GECKO RESEARCH

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Design, Modeling, and Optimization of Power Electronics Systems
Virtual Prototyping

Andrija Stupar, Andreas Müsing

ECPE
Bayern Innovativ
Kooperationsforum mit Fachausstellung
Leistungselektronik
Hilton Hotel, Nürnberg, 27.10.2011.
BAIKEM
Cluster Leistungselektronik

Outline

- State of "Virtual Prototyping" today: Problems
- Solution: PE Design Suite with GeckoCIRCUITS as the core
- Comparison via design example:
  - Analytical approach to converter design and optimization
  - Simulation approach and its advantages
- Modeling Different Design Domains: Electrical, Magnetic, Thermal: Modeling Everything as a Circuit?
- Coupling Domains: Model Reduction and Simplification

Motivation

- Power Electronics Engineer must consider many factors when making design decisions:

  - System performance & Efficiency

  - Power Density (Volume, size) & Weight

  - Cost, Reliability, etc.

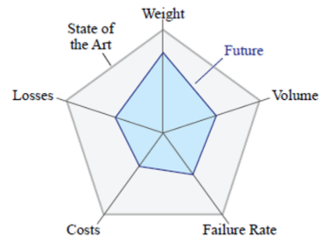
- Must deal with Thermal & Electromagnetic issues

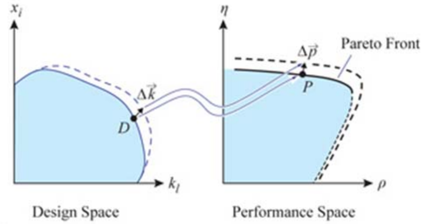
- Many choices to make:

  - Topology?

    - Control/modulation scheme?

      - Components?





Need **<u>Virtual Prototyping</u>**: evaluate on a computer, relatively quickly, a large number of design possibilities, and gain insight into relationships between the different aspects of the design problem.

State of "Virtual Prototyping" Today

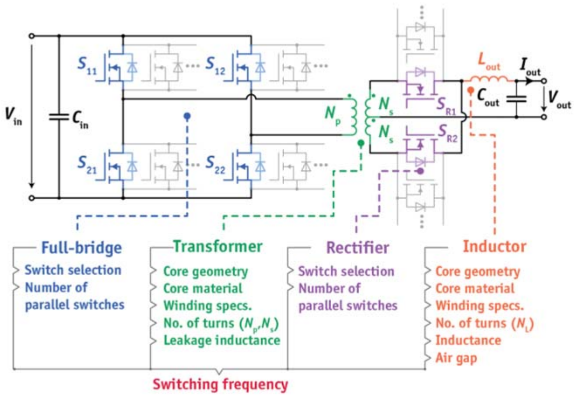
- Generally speaking, the theory to do virtual prototyping already exists
- It seems that we have software tools for almost all necessary domains:

  - Very detailed and precise circuit simulators (e.g. SPICE, etc.)
  - Very powerful electromagnetic simulators (e.g. Maxwell)
  - 3D-FEM simulators for thermal design (e.g. Icepak, COMSOL)
- We have a large body of knowledge on the behaviour of power electronics (PE) and the necessary sub-components
- **So what is the problem?**
  - Tedious: it takes very long to set up all relevant models
  - Tools not made specifically for PE: large skill set needed
  - Detailed simulation slow; not easy to transfer relevant data
  - **Result:** Engineer concludes not worth the effort, does limited simulation and calculations, relies on past designs, experience and actual prototyping
- **Solution:** Create a software package that has relevant models and simulators, is fast, and "fits well" with the knowledge of PE engineers

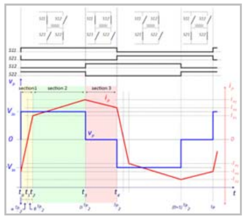
Optimization Example: Analytical Approach

Phase-shift PWM DC-DC Converter for Telecom Power Supplies (5 kW)

Papers: Badstuebner, Biela, and Kolar, APEC 2010 and IPEC 2010



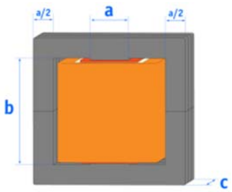
Derive steady-state Operating point



- Formulae for RMS, average currents, voltages
- FFT for AC losses

(check by simulation)

Set up loss models



$$P_{proxP} = \sum_{n=1}^{n_h}\sum_{m=1}^{N_P} \frac{b_{foil}\cdot\rho}{\sqrt{\frac{\rho}{\pi\sigma f_{sw}\mu_0}}}\cdot\frac{\sinh\nu_P+\sin\nu_P}{\cosh\nu_P-\cos\nu_P}\cdots$$

$$\cdot\left(\frac{1}{2\cdot b}\cdot I_{P,h,n}\cdot(2\cdot m-1)\right)^2\cdot l_{w,P}$$

