Technische Universität Braunschweig

Institute of Operating Systems and Computer Networks

Energy-Efficient Voltage Scheduling of Peripheral Components on Wireless Sensor Nodes

E2Nets 2014

Stephan Friedrichs, Ulf Kulau and Lars Wolf, June 14, 2014
Technische Universität Braunschweig, IBR

Energy Efficiency in WSNs – Motivation

- Energy Efficiency in WSNs / IoT plays a significant role
  - Usability, feasibility, acceptance...
- Limping evolution of batteries (capacity)
- Various existing approaches on several layers

Energy Efficiency in WSNs – Motivation

- Energy Efficiency in WSNs / IoT plays a significant role
  - Usability, feasibility, acceptance...
- Limping evolution of batteries (capacity)
- Various existing approaches on several layers

Application Layer	Data Management, Compression, ...
Network Layer	Routing (RPL), ...
Data Link Layer	TDMA, LPL (X-MAC), ...
Physical Layer	DPM, PVS, DVS, ...

Starting Point – Peripherals

Existing work focuses on the transceiver and the processing unit:

- Peripheral energy consumption may dominate
  - Transceiver (16 mA) vs. GPS(44 mA)
  - Various multi-sensing applications (many peripherals)

Starting Point – Peripherals

Existing work focuses on the transceiver and the processing unit:

- Peripheral energy consumption may dominate
  - Transceiver (16 mA) vs. GPS(44 mA)
  - Various multi-sensing applications (many peripherals)

Basic Idea:

- Reducing overall energy consumption by adding DVS to peripherals

Voltage Level and Energy Consumption – Basics

Background:

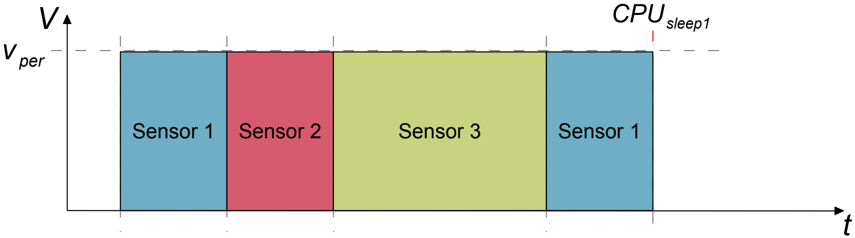
- Energy consumption of peripherals depends on voltage level
- Each peripheral requires a different minimum voltage level

Voltage Level and Energy Consumption – Basics

Background:

- Energy consumption of peripherals depends on voltage level
- Each peripheral requires a different minimum voltage level

Classical approach:

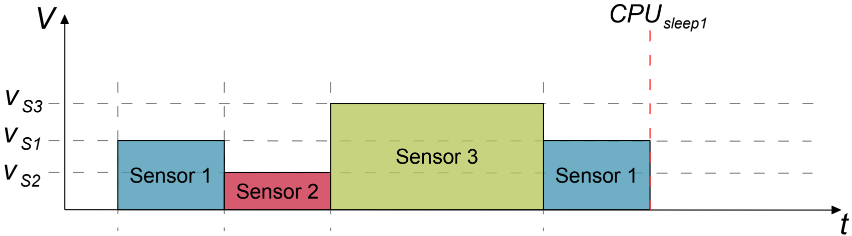


Voltage Level and Energy Consumption – Basics

Background:

- Energy consumption of peripherals depends on voltage level
- Each peripheral requires a different minimum voltage level

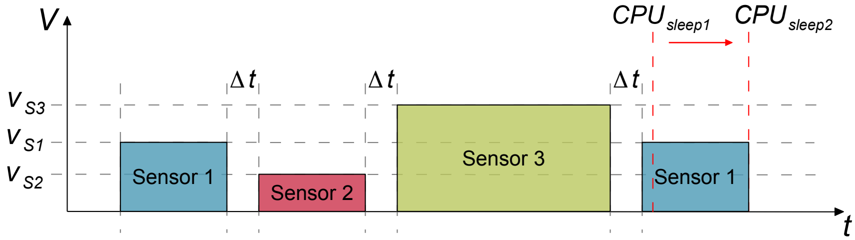
Our approach: Dynamic adaptation of the peripherals voltage



Individual Voltage Adaptation – Tradeoff

Dynamic adjustment of the peripherals voltage, but how?

