8 Early Energy Estimation of Heterogeneous Embedded Networks within Adaptive Systems

Patrick Heinrich[1], M.Eng., **Dietrich Gossen**[1], B.Sc., Dr. **Erik Oswald**[1], Prof. Dr.-Ing. **Rudi Knorr**[1,2], [1]Fraunhofer Institute for Embedded Systems and Communication Technologies ESK, Munich, Germany, [2]Chair for Communication Systems, Institute of Computer Science, University of Augsburg, Germany

Abstract

This paper presents and evaluates a new approach of modeling energy consumption of communication within adaptive networked embedded systems. The objective is to enable energy estimation within early phases of system development, which allows system designers to compare different allocations of software components.
As networked embedded systems consist of multiple specialized networks (with different protocols and topologies) and are characterized by a high degree of interaction, existing network-centric approaches have significant disadvantages describing entire systems. To overcome this problem a model was created which is based on individual communication connections between software components. This enables technology-transparent mapping to network topologies (across borders of networks) which significantly simplifies the evaluation of different software placements.
The energy estimation model was evaluated comparing the estimated values with real-world measurement results.

1. Introduction

Embedded networked systems of modern cars consist of up to 80 independent electronic control units (ECUs) which cooperate to realize complex applications. These systems are characterized by a high degree of interaction and consist of different (specialized) communication networks with different protocols and topologies [1]. Future cars will probably enable autonomous driving which further increases the number of ECUs and the interaction between these ECUs. However, future automobiles will also be adaptive to decrease the energy consumption by deactivating temporarily unnecessary applications. Hereby, the chosen placement of software components on the network's ECUs is relevant w.r.t. energy consumption [2]. This is caused by different applicable energy saving states and energy necessary to (de)activate hardware and software. That placement is done by system designers relatively early within the development process [3], which makes it necessary to estimate the later energy consumption at that time using the available information.

In today's luxury-class vehicles for instance, the electrical and electronic components draw up to 2.5 kW ([4], [5]). Compared to what the vehicle engine requires (for example 55 kW), 2.5 kW seems small. However, the electrical components consume energy during every mode of operation, even when in standby mode. The vehicle engine consumes most of its energy during acceleration and even here the maximum power is seldom demanded. Communication causes an increasing amount of energy consumption of networked embedded systems. (The hardware setup of the evaluation section of this paper, for example, uses up to 20 % of the energy consumption for communication.) An increase of 100 Watt thus means that fuel consumption rises by 0.1 liter per 100 kilometer, leading to an increase in CO_2 emissions of 2.5 gram per kilometer [4]. This illustrates the considerable potential for energy savings, an aspect that should be factored in during the development process.

This paper presents an energy estimation model which focuses on energy consumption caused by communication. This includes communication controllers and transceivers of automotive ECUs including times of idle and other energy saving states. Additionally, functional properties are modeled, such as gateway functionalities between networks and the reception of non-required messages. The main advantages are the modeling based on individual communication connections between software components and the technology-transparent mapping to network topologies (across borders of networks) which significantly simplifies the evaluation of different software placements compared to existing network-centric approaches (cf. Section 2). The model supports different kind of communication protocols and allows different topologies (e.g. bus, ring, star) of the networked embedded system. The energy estimation model is evaluated comparing the estimated values with real-world measurement results. Therefore, a fixed hardware setup of three ECUs and one gateway with varying software components and placements are implemented.

The paper is structured as follows: Section 2 discusses different existing energy estimation models. In Section 3 the characteristics of adaptive networked embedded systems including the considered constraints are presented. Within Section 4 the energy estimation model is presented in detail. The model is evaluated within Section 5 using an exemplary embedded system and the paper closes with conclusion and future work in Section 6.