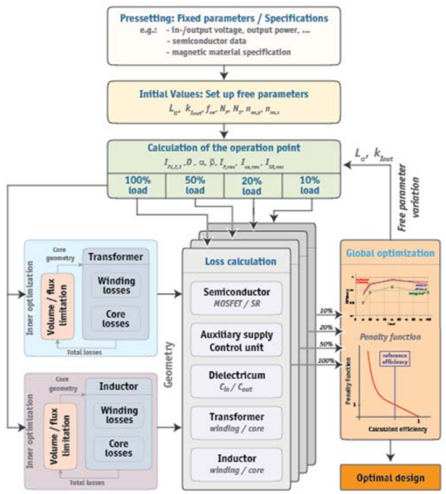
$$P_{skin,P} = R_{DC,P}\cdot(I_{P,h,0})^2+\ldots$$

$$+\sum_{n=1}^{n_h} R_{DC,P}\cdot\frac{\nu_P}{4}\cdot\frac{\sinh\nu_P+\sin\nu_P}{\cosh\nu_P-\cos\nu_P}\cdot(I_{P,h,n})^2$$

$$P_{sw} = 2\cdot n_{sw,P}\cdot E(V_{res})\cdot f_{sw}$$

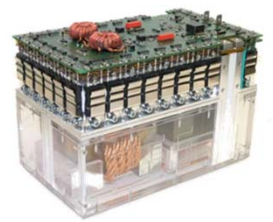
$$P_{cond} = \frac{R_{DS,on}\cdot I_{sw,rms}^2}{n_{sw,p}}$$

Optimization goal:
Maximum **Efficiency** (99%)



Optimization procedure

Built prototype:



Optimal design

Calc. eff.: 98.9%
Meas. eff: 98.5%

Java program/Maple script

Optimization Example: Analytical Approach

- How long does this take, start to finish? (not incl. prototype construction)
  - Derive and setup all models: 2-4 months
  - Execute optimization procedure: 1-2 weeks
- Great deal of effort required
- Want to try different topology? Start again, from beginning
- Change operating mode? Start again
- Change control/modulation scheme? Start again
- Error in deriving analytical models? Start again
- Change of components, geometries? New loss models needed

- **The need for a better, more general approach is clear**

Optimization by Simulation: Requirements

- Replace as much as possible analytical work by numerical simulation:

Build model in PE-engineer-friendly software environment

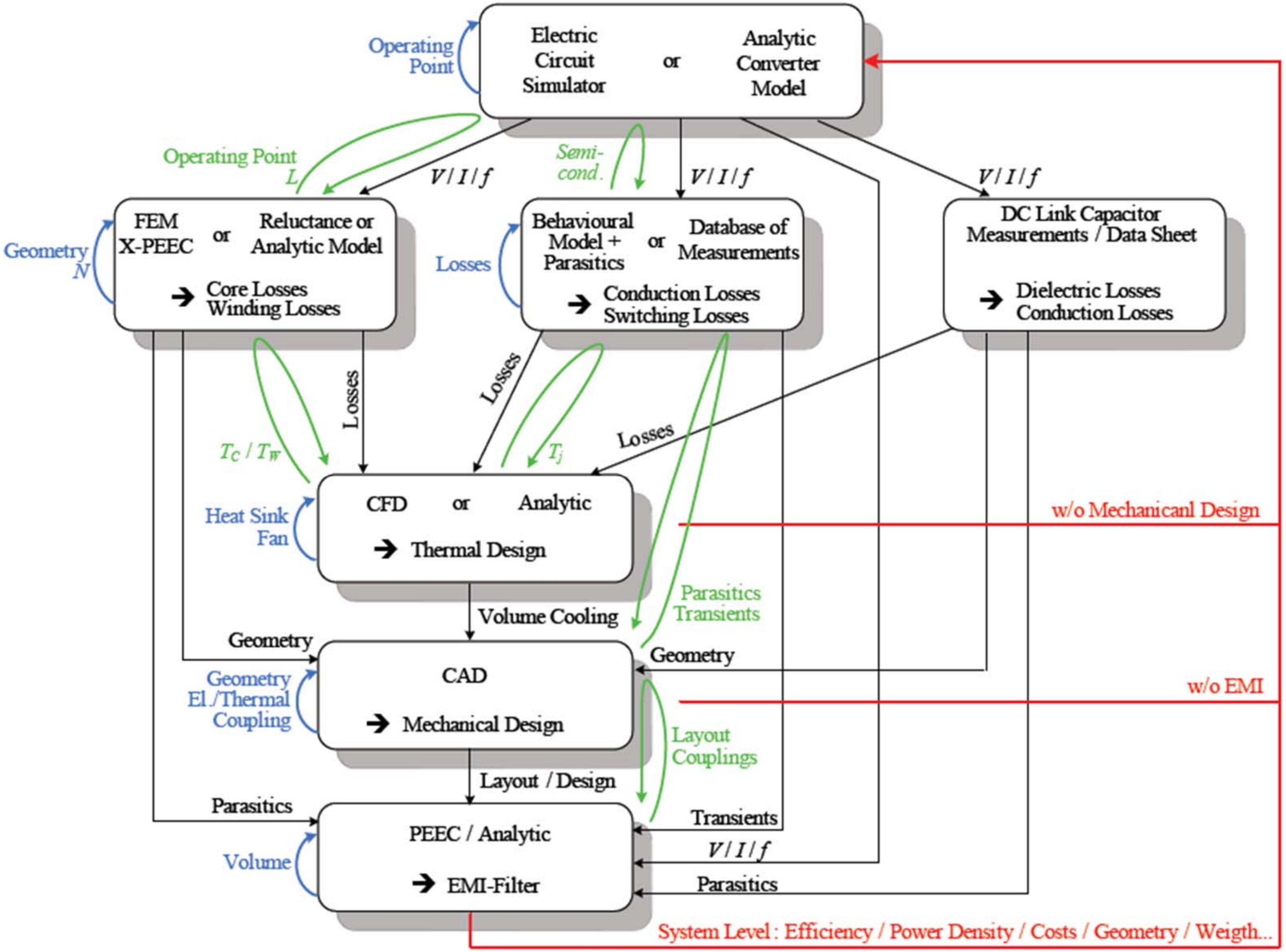
⬇

Do minimum amount of simulation necessary

⬇

Extract automatically from simulation results all required parameters for system evaluation

Coupling of Physical Domains



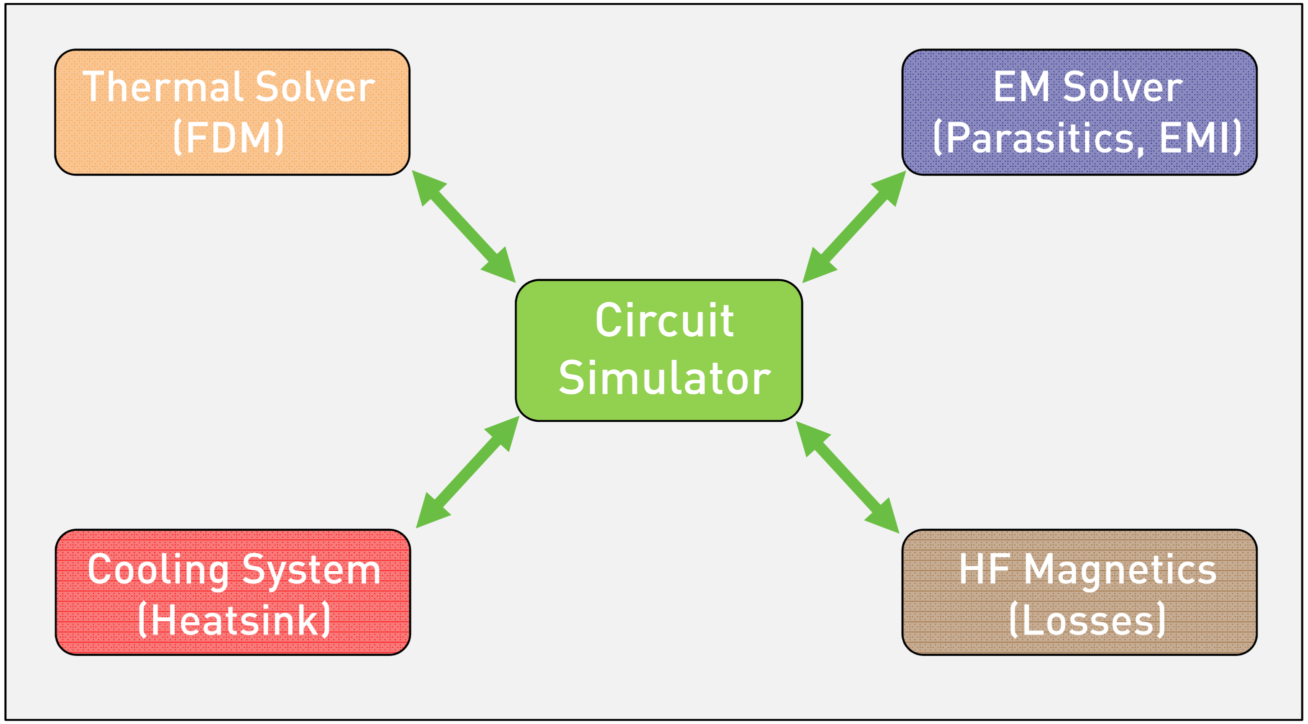
Is this a realistic approach for a PE Design?

GECKO RESEARCH

Multi-Domain Simulation in Power Electronics

- PE Engineer challenged with different domains
- Circuit Simulator should be „central part“ of design toolbox
- Direct tool interconnection not realistic

→ Consider different abstraction levels (model order reduction)