- Additional overhead: Switching the voltage consumes energy
  - Timing overhead (Δt) leads to higher CPU duty cycles
  - Static power dissipation of a voltage scaling module

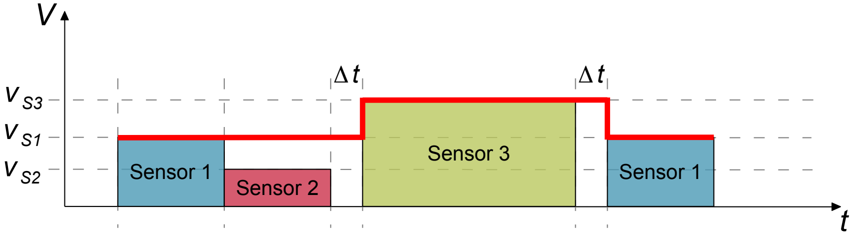


Individual Voltage Adaptation – Tradeoff

Dynamic adjustment of the peripherals voltage, but how?

- Additional overhead: Switching the voltage consumes energy
  - Timing overhead (Δt) leads to higher CPU duty cycles
  - Static power dissipation of a voltage scaling module

Calculation of an optimal voltage schedule for arbitrary sequences



Voltage Scheduling Problem – Definitions

- Consider a sensor node with a set S of peripheral hardware
- Energy consumption of a peripheral $s \in S$

$$e_s(v) = v \int_0^{t_s} I_s(v, t), dt$$

$I_s(v, t)$ flowing through s as well as through the inactive peripheral hardware $S \setminus \{s\}$

- Constant amount C of energy – the switching overhead
- Given sequence of queries $[1, \ldots, n]$
  - Query i operates the peripheral device $s_i \in S$
  - s_i requires minimum voltage level $v_{min}(s_i)$
- Searched for optimal voltage schedule $v(1), \ldots, v(n)$

Voltage Scheduling Problem – Optimal Solution

- The energy consumption E of a voltage schedule is:

$$E = \sum_{i=1}^{n} e_{s_i}(v(i)) + \sum_{i=2}^{n} \begin{cases} C & \text{if } v(i-1) \neq v(i), \\ 0 & \text{otherwise.} \end{cases}$$

- Result: Voltage schedule is optimal if E is minimal

Voltage Scheduling Problem – Optimal Solution

- The energy consumption E of a voltage schedule is:

$$E = \sum_{i=1}^{n} e_{s_i}(v(i)) + \sum_{i=2}^{n} \begin{cases} C & \text{if } v(i-1) \neq v(i), \\ 0 & \text{otherwise.} \end{cases}$$

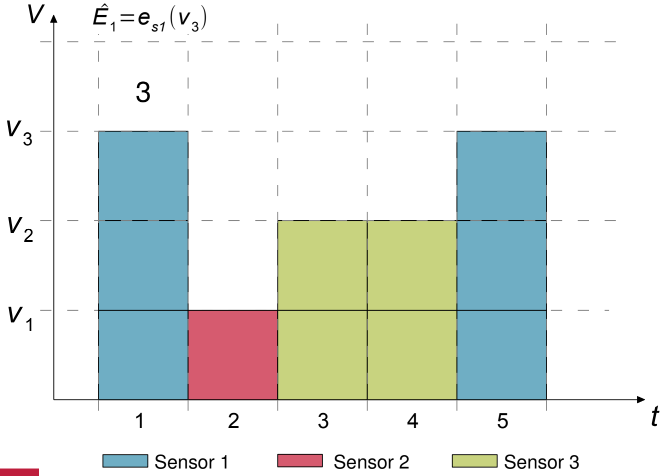
- Result: Voltage schedule is optimal if E is minimal

Solution:

- Optimal voltage schedule would only use $v(i) \in \{v_{min}(s) \mid s \in S\}$
- Determining an optimal schedule by *dynamic programming*