2. Related Work

Existing energy models are categorizable concerning their target system for example global networks (e.g. the Internet), radio networks and (wired) embedded networks. The characteristics of these networks are very different. Global networks include for example routers and access networks and long-distance data transmissions and thousands of participants using a plurality of network protocols. Compared to those networks, embedded networks have short cable lengths and a lot of communication is realized as one- or two-hop communication. That is why energy losses resulted by cable lengths are commonly neglected within embedded systems [6] and routers and others are normally not part of embedded systems. This causes that energy models within the area of global networks are not applicable for embedded networks. That includes for example research on global networks with the focus on optic fibers as physical layer as modeled by Tucker et al. [7], the comparison of communication architecture styles (client-server and publish-subscribe) w.r.t. energy consumption [8] and also models for radio networks which commonly focus propagation, absorption and transmission errors with the aim to minimize energy consumption per node, e.g. by clustering [9].

Modeling energy consumption of typical automotive networks is done by Balbierer et al. [10] by modeling every network on its own. The energy consumption depends on the network load given by a percentage value. The model is not applicable to heterogeneous embedded networks with different kind of networks and topologies caused by the difference of the individual models. Gurun and Krintz [11] use a set of first order, linear regression equations to model the energy consumption during runtime of the system. This approach is not usable during the development of embedded systems and moreover idle and sleep states are not supported.

The fact that energy is the product of power and time may induce the assumption to enable energy estimation by time estimations. However, this would require constant power consumption within the embedded network which is not the case – especially not for adaptive systems which deactivate unused hardware and software components. Different power levels of components are not included within timing analysis (which often focuses on worst-case times) such as [12], [13], [14]. Analyzing the time for communication, for example, does not consider the number of network participants which also consume energy by receiving communication.

As shown above, existing models have significant disadvantages related to modeling heterogeneous networks of networked embedded systems. Additionally, changing one communication connection would result a complete recalculation of the entire system. This is especially disadvantageous for adaptive systems. For that reason, an energy estimation model was created which enables the easy evaluation of communication relevant energy consumptions within adaptive networked embedded systems. This model is presented and evaluated within the next sections.

3. Adaptive Networked Embedded Systems

In the following subsection the characteristics of adaptive networked embedded systems and the constraints of energy estimation models are presented.

Networked embedded systems as found within the automotive sector are characterized by a high degree of heterogeneity [15]. This means different kind of ECUs by various suppliers and different types of communication protocols and topologies resulted by specialized technologies. Common communication topologies are bus, ring and star which are used within this kind of networked embedded systems as shown within Figure 1. Embedded systems within the automobiles commonly execute safety-critical applications which results in real-time constraints, for example concerning transmission and response times. Through that the communication behavior differs e.g. to consumer electronics, i.e. within automotive embedded systems communication messages are commonly transmitted cyclic and using a small bandwidth.

Adaptivity within embedded systems enables the deactivation of temporarily unnecessary functionality including its hardware. This enables the design of more energy efficient systems, but significantly enlarges the number of possible system designs. Furthermore, factors such as time and energy for (de)activation and availability of different kind of sleep modes are relevant now. Adaptivity necessitates considering the whole systems instead of just looking at individual components, because it is possible that individual sub-optimal systems form together an optimized system [17].