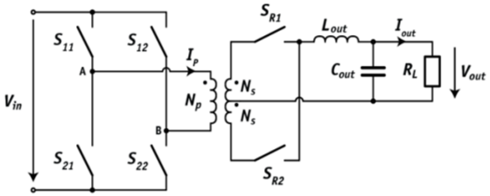


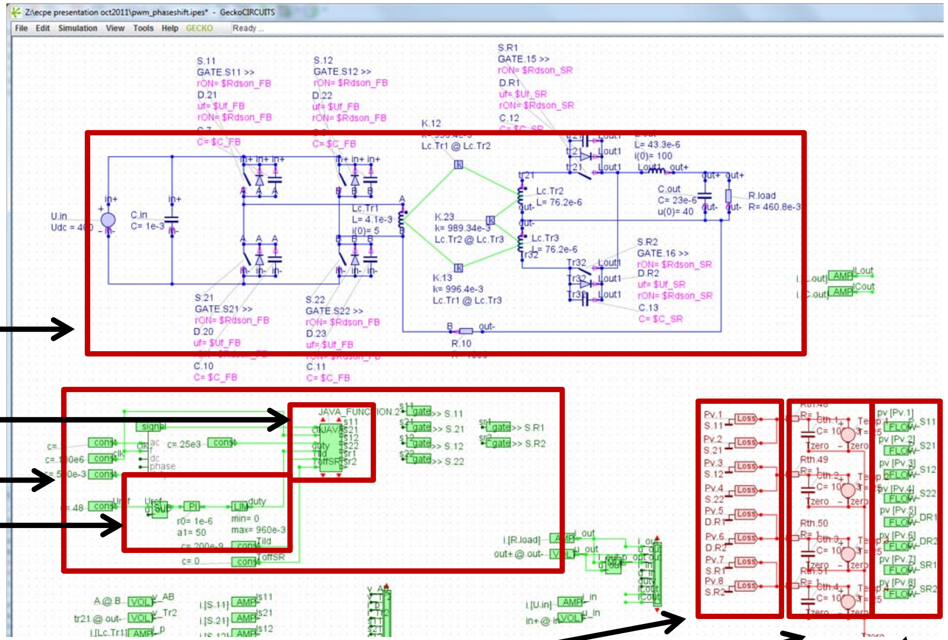
Circuit interpretation possible?

- Power Circuit
- Electromagnetics
- Thermal
- Magnetics

PE Circuit Simulator: GeckoCIRCUITS

- Model of converter for simulation





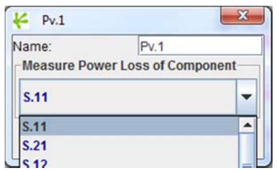
Circuit model

Java block simulates any control/modulation scheme

Control model

PI control

Calculate loss of semiconductors

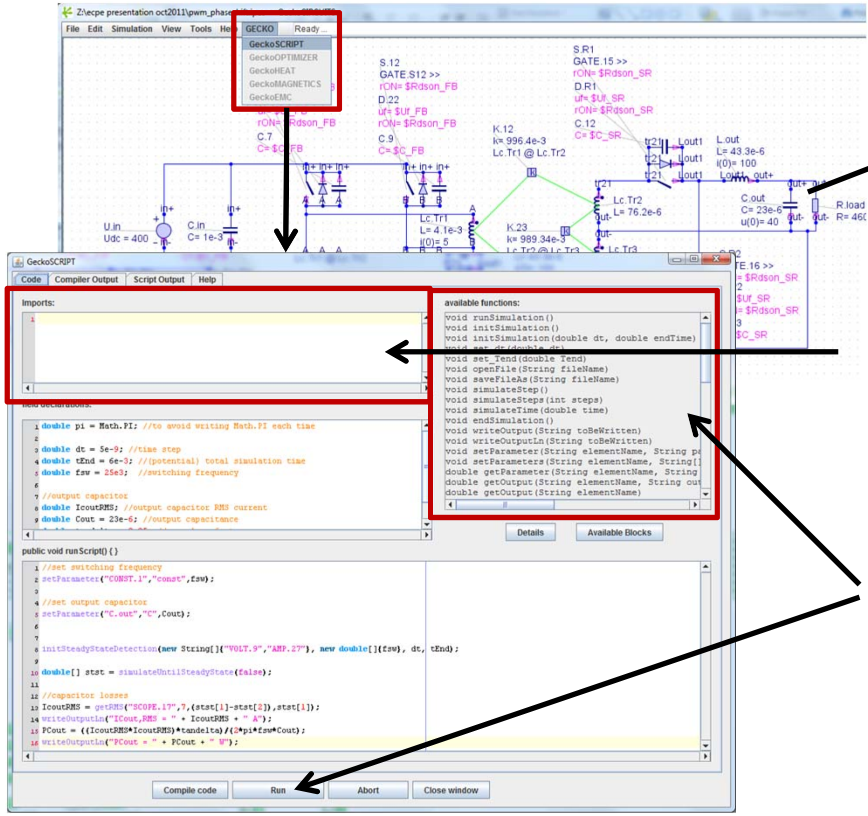


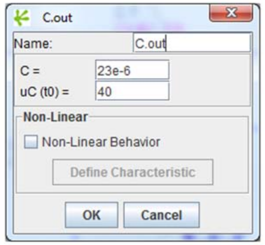
Thermal RC circuit model of semiconductor + heat sink

Send temperature waveforms to scope

Setting Model Parameters in GeckoCIRCUITS

- For virtual prototyping and optimization, must be able to simulate, change system parameters, simulate again, change parameters, simulate…