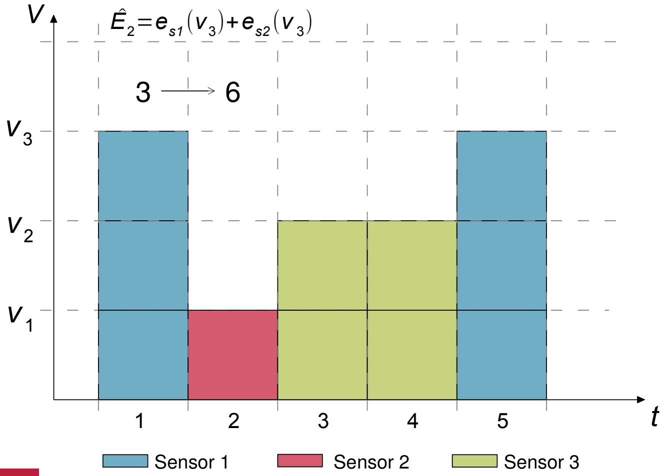
Example – Optimal Voltage Schedule

Simplification: $C = 1$, $v_i = i$ and $e_s i = v_i$,



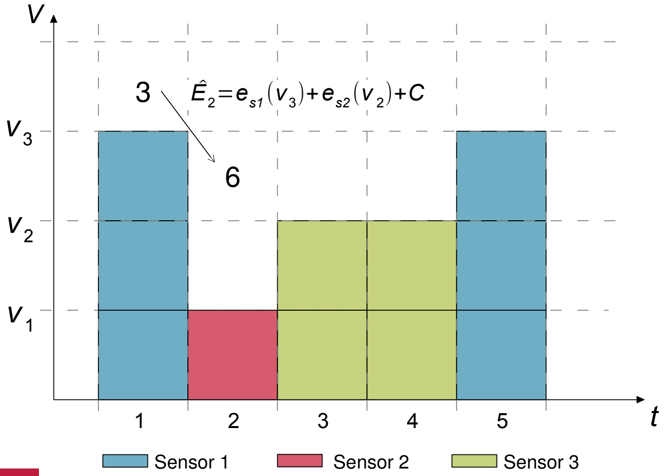
Example – Optimal Voltage Schedule

Simplification: $C = 1$, $v_i = i$ and $e_s i = v_i$,



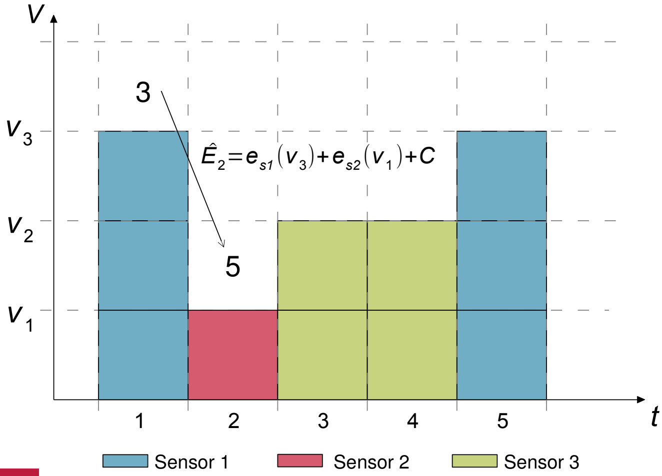
Example – Optimal Voltage Schedule

Simplification: $C = 1$, $v_i = i$ and $e_s i = v_i$,



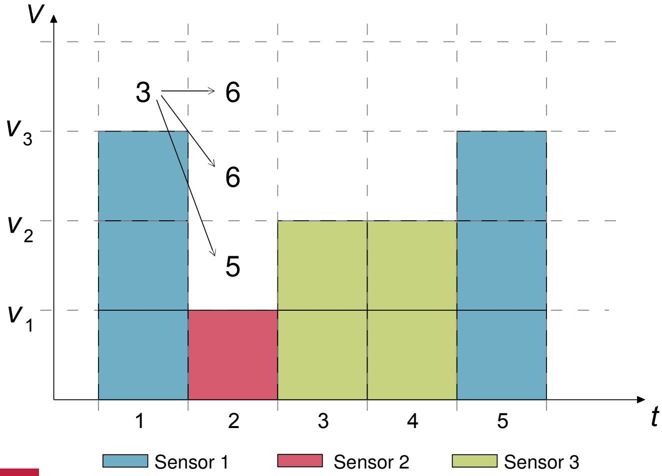
Example – Optimal Voltage Schedule

Simplification: $C = 1$, $v_i = i$ and $e_s i = v_i$,



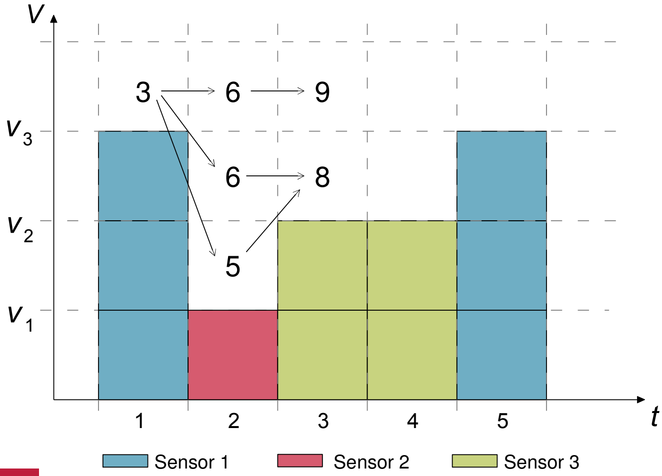
Example – Optimal Voltage Schedule

Simplification: $C = 1$, $v_i = i$ and $e_si = v_i$,



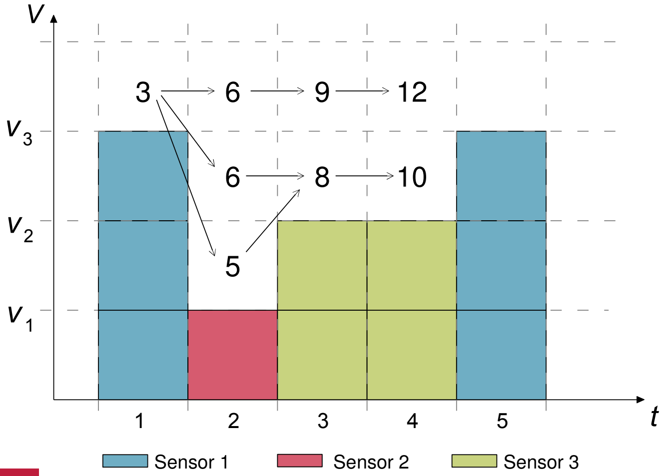
Example – Optimal Voltage Schedule