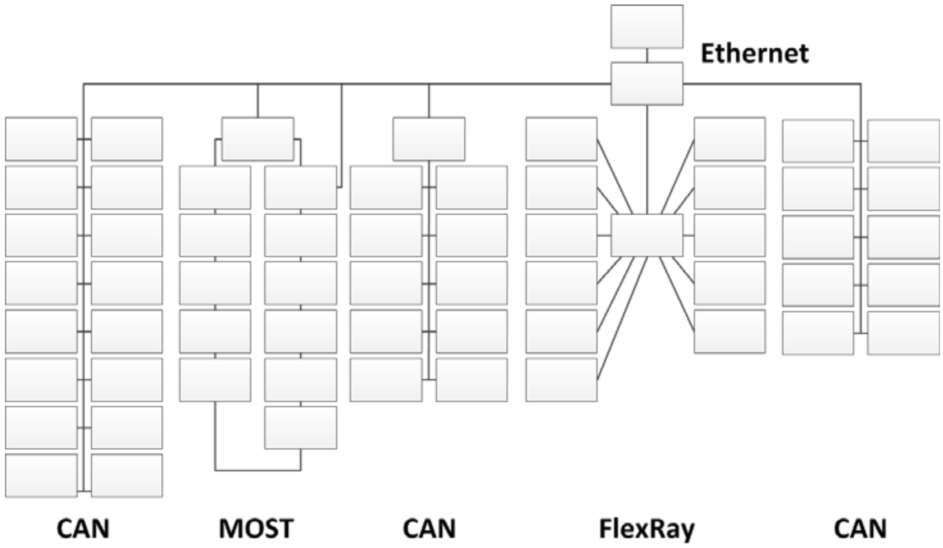

Figure 1: *Typical networked embedded system (BMW 7 series) [adapted from BMW 2005, quoted from [16], p.34]*

System designers of networked embedded systems are faced with a lot of different constraints and limitations (e.g. performance and memory limitations, real-time constraints and heterogeneity). The work of this paper focuses the energy consumption of embedded networks and components which are directly influenced by communication, such as components with gateway functionality. Through that the considered constraint for system design within this paper is only the limitation of network bandwidth within individual networks (cf. Equation 1). The bandwidth within a network *net* is described as b_{net}, which is the sum of bandwidths of individual communication connections b_{com}. The maximum bandwidth per network is defined by $b_{net,max}$.

$$b_{net} = \sum_{j=0}^{n_{com}} b_{com,j} \leq b_{net,max} \tag{1}$$

Within the following section the energy estimation of mixed embedded networks is explained including assumptions and challenges.

4. Energy Estimation of Embedded Networks

In this section the energy estimation model is presented. At the beginning the requirements and challenges of energy estimation of heterogeneous embedded networks are discussed and the necessary information presented. Afterwards, the energy model itself and the underlying assumptions are explained.

4.1 Requirements and Challenges

Energy estimation during early stages of the design phase enables system designers to evaluate their design choices with respect to the energy consumption of the later system. This enables for example the comparison of different placements of software components on ECUs. The earlier energy evaluations are possible during the development process, the more possibilities to influence the design towards more energy efficiency are available, because fewer decisions have already been made [3].

An energy estimation model has to fulfill the following functional/technical requirements to be usable for networked embedded systems:

- **Communication protocols** – The energy estimation model has to be generic enough to be usable for different communication protocols and technologies. That means the model is usable for example for CAN, FlexRay or Ethernet.
- **Topology and communication patterns** – Networks such as within automotive, networked embedded systems consist of several individual networks. That means, an energy estimation model has to deal with different topologies (e.g. bus, ring, star) and communication patterns (e.g. uni-directional, multicast, full-duplex).
- **Bandwidth and payload/overhead size** – The energy model has to be parametrizable for several bandwidths and different payload and overhead sizes within the networks. Also the transfer between different networks and checking the bandwidth limitations must be possible.
- **Interaction between networks (gateways)** – To realize complex applications different networks within networked embedded systems have to communicate and exchange information. Through that specialized control units with gateway functionality are involved to handle that kind of communication, which are necessary to be modeled.

Furthermore, the estimation model should fulfill the following non-functional requirements:

- **Performance and overhead** – Changing the task placement should not cause a complete recalculation of the system's energy consumption. Changing or adding a communication connections should just cause the recalculation of affected energy consumptions.
- **Automatable** – It has to be possible to automate the energy estimation. This enables the use of optimization algorithms to automatically optimize the energy consumption of the later system.
- **Scalable** – The energy estimation should be scalable concerning the number of communication connections and ECUs.

Within the following subsection the assumption are presented on which the energy consumption model is based.

4.2 Assumptions

The energy consumption model is based on the following underlying assumptions:

- The communication bandwidth is evenly distributed, caused by the cyclic communication which is very typical within real-time critical embedded systems to increase robustness against short-term errors [6].

- Communication nodes, which are active within the current system configuration *c*, but do not communicating at time *t*, are within an idle mode and receive messages on the network (if technically possible, e.g. within CAN networks).
- Communication nodes, which are not active within the current system configuration *c*, are within a sleep mode and do not receive messages on the network (even within networks such as CAN).
- Communication networks are not operating at full capacity, e.g. 40 % of maximum capacity is a common value for CAN within the automotive industry [18]. This is a general practice of system designers, because of a more constant timing behavior of real-time systems (for example caused by less collisions and retransmissions at the communication networks).[1]

The following section presents the energy estimation model and explains its details.