Shouldn't do this manually every time

Full Java API available, can utilise full power of Java programming language

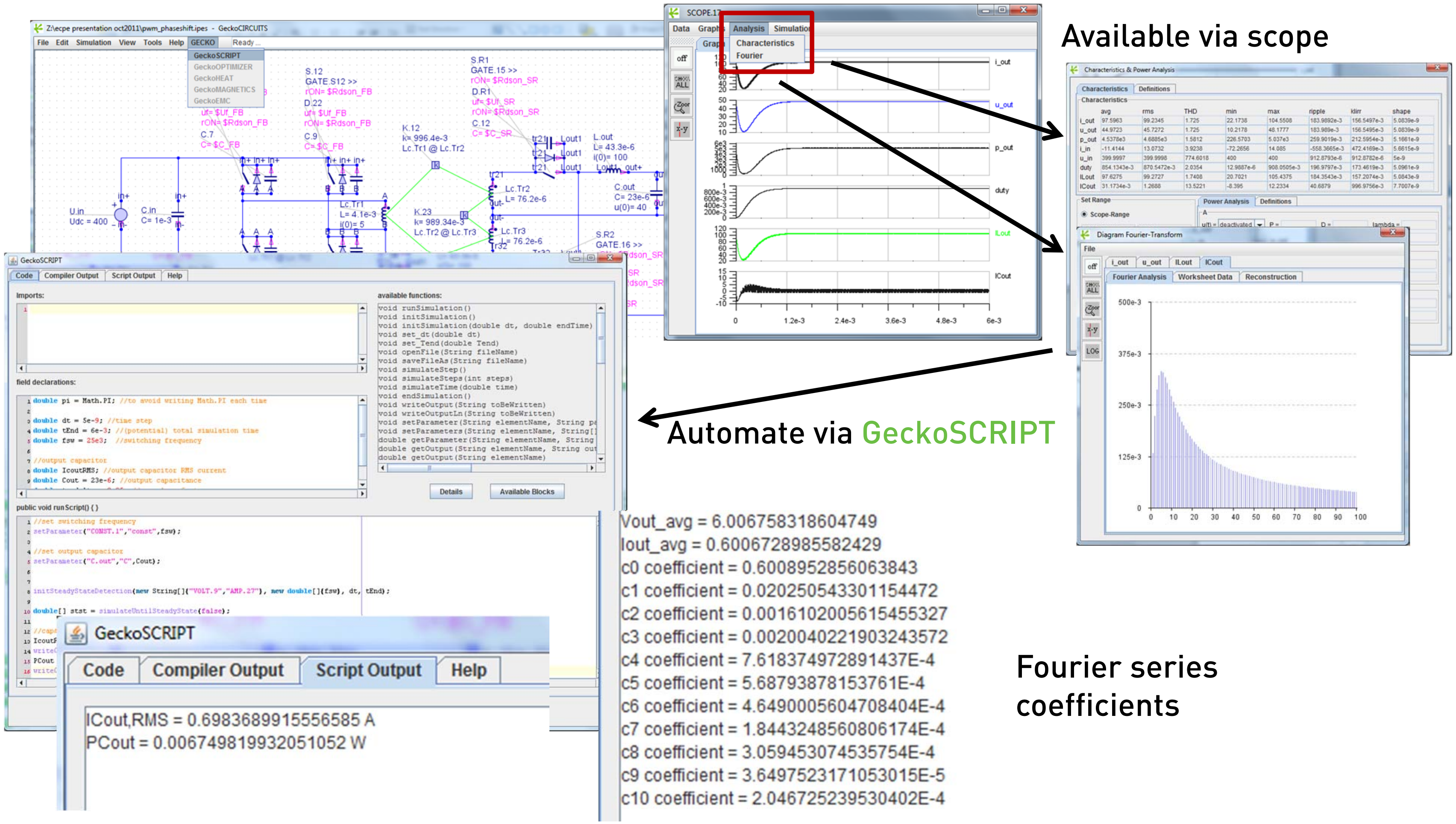
Functions to set all model parameters, control simulation, simulate step-by-step, or by time interval

GeckoSCRIPT: model manipulation and simulation control scripting environment within GeckoCIRCUITS

Tutorial for GeckoSCRIPT available on GeckoCIRCUITS CD

Extract relevant information from simulation

- Need: RMS, avg, min/max values of currents, voltages, FFT of signals…



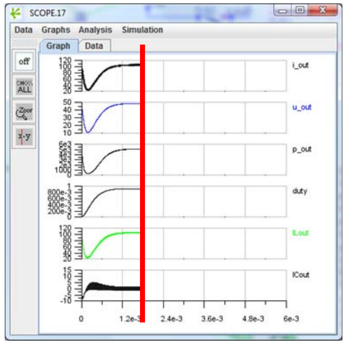
RMS, etc. values

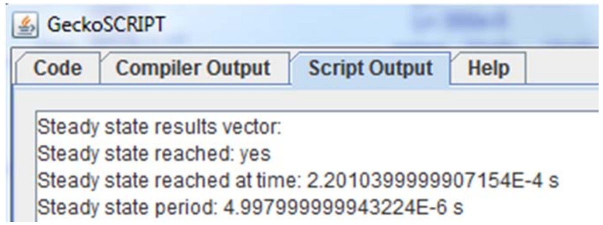
GeckoCIRCUITS: Steady-State Detection

- Usually interested what happens during **steady-state operation**
- GeckoSCRIPT provides functions for periodic steady-state operation: simulate until steady-state and stop, then extract parameters

```
initSteadyStateDetection(new String[]{"VOLT.9","AMP.27"}, new double[]{fsw}, dt, tEnd);

double[] stst = simulateUntilSteadyState(false);
```