Simplification: $C = 1$, $v_i = i$ and $e_s i = v_i$,



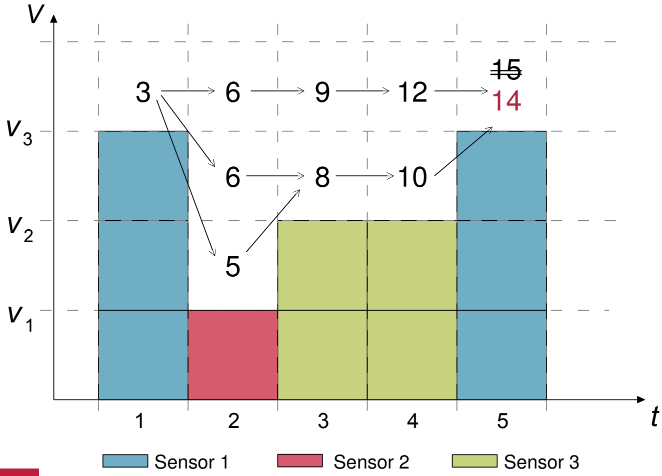
Example – Optimal Voltage Schedule

Simplification: $C = 1$, $v_i = i$ and $e_s i = v_i$,



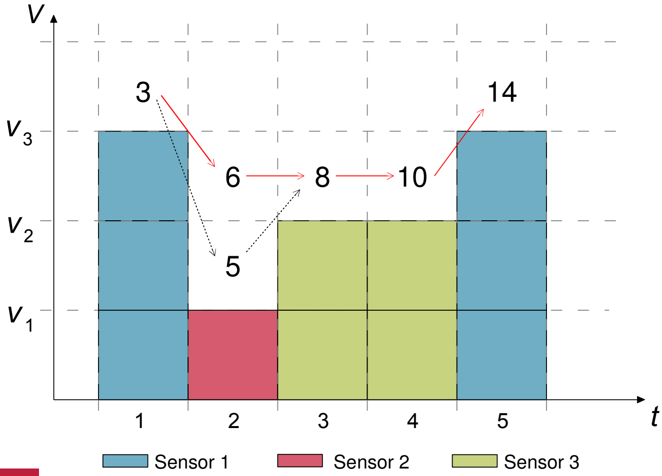
Example – Optimal Voltage Schedule

Simplification: $C = 1$, $v_i = i$ and $e_s i = v_i$,



Example – Optimal Voltage Schedule

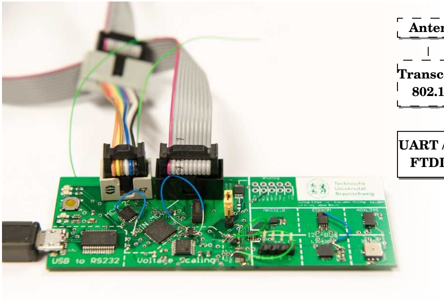
Simplification: $C = 1$, $v_i = i$ and $e_s i = v_i$,

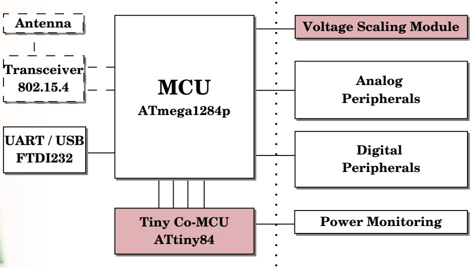


Theory and Practice – Prototype Implementation

Implementation of a HW and SW prototype

- Equipped with common peripherals for WSN applications
- Evaluation of the effect of Voltage Scheduling





Self-Parametrization $e_s(v)$