4.3 Energy Estimation Model

The presented model focuses on energy consumption resulted by communication. This includes communication controllers (CC) and transceivers (TRX) of ECUs including times of idle and sleep. Additionally, functional properties are modeled, such as gateway functionality between networks and reception of non-required communication. The main advantages are the modeling based on individual communication connections between software components and the technology-transparent mapping to network topologies (across borders of networks) which significantly simplifies the evaluation of different software placements, compared to existing network-centric approaches (cf. Section 2). The model supports different kind of communication protocols and allows different topologies (e.g. bus, ring, star) within the networked embedded system. The further subsections explain structure, parameters and usage of the presented energy estimation model.

1) Structure of the Model: The model enables the estimation of energy consumption of components which are part of communication within networked embedded systems. The model is depicted within Equation 2. The total energy consumption of communication E_{Com} consists of active and inactive ECUs. Active ECUs transmit and/or receive communication. ECUs which are part of a communication connection consume energy which is represented by E_{Conn} (Equation 3) per each connection. Energy consumption of ECUs resulted by receiving non-required communication is represented by $E_{Listener}$ (cf. Equation 4)[2]. Additionally, CC and TRX consume a constant amount of energy during transmitting and receiving. This energy consumption is independent of the amount of data and is represented by E_{Offset}. (Note: During times of no communication this values also represents the energy consumption resulted by idling of CC and TRX.)

---

[1] Through that the energy consumption varies linear with the number of messages/bandwidth, caused by the assumption of neglectable collisions. (Note: It would be easily possible to extend the energy estimation model to represent non-linearity, e.g. by evaluating the total bandwidth within a network.) Caused by the fact that operation at full capacity is very unusual within real-time embedded systems and the intention to preserve the advantages of energy estimation based on communication connections, this was not realized within the energy estimation model.

[2] Networks with bus-based topology (e.g. CAN or FlexRay) are characterized by the fact, that every ECU within the network receives every message, which also consumes energy.

Inactive ECUs are deactivated (e.g. switched to sleep mode) which means CC and TRX consume energy resulted by sleeping (E_{Sleep}). (Note: More than one sleep or standby mode may be available. In that case the equation is simply extended by an additional sum which adds for example the energy consumptions of $E_{Standby}$.)

$$E_{Com} = \overbrace{\underbrace{\sum_{i=0}^{\#ComConn} (E_{Conn,i} + E_{Listener,i})}_{\text{Communication Connections}} + \sum_{j=0}^{\#Networks} \sum_{k=0}^{\#ActiveECUs} (E_{Offset,j,k})}^{\text{Active ECUs}} + \overbrace{\sum_{l=0}^{\#InactiveECUs} (E_{Sleep,l})}^{\text{Inactive ECUs}} \tag{2}$$

$$E_{Conn} = \overbrace{\underbrace{E_{TX,Src}}_{\text{"Source ECU"}} + \underbrace{\sum_{m=1}^{\#ParticipatingECUs-2} (E_{RX,m} + E_{Transfer,m} + E_{TX,m})}_{\text{ECUs with gateway functionality within this communication connection}} + \underbrace{E_{RX,Sink}}_{\text{"Sink ECU"}}}^{\text{ECUs which are part of the communication connection}} \tag{3}$$

$$E_{Listener} = \sum_{n=0}^{\#Listener} (E_{RX,n}) \tag{4}$$

As depicted within Equation 3 the value of E_{Conn} is calculated by the sum of the energy values which are necessary to transmit (E_{TX}) and receive (E_{RX}) communication. The communication starts at the "Source ECU" ($E_{TX,Src}$) and ends at the "Sink ECU" ($E_{TX,Sink}$). If a direct communication (one-hop) is not available, then ECUs with routing/switching or gateway functionality are necessary. (Within automobile embedded systems gateways commonly transfer and translate between different networks.) These ECUs are modeled by the energy consumption of receiving ($E_{RX,m}$), transmitting ($E_{TX,m}$) and transferring (and/or translating) the communication to other networks ($E_{Transfer,m}$). Equation 5 depicts the calculation of E_{TX} which value depends on the number of ECUs (n_{ECU}) within a network. This means a network with two ECUs (e.g. communicating directly to each other) consumes E_{TX_2n}. Adding more ECUs to the network increases the energy consumption by the value of E_{TX_xn} per ECU.

