching times is sufficient, but also the interplay of the network elements requires information exchange – and, thus, time for organizing the network. Those network switching times or network organization times originate for example from distributing routing tables in IP networks or, in general, from the distribution of network state information necessary for operating a network reliably. Therefore, when introducing a load-adaptive mode into network operations, it is critical to implement a strategy that – with respect to current network operation processes – exhibits a low threshold for adopting the new, dynamic operation paradigm. A network switching strategy therefore should decouple the logical and the physical switching in order to maintain the network stability and reliability. Two sequential steps are favorable:

1. Logical switching: Adaptation of the link weights in order to "manipulate" current routing algorithms in a way that the desired result from the aspect of energy efficiency improvement is obtained, e.g. regarding traffic routing in the network.
2. Physical switching: Once step 1 is safely converged onto a stable state, links that are freed from traffic are physically switched off and the energy saving mechanisms implemented in the network element hardware become effective.

It should be mentioned that "network-wide" in this context refers to only the meshed part of the network since only in this part centrally triggered switching via a strategy as the one discussed above is necessary. Energy saving mechanisms in the network parts with star- or tree-like propagation structure may be triggered completely autonomously on the basis of local information. Hence, interaction with other network elements does not occur and

no contribution to the organization time considered here arises.

# 4 Model validation

In order to validate the model derived in Section 3 we analyzed the activation behavior of interfaces and line cards of a state-of-the-art OTN (Optical Transport Network) node. As this network element not only performs optical transport but also multiplexes and switches data on sub-lambda level, different activation mechanisms could be investigated. In the following, we consider a line card (LC) consisting of port groups (PG) with two ports each.

During product development it is common to re-use existing hardware and software designs and extend it for new models and/or products: One of the early line cards of the analyzed NE consisted of one PG while the newer models have up to five PGs. Below we will show how the re-used design became inefficient and describe the options we identified to speed up PG and LC activation times.

On the LC a flash device contains selectable FPGA images for the PGs. A special purpose device handles image updates, CRC calculations on the flash and FPGA pumping from flash towards FPGA devices. The special purpose device is reachable along an "Image bus" while the initializations and configurations of the PGs are done via a "Config bus" (**Figure 4**).

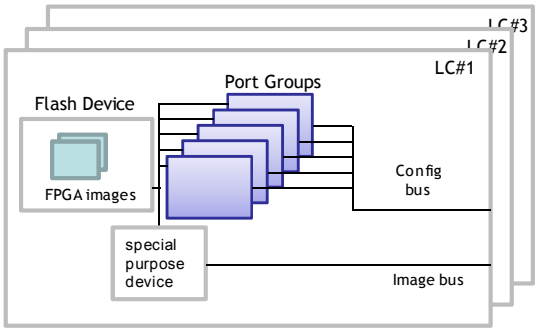


**Figure 4:** Line card architecture

In order to establish data traffic on a LC, software has to select the required image from flash, pump it into the FPGA device before initializing and configuring it. In our example we have two software components responsible for that: an equipment manager (EM) who takes care of getting the FPGAs to work with initialized values, while a transmission manager (TM) deals with the real traffic configurations.

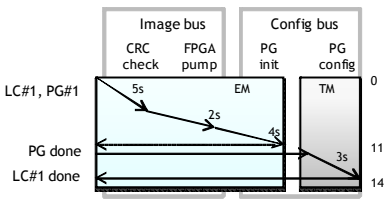


**Figure 5:** Activation sequence for 1 LC / 1 PG

While on the original LC traffic was running after 14 seconds (**Figure 5**), it takes approximately 70 seconds to get traffic on the new LC with five PGs that re-used the old design (**Figure 6**). This behavior can be modeled best by two serial processes.

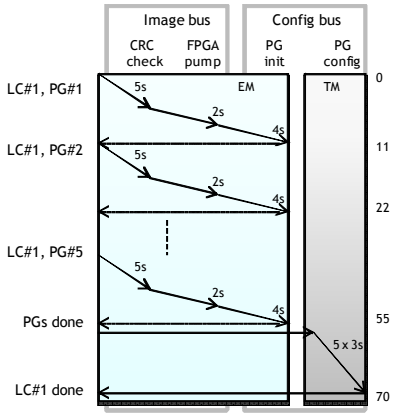


**Figure 6:** Activation sequence for 1 LC / 5 PG

For two LCs, the startup time becomes "number of LCs" × 70 seconds = 140 seconds when each LC is handled in sequential order.