Prototype Peripherals:

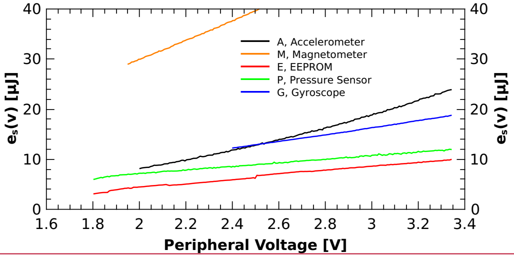
Peripheral s	Device	Description	$v_{min}(s)$ [V]
A	ADXL345	Accelerometer	2.000
E	AT24C08C	EEPROM	1.800
P	BMP085	Pressure Sensor	1.800
G	L3G4200D	Gyroscope	2.400
M	MAG3110	Magnetometer	1.950
M_b	MAG3110	Magnetometer (high accuracy)	1.950

Self-Parametrization $e_s(v)$

Prototype Peripherals:

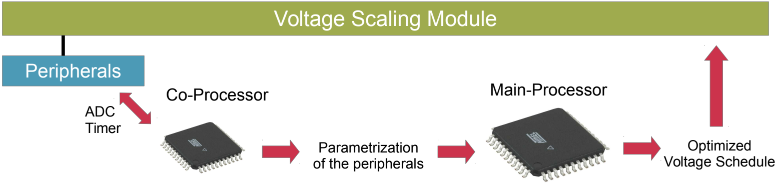
Peripheral s	Device	Description	$v_{min}(s)$ [V]
A	ADXL345	Accelerometer	2.000
E	AT24C08C	EEPROM	1.800
P	BMP085	Pressure Sensor	1.800
G	L3G4200D	Gyroscope	2.400
M	MAG3110	Magnetometer	1.950
M_b	MAG3110	Magnetometer (high accuracy)	1.950

- How to get information about the energy consumption?
- Modelling of $e_s(v)$ might be arbitrary complicated