$$E_{TX} = E_{TX\_2n} + (n_{ECU} - 2) \cdot E_{TX\_xn}, \quad n_{ECU} \geq 2 \tag{5}$$

As shown the energy estimation model is based on individual communication connections which enables an easy calculation, e.g. for adaptive systems or optimization algorithms, because only changed communication connections and network-wide values need to be updated. Another benefit is the usability of that model with arbitrary topologies (including mixtures of networks and topologies), caused by the connection-centric approach.

2) Parameterization of the Model: The previously presented model is based on individual communication connections which make it necessary to parameterize every

connection on its own. However, the number of parameters is limited by the number of ECUs within the embedded system, because software components placed on the same ECU have the same energy consumption for receiving and transmitting.
The values of E_{TX}, E_{RX} and $E_{Listener}$ depend on the used hardware, the kind of network and the used protocol including its configuration. The power consumption per hardware module is for example given by the manufacturer or even educated guesses are possible. To calculate the energy consumption it is necessary to know the transmitting and receiving time, which is derivable from the given bandwidth (e.g. based on the amount of messages or bits). Depending on the network more or less payload overhead per message is necessary. For this paper it is assumed to efficiently use the communication bandwidth, i.e. using the maximum payload to transmit the data. This results in a necessary number of messages and through that the amount of overhead. Multiplying with the time per bit – derived from the data rate of the networks – results in the transmission and reception times. $E_{Transfer}$ depends on the used microcontroller (µC). These values are estimateable using energy estimation techniques (such as [19] and [20]) or measurements on evaluation boards are possible, if available. The value of E_{Sleep} depends on the available energy saving capability of the used communication hardware. These values are generally given by the hardware manufacture, because this is already relevant for today's system designers.

3) Usage of the Model: The previous subsections presented how to differentiate between communication protocols, different bandwidth and protocol overheads. Different topologies and communication patterns are also representable by the presented connection-based approach. More precisely, the kind of communication within different topologies has to be emulated which is easily possible. Switched Ethernet with full-duplex for example is emulated using two unidirectional communication links. Emulating a MOST topology (ring) is done by unidirectional communication between the consecutive ECUs. Bus topologies (e.g. CAN or FlexRay) are represented by unidirectional communication links from sending ECUs to all other (receiving) ECUs of the bus network. Different communication patters are emulated in the same manner as described above using a formal description of software functions and their relations. For example multicast-communication is represented by data flows with multiple receivers.

Using the energy estimation model also allows the check of constraints such as the maximum bandwidth usage (cf. Section 3). This maximum is given per network link which correlates with the topology emulation presented above. The sum of communication connections on one hardware connection is compared to the given maximum value. (Note: More specific constraint checks are possible such as the maximum switching bandwidth within switches. Such extensions are possible, but not part of this paper.) This bandwidth checks are possible on network- and ECU-base which is relevant for example for automatic optimization of software component placement.

As shown the different requirements and challenges of Section 4.1 are performed by the energy model. The calculation of energy consumptions are automatable using a formal description of the system and the given energy estimation equations. Caused by the calculation of the energy consumption based on (individual) communication connections the model scales with the number of communication connections within this kind of networked embedded systems. The following section evaluates the presented energy estimation model using an exemplary embedded system.

5. Evaluation

The presented energy estimation model is evaluated using an exemplary networked embedded system consisting of three ECUs and one gateway. This system is used to measure the energy consumption of communication resulted by different placements of software components and compare these real-world measurements to the estimated values. The analyzed communication represents possible choices of system design and is chosen to cover different possibilities within the available design space. Within the next subsection the analyzed communication connections are defined. Afterwards, the exemplary embedded system and the measurement setup is explained. The section closes with the comparison to real-world measurement results and discussion.