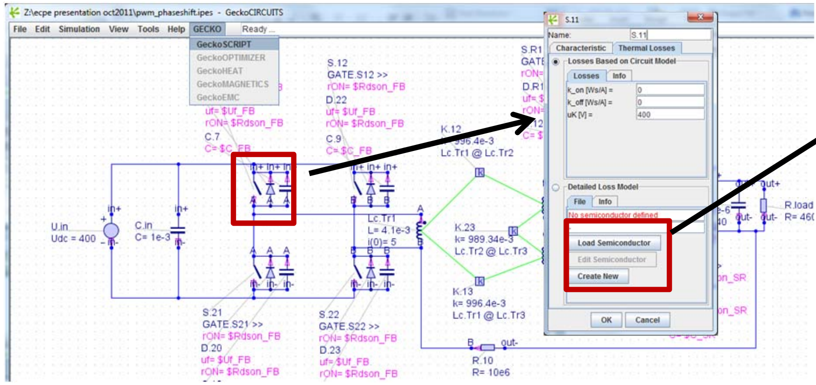
Stops when steady-state reached

Currently (v.1.5) works for PWM DC-DC systems
- Development ongoing to cover other types of systems

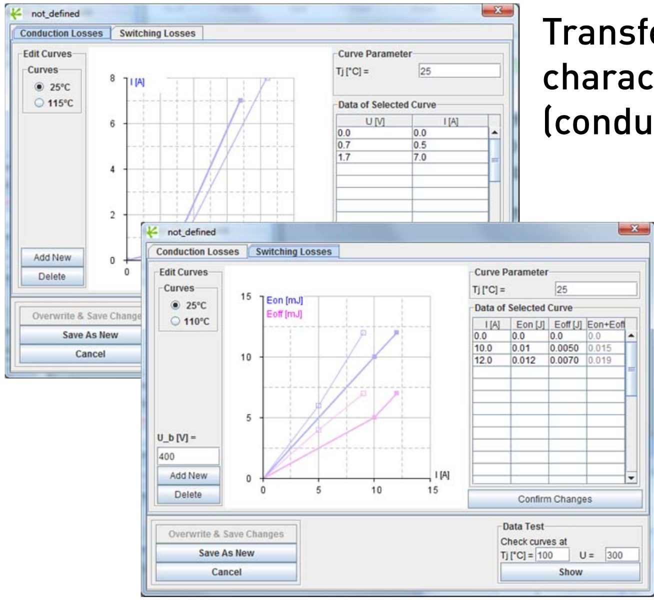
All analytical analysis of power converter circuit has been replaced by simulation!

Loss Modeling: Semiconductors

- Rather than simulate semiconductors in great detail to extract all losses from parasitics, etc. (too slow), have functionally correct model for PE circuits for fast simulation
- Use electrical simulation results to calculate losses based on loss models - > data entered from data sheet curves or experimental measurements



“Real-time” loss and temperature curves produced by simulation



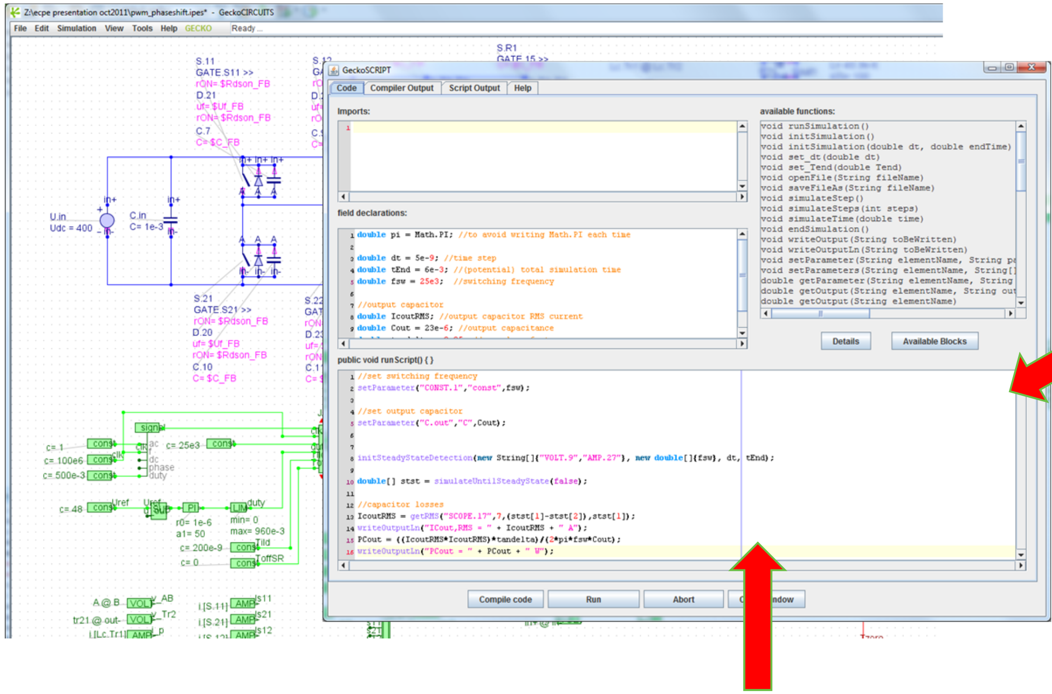
Transfer characteristic (conduction losses)