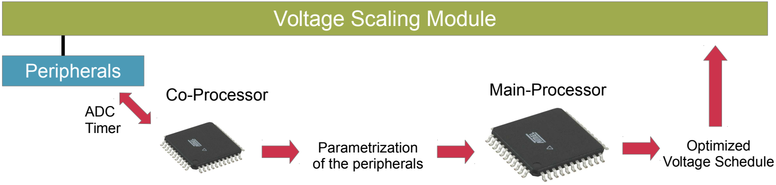
Self-Parametrization $e(s)$

- Fully self-optimizing approach
  - Parametrization ($e_s(v)$) is done by second micro controller
  - Resource-friendly calculation (no expensive floating-point needed)



Self-Parametrization $e(s)$

- Fully self-optimizing approach
  - Parametrization ($e_s(v)$) is done by second micro controller
  - Resource-friendly calculation (no expensive floating-point needed)



Monotonicity: Schedule only uses minimum voltage levels

Reference Voltage		Energy Consumption $e_s(v_{min}(s'))$ [μJ] per Query				
s'	$v_{min}(s')$ [V]	$s = E$	$s = P$	$s = M$	$s = A$	$s = G$
E, P	$V_1 = 1.800$	4.943	13.140	—	—	—
M	$V_2 = 1.950$	6.106	15.072	282.827	—	—
A	$V_3 = 2.000$	6.363	15.466	292.033	18.962	—
G	$V_4 = 2.400$	8.019	18.915	365.285	25.585	14.391

Evaluation – Results

Test sequences are compared to three trivial strategies

		Energy saved by Scheduled compared to		
#	Query Sequence	ConstDefault	ConstMaxMin	AlwaysSwitch
1	AEPGMAEPGM	45.80 %	17.13 %	0.97 %
2	GAMGAMGAM	46.15 %	17.04 %	0.49 %
3	GAMPE	46.91 %	18.52 %	1.40 %
4	GAMPEGAMPE	46.26 %	17.50 %	1.79 %
5	GEPGEPGEP	33.33 %	0.81 %	10.29 %
6	PEMAG	46.91 %	18.52 %	1.68 %
7	PEMAGPEMAG	46.39 %	17.59 %	1.79 %
8	GAPEGAPE	37.80 %	2.26 %	8.45 %
9	GPGPGPGPGP	31.54 %	0.00 %	20.29 %
10	PAMPE	47.90 %	20.41 %	2.53 %
11	APEGAME	46.53 %	17.20 %	3.27 %

Note: A = Accelerometer, E = EEPROM, P = Pressure Sensor, G = Gyroscope, M = Magnetometer

- Scheduled voltage level saves energy
- Scheduling never uses more energy than other strategies (optimal)

Summary

Peripherals are less considered:

- Energy consumption of peripherals may dominate
- Energy consumption depends on applied voltage level

Summary

Peripherals are less considered:

- Energy consumption of peripherals may dominate
- Energy consumption depends on applied voltage level

Many peripherals with different minimum voltage levels

- Trade-off between decreased voltages and switching overhead
- Calculation of optimal schedule by dynamic programming

Summary

Peripherals are less considered:

- Energy consumption of peripherals may dominate
- Energy consumption depends on applied voltage level

Many peripherals with different minimum voltage levels

- Trade-off between decreased voltages and switching overhead
- Calculation of optimal schedule by dynamic programming

Prototype implementation and evaluation

- Fully self-optimizing approach through second MCU
- Scheduling is always optimal
- Saves up to 47.9% compared to related solutions

Summary

Peripherals are less considered:

- Energy consumption of peripherals may dominate
- Energy consumption depends on applied voltage level

Many peripherals with different minimum voltage levels

- Trade-off between decreased voltages and switching overhead
- Calculation of optimal schedule by dynamic programming

Prototype implementation and evaluation

- Fully self-optimizing approach through second MCU
- Scheduling is always optimal
- Saves up to 47.9% compared to related solutions

Thank you for your attention! Questions?

Ulf Kulau
kulau@ibr.cs.tu-bs.de

Appendix – Quality of Sensing