5.1 Communication Connections

To evaluate the energy estimation model different communication connections are analyzed within a network of ECUs. This network consists of three “normal” ECUs and one gateway ECU. The topology of the exemplary embedded system is shown in Figure 2. The topology with two networks was chosen, because this system represents all components of the energy estimation model, i.e. “Source ECU”, “Sink ECU” and ECUs working as gateway or listener (cf. Section 4.3). First of all, single communication connections are evaluated, i.e. one ECU sends messages to another ECU. For example 1 → 3, 3 → 4 or 3 → 1. If the receiving ECU is not within the same network, the communication has to be transferred by the gateway ECU (*#2*). To evaluate different loads of the network, the communication connections are repeated using different cycle times. Afterwards, mixed communication connections are evaluated which transmit data in parallel.

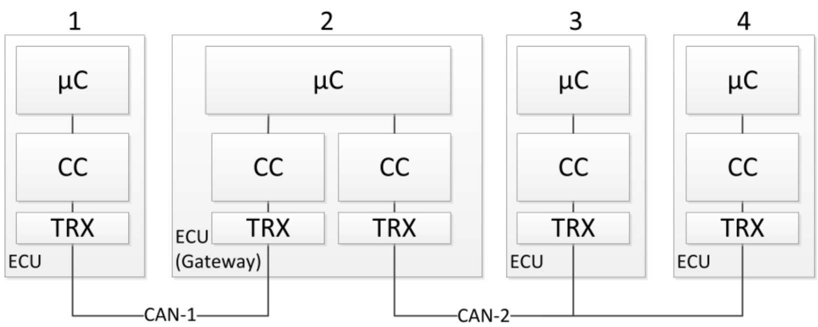

Figure 2: *Evaluated embedded system with three “normal” ECUs and one gateway ECU. (Note: The numbers above are just for identification.)*

The following subsection details the measurement setup and the hardware used for the exemplary embedded system.

5.2 Measurement Setup

The exemplary embedded system is composed of four evaluation boards as described above. Three “normal” ECUs are represented by MCP2515 CAN Bus Monitor Demo Boards of Microchip Technology Inc. with Microchip PIC18F4550 processor, MCP2515 CAN communication controller and MCP2561, a second generation high-speed CAN transceiver of Microchip. The ECU with gateway functionality is represented by a chipKIT Max32 Prototyping Platform of Digilent, Inc. with a Microchip PIC32MX795F512L processor and two CAN communication controllers on-chip. The transceivers of that platform are realized by an additional board, i.e. the chipKIT Network Shield Board of Digilent, Inc. with two Microchip MCP2551 high-speed CAN Transceivers. The transmitted CAN frames are all equal and consist of eight bytes payload and evenly spread number of “dominant” and “recessive” bits. The energy consumption is measured using a DC Power Analyzer N6705B of Agilent Technologies, Inc. which is equipped with “N6762A Precision DC Power Modules”. The analyzer enables very precise measurements in the microampere region, i.e. the accuracy of voltage output (low range) is 0.016 % + 1.5 mV and for current output (low range) is 0.04 % + 15 µA. The maximum sampling rate is 48.9 kHz. The following subsection presents and discusses the measurement results.

5.3 Measurement Results and Discussion

The previously presented measurement setup was used to obtain real-world values and compare these with estimations of the energy estimation model (cf. Section 4.3). The results are shown within Table 1. At first, single communication connections with different cycle times were measured. The range of error of these estimations varies between -0.7 % and +1.8 %. Afterwards, different communication connections were mixed to model a more realistic network. Using different cycle times the range of error varies between -2.4 % and +2.8 %. The remaining error is resulted by unpredictable collisions within networks which use a shared medium and probably by different realizations of the component hardware. The energy estimation model shows a good accuracy even for mixed communication connections.