Alternatively to sequential handling, an improvement in software design would be to use separate software threads inside a process, for example the EM: Each thread takes care of its own LC (**Figure 7**). This only makes sense if the various hardware resources (like Config- and Image bus) allow for multi-threaded and/or multi-process requests. Modeling the behavior of **Figure 7** is more complex due to the unfavorable mixture between software (process) and hardware (bus) granularities.

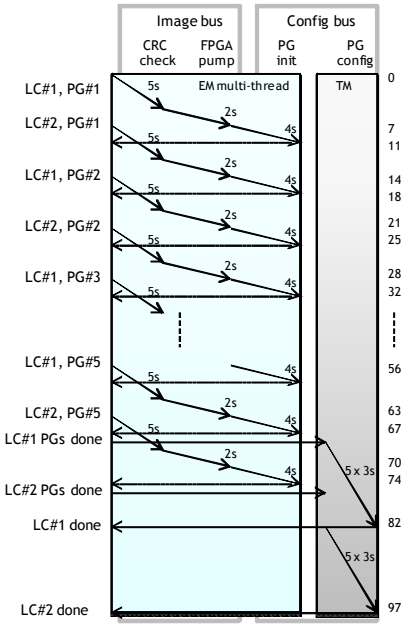


**Figure 7:** Activation sequence for 2 LC / 5 PG

Further optimizing the Image/Config bus software infrastructure enables fully parallel PG initialization (**Figure 8**). This design can again easily be modeled by a combination of serial and parallel processes. Note that the behavior is still not optimal as the TM process can start the configuration only after all PGs are initialized.

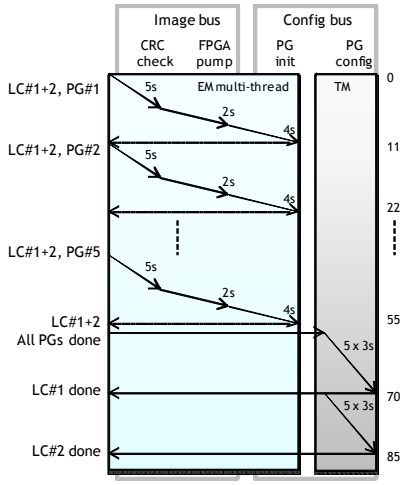


**Figure 8:** Activation sequence for 2 LC / 5 PG (w/ additional Image bus software infrastructure optimization)

Removing that deficiency leads to the startup behavior of **Figure 9**.

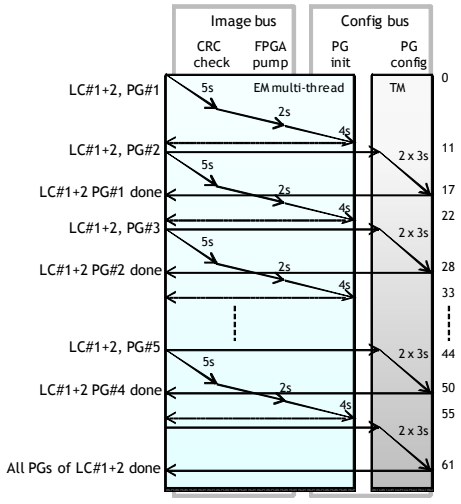


**Figure 9:** Activation sequence for 2 LC / 5 PG (w/ additional TM process optimization)

It becomes clear that when the total amount of LCs increase, extra delays will be introduced by TM.
As the Config bus is used by both TM and EM, at some point in time this bus could become the bottleneck in the system. This is expressed by the term $\Delta T$ in Section 3.1. By dividing the chassis into segments and having multiple Config busses this restriction can be mitigated again.

# 5 Conclusion

In this article, we have presented a categorization of power management techniques according to their temporal properties. Network-level technologies that affect several systems and thus require coordination, e.g. by an energy-aware control plane, have been identified as most critical with respect to network performance degradation.
In order to better understand network resource activation processes the network element power-vs.-capacity model developed in [1] has been extended in a way, that now also the switching times of the network elements and their components are included. Different switching strategies of a network element's components – serial and parallel – have been considered and evaluated and important parameters have been identified and modeled. So, the extended model reflects in addition to the previous one from [1] an essential parameter for quality of service continuity in networks and this way can serve as a basis for network simulations and optimizations.
An analysis of the line card activation behavior for a particular network element shows good agreement with the derived theoretical models. Further work remains to be done for inclusion of other network-wide switching time contributors that were not part of this work, namely topology and protocol convergence processes.

# Acknowledgement

The authors wish to thank Dirk Kosiankowski and Andreas Betker of Deutsche Telekom, Telekom Innovation Laboratories, as well as Mudesir Mohammed Ali of University of Applied Sciences (Hochschule für Telekommunikation, HfTL) Leipzig for valuable discussions and support.
The work leading to this article was supported by the IT2Green programme of the German Federal Ministry of Economics and Technology in the project DESI (Durchgängig Energie-Sensible IKT-Produktion).