Turn-on and turn-off energies (switching losses)

GECKO RESEARCH

Loss Modeling: Passives

- Current GeckoCIRCUITS version (1.5): still must work-out and enter loss models for inductors, transformers, capacitors "by hand" (standard models available in literature for most common arrangements)



Enter loss model formulae for passive components here:

$$P_{proxP} = \sum_{n=1}^{n_h} \sum_{m=1}^{N_P} \frac{b_{foil} \cdot \rho}{\sqrt{\frac{\rho}{\pi n f_{sw} \mu_0}}} \cdot \frac{\sinh \nu_P + \sin \nu_P}{\cosh \nu_P - \cos \nu_P} \ldots$$
$$\cdot \left(\frac{1}{2 \cdot b} \cdot I_{P,h,n} \cdot (2 \cdot m - 1) \right)^2 \cdot l_{w,P}$$

$$P_{skin,P} = R_{DC,P} \cdot (I_{P,h,0})^2 + \ldots$$
$$+ \sum_{n=1}^{n_h} R_{DC,P} \cdot \frac{\nu_P}{4} \cdot \frac{\sinh \nu_P + \sin \nu_P}{\cosh \nu_P - \cos \nu_P} \cdot (I_{P,h,n})^2$$

$$P_{cond} = \frac{R_{DS,on} \cdot I_{sw,rms}^2}{n_{sw,p}}$$

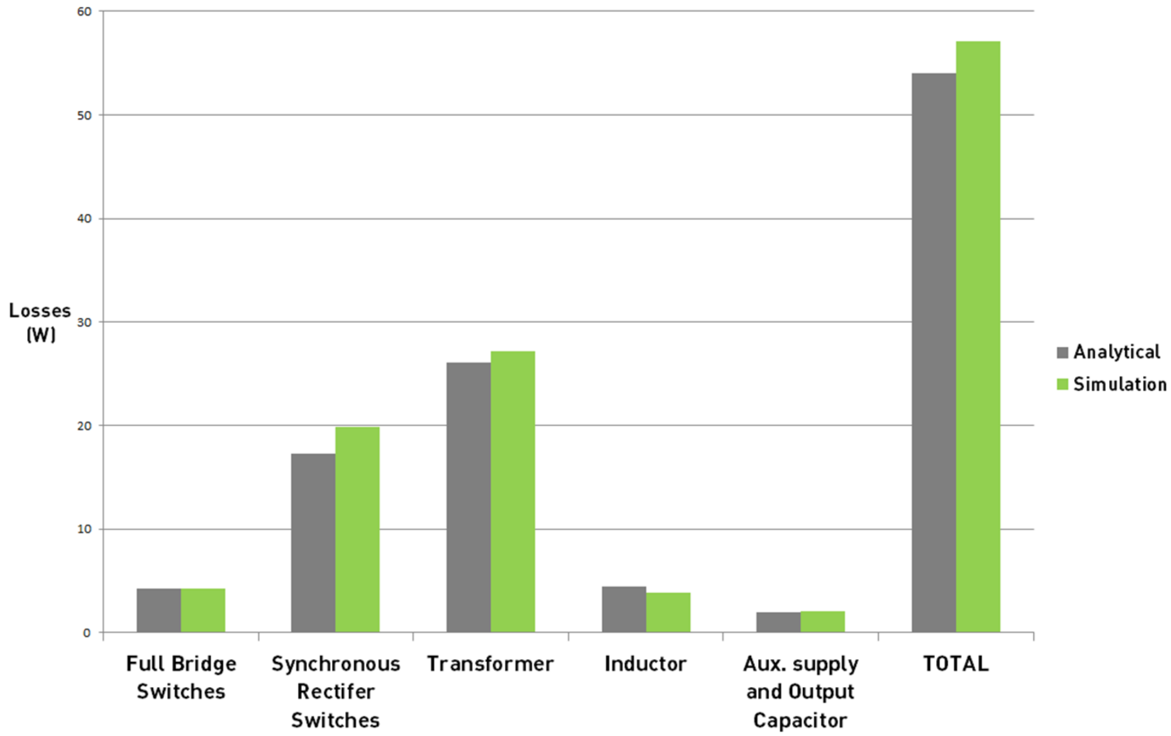
"Plug-in" extracted data (RMS, avg., FFT)