***Table 1**: Measurement results of different communication connections within the exemplary embedded system*

Single Communication Connections (Cycle Time: 0.5 ms)						
	1 → 3	1 → 4	3 → 1	3 → 4	4 → 3	4 → 1
Estimated energy cons. *[mWs]*	345.60	345.60	345.60	278.93	278.93	345.60
Measured energy cons. *[mWs]*	351.31	351.80	345.64	281.60	281.82	345.60
Error	1.62 %	1.76 %	0.01 %	0.95 %	1.03 %	0.00 %
Single Communication Connections (Cycle Time: 2.5 ms)						
	1 → 3	1 → 4	3 → 1	3 → 4	4 → 3	4 → 1
Estimated energy cons. *[mWs]*	217.61	217.61	217.61	204.27	204.27	217.61
Measured energy cons. *[mWs]*	217.13	217.38	216.11	203.30	203.33	216.16
Error	-0.22 %	-0.11 %	-0.69 %	-0.48 %	-0.46 %	-0.67 %
Mixed Communication Connections (Cycle Time: 2.5 ms)						
	1 → 3, 1 → 4		1 → 3, 3 → 4, 4 → 1		1 ↔ 3, 3 ↔ 4, 4 ↔ 1	
Estimated energy cons. *[mWs]*	243.60		265.49		272.60	
Measured energy cons. *[mWs]*	241.80		267.17		280.33	
Error	-0.75 %		0.63 %		2.76 %	
Mixed Communication Connections (Cycle Time: 5 ms)						
	1 → 3, 1 → 4		1 → 3, 3 → 4, 4 → 1		1 ↔ 3, 3 ↔ 4, 4 ↔ 1	
Estimated energy cons. *[mWs]*	204.27		214.61		224.98	
Measured energy cons. *[mWs]*	203.33		209.54		225.07	
Error	-0.46 %		-2.42 %		0.04 %	

6. Conclusion and Future Work

This paper presented and evaluated a new approach of modeling energy consumption of communication within adaptive networked embedded systems such as found within modern automobiles. These systems commonly consist of multiple specialized networks (with different protocols and topologies) and are characterized by a high degree of interaction. To overcome the problems of modeling heterogeneous embedded networks an energy estimation model was created which is based on individual communication connections between software components. This enables technology-transparent mapping to network topologies (across borders of networks) which significantly simplifies the evaluation of different software placements. The energy estimation model was evaluated comparing the estimated values with real-world measurement results. Therefore, a fixed hardware setup of three ECUs and one gateway ECU with varying software component placements were implemented and their communication analyzed, which pointed out a range of estimation error between -0.7 % and +1.8 % for individual and -2.4 % and +2.8 % for mixed communication connections. The model is able to represent different communication protocols and topologies which enable the usage within other areas of networked embedded systems such as aircrafts and industrial systems.

Future work will include the modeling of a larger system to evaluate the energy saving potential by optimizing the placements of software components within networked embedded systems. Additionally, the presented energy estimation model will be included into the energy estimation model of entire networked embedded systems which include microcontrollers and sensors/actuators. Beside the energy consumption of communication itself the model influences the constraints of ECUs. This is occurred by the resulting bandwidth per ECU which can also be used to define the lower bound of the ECU's active times, because during the time of communication the ECUs normally have to be awake and are not able to use energy saving techniques.

Acknowledgement

The research leading to these results has received funding from the German Federal Ministry for Economic Affairs and Energy (BMWi).

References

[1] A. Barthels, J. Fröschl, H.-U. Michel, and U. Baumgarten, "An Architecture for Power Management in Automotive Systems," in Architecture of Computing Systems – ARCS 2012, ser. Lecture Notes in Computer Science, vol. 7179. Springer Berlin Heidelberg, 2012, pp. 63–73.

[2] C. Schmutzler, A. Krüger, F. Schuster, and M. Simons, "Energy efficient automotive networks: state of the art and challenges ahead," International Journal of Communication Networks and Distributed Systems, vol. 9, no. 3/4, 2012.

[3] J. Weber, Automotive Development Processes: Processes for Successful Customer Oriented Vehicle Development. Springer-Verlag Berlin Heidelberg, 2009.