# References

[1] Lange, C.; Schlenk, R.; Lehmann, H.: Network Element Characteristics for Traffic Load Adaptive Network Operation. In: *13. ITG-Fachtagung Photonische Netze*. Leipzig, 07.–08. Mai 2012 — ITG (Hrsg.): *Photonische Netze*. ITG-Fachbericht (Band 233), Berlin; Offenbach: VDE-Verlag, 2012, pp. 124–131

[2] Van Heddeghem, W.; Idzikowski, F., Vereecken, W.; Colle, D.; Pickavet, M.; Demeester, P.: Power consumption modeling in optical multilayer networks. *Photonic Network Communications*, 24 (2012), October, no. 2, pp. 86–102

[3] International Telecommunication Union – Telecommunication Standardization Sector (ITU-T): Framework of energy saving for future networks.

ITU-T Rec. Y.3021, January 2012, <http://www.itu.int>

[4] Christensen, K.; Gunaratne, C.; Nordman, B.; George, A.: The next frontier for communications networks: power management. *Computer Communications*, 27 (2004), December, no. 18, pp. 1758-1770

[5] Nedevschi, S.; Popa, L.; Iannaccone, G.; Ratnasamy, S.; Wetherall, D.: Reducing network energy consumption via sleeping and rate-adaptation. In *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*. — USENIX Association (ed.): *NSDI'08*. Berkeley, CA, USA, 2008, pp. 323-336

[6] Advanced Configuration and Power Interface (ACPI) Specification, Revision 4.0a, April 2010, <http://www.acpi.info>

[7] David, H.; Fallin, C.; Gorbatov, E.; Hanebutte, U.; Mutlu, O.: Memory power management via dynamic voltage/frequency scaling. In: *Proceedings of the 8th ACM international conference on Autonomic computing*. Karlsruhe, 14–18 Juni 2011 — ACM (ed.): *ICAC '11*. New York, NY, USA, 2011, pp. 31–40

[8] JEDEC Solid State Technology Association: DDR3 SDRAM Standard, Document JESD79-3, July 2012, <http://www.jedec.org>

[9] Bolzani, L.; Calimera, A.; Macii, A.; Macii, E.; Poncino, M.: Enabling concurrent clock and power gating in an industrial design flow. In: *Design, Automation & Test in Europe Conference & Exhibition*. Nice, France, 20–24 April 2009 — EDAA (ed.): *DATE '09*. 2009, pp. 334–339

[10] Xilinx Inc., Spartan-6 FPGA, Data Sheet DS162, October 2011, <http://www.xilinx.com>

[11] Gleixner, T.; McKenney, P.; Guittot, V.: Cleaning Up Linux's CPU Hotplug For Real Time and Energy Management. In: *24th Euromicro Conference on Real-Time Systems*. Pisa, July 2012 — ACM (ed.): SIGBED Review (vol. 9, no. 4), November 2012

[12] Huang, H.; Pillai, P.; Shin, K.: Design and Implementation of Power-Aware Virtual Memory. In: *Proceedings of the General Track: 2003 USENIX Annual Technical Conference*. San Antonio, Texas, June 9-14, 2003 — USENIX Association (ed.): *USENIX'03*. 2003, pp. 57–70

[13] International Telecommunication Union – Telecommunication Standardization Sector (ITU-T): Asymmetric digital subscriber line transceivers 2 (ADSL2). ITU-T Rec. G.992.3, April 2009, <http://www.itu.int>

[14] Institute of Electrical and Electronics Engineers (IEEE): Energy Efficient Ethernet. IEEE P802.3az, <http://grouper.ieee.org/groups/802/3/az>

[15] Institute of Electrical and Electronics Engineers (IEEE): Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications – Amendment 5: Media Access Control Parameters, Physical Layers, and Management Parameters for Energy-Efficient Ethernet. IEEE 802.3az-2010, September 2010, <http://www.ieee.org>

[16] International Telecommunication Union – Telecommunication Standardization Sector (ITU-T): GPON power conservation. ITU-T Rec. G.Sup45, May 2009, <http://www.itu.int>

[17] Gobriel, S.; Maciocco, C.; Tai, T.: When Idle Is Not Quiet: Energy-Efficient Platform Design in Presence of Network Background and Management Traffic. In: *Global Telecommunications Conference (GLOBECOM 2010)*. Miami, Florida, 6–10 December, 2010 — IEEE (ed.): *GLOBECOM'10*. 2010, pp.1–6