[4] A. Monetti, T. Otter, and N. Ulshöfer, “Spritverbrauch senken, Reichweite erhöhen: System-Basis-Chip für den Teilnetzbetrieb am CAN-Bus,” *Elektronik Automotive*, no. 11, pp. 24–27, 2011.
[5] Arthur D. Little, “Market and Technology Study Automotive Power Electronics 2015,” 2006.
[6] C. Schmutzler, Hardwaregestützte Energieoptimierung von Elektrik/Elektronik-Architekturen durch adaptive Abschaltung von verteilten, eingebetteten Systemen: Dissertation, ser. *Steinbuch Series on Advances in Information Technology*. Karlsruhe: KIT Scientific Publishing, 2012, vol. 7.
[7] R. Tucker, “Green Optical Communications – Part II: Energy Limitations in Networks,” *IEEE Journal of Sel. Top. in Quantum Electronics*, vol. 17, no. 2, 2011.
[8] C. Seo, G. Edwards, D. Popescu, S. Malek, and N. Medvidovic, “A Framework for Estimating the Energy Consumption Induced by a Distributed System’s Architectural Style,” in *Proceedings of the 8th Int. Workshop on Specification and Verification of Component-Based Systems*. ACM Press, p. 27, 2009.
[9] R. Randriatsiferana, R. Lorion, F. Alicalapa, and F. Harivelo, “Energy-efficient clustering algorithm based on energy variance for Wireless Sensor Networks,” in *Proceedings of 2013 International Conference on Smart Communications in Network Technologies* (SaCoNeT), vol. 01, pp. 1–5, 2013.
[10] N. Balbierer, T. Waas, J. Noebauer, and J. Seitz, “Energy Consumption of Ethernet compared to Automotive Bus Networks,” in *Proceedings of the 9th Workshop on Intelligent Solutions in Embedded Systems*, pp. 61–66, 2011.
[11] S. Gurun and C. Krintz, “A Run-Time, Feedback-Based Energy Estimation Model for Embedded Devices,” in *Proceedings of the 4th Int. Conference on Hardware/Software Codesign and System Synthesis*, pp. 28–33, 2006.
[12] R. B. Ortega and G. Borriello, “Communication synthesis for distributed embedded systems,” in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design* (ICCAD), pp. 437–444, 1998.
[13] A. Schranzhofer, J.-J. Chen, and L. Thiele, “Timing Analysis for TDMA Arbitration in Resource Sharing Systems,” in *Proceedings of the 16th IEEE Real-Time and Embedded Technology and Applications Symposium*, pp. 215–224, 2010.
[14] R. Wilhelm, J. Engblom, A. Ermedahl, N. Holsti, S. Thesing, D. Whalley, G. Bernat, C. Ferdinand, R. Heckmann, T. Mitra, F. Mueller, I. Puaut, P. Puschner, J. Staschulat, and P. Stenström, “The Worst-Case Execution Time Problem: Overview of Methods and Survey of Tools,” *ACM Transactions on Embedded Computing Systems* (TECS), vol. 7, no. 3, pp. 36:1–36:53, 2008.
[15] P. Marwedel, Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems, 2nd ed. Springer, 2011.
[16] A. Metzner, Effizienter Entwurf verteilter eingebetteter Echtzeit-Systeme: Dissertation, 2007.
[17] P. Heinrich and C. Prehofer, “Network-Wide Energy Optimization for Adaptive Embedded Systems,” *ACM SIGBED Review*, vol. 10, no. 1, pp. 33–36, 2013.
[18] A. Kern, Ethernet and IP for Automotive E/E-Architectures: Technology Analysis, Migration Concepts and Infrastructure: Dissertation, Erlangen, 2012.
[19] P. Heinrich, H. Bergler, and D. Eilers, “Energy Consumption Estimation of Software Components based on Program Flowcharts,” in *Proceedings of the 11th IEEE International Conference on Embedded Software and Systems*, 2014.
[20] P. Heinrich, H. Bergler, and E. Oswald, “Early Energy Estimation of Networked Embedded Systems Executing Concurrent Software Components,” *International Journal of Modeling and Optimization*, vol. 5, no. 2, pp. 119-127, 2015.