Code optimization loop here

GECKO RESEARCH

Comparison: Analytic vs. Simulation

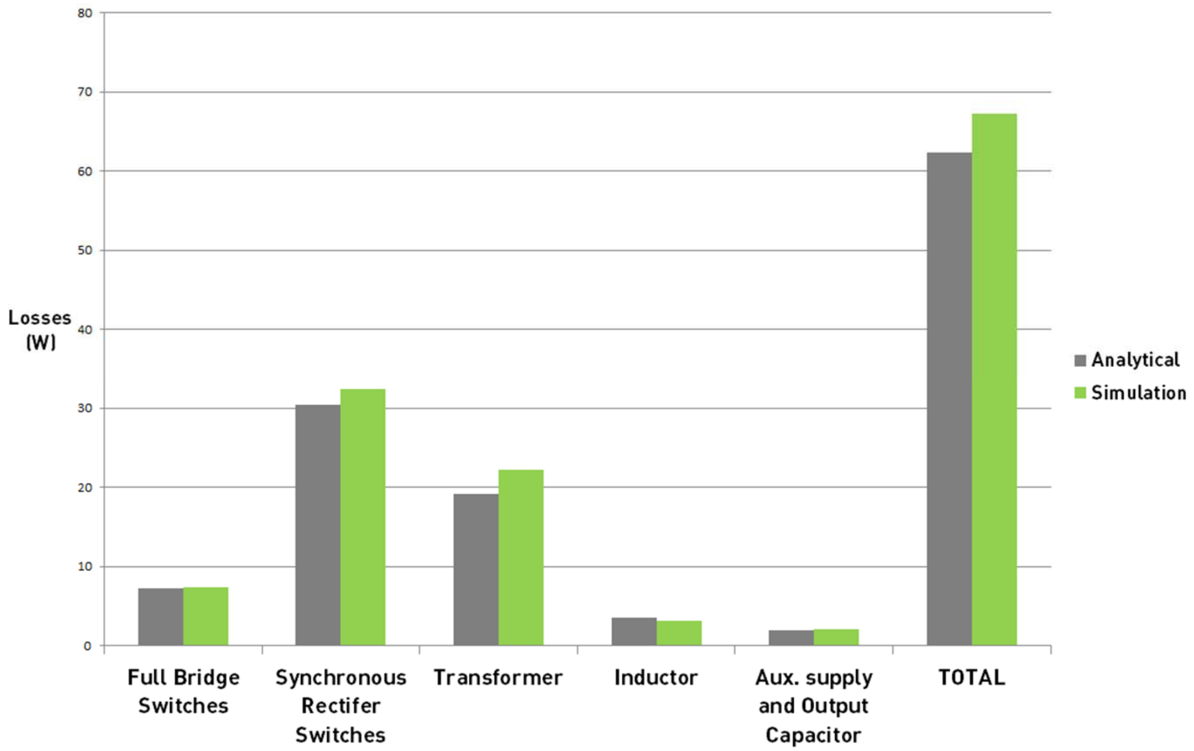
- Optimum system, switching frequency 16 kHz



Efficiency:
- Analytical calculations: 98.9%
- Derived from simulation: 98.8%

Comparison: Analytic vs. Simulation

- Possible converter design, switching frequency 50 kHz

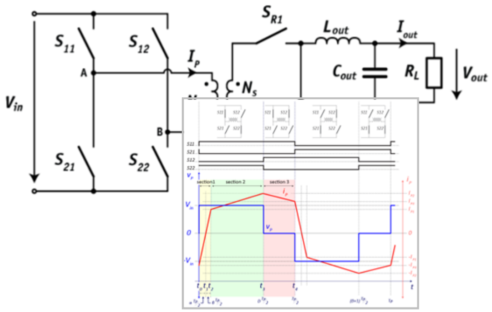


Efficiency:

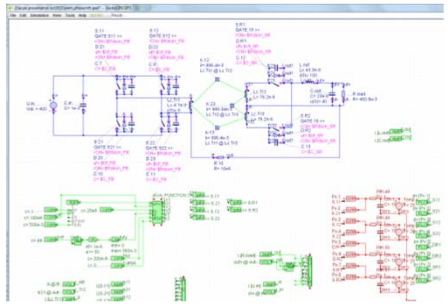
- Analytical calculations: 98.7%
- Derived from simulation: 98.6%

Comparison: Analytic vs. Simulation

Analytical



Simulation



Calculate one operating point: ~1 s

Set-up model: month(s)

Non-linearities: difficult (e.g. C_{oss})

Model adaptability: low to none, difficult

Calculate one operating point: 8 s (slower)
- to be much improved in the future!

Set-up model: days – 2 weeks (much faster)

Non-linearities: easy

Model adaptability: high and simple

Results: match well

Future Development of GeckoCIRCUITS (Version 2.0)

- Variable / adaptive simulation step-width √
- **Fast direct steady state calculation** √
- Reluctance models for transformers / magnetic circuits √
- Magnetics losses calculation √
- More detailed switch models (MOSFETS, bipolar transistors, ...)
- Built-in **optimization algorithms**
- **Connection of GeckoCIRCUITS to 3D field solvers:**
  - **GeckoEMC:** calculation of layout parasitics √
  - **GeckoHEAT:** 3D finite element thermal simulation √

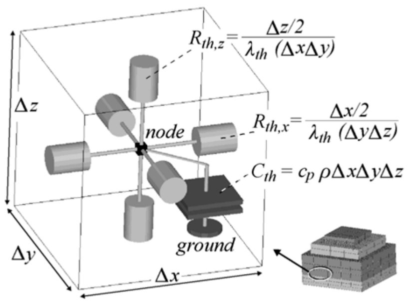
Further increases calculation speed → Optimization!

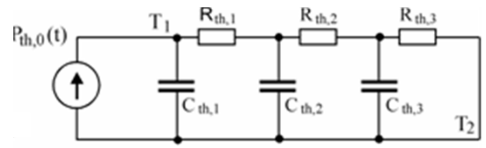
Version 2.0 Release: June 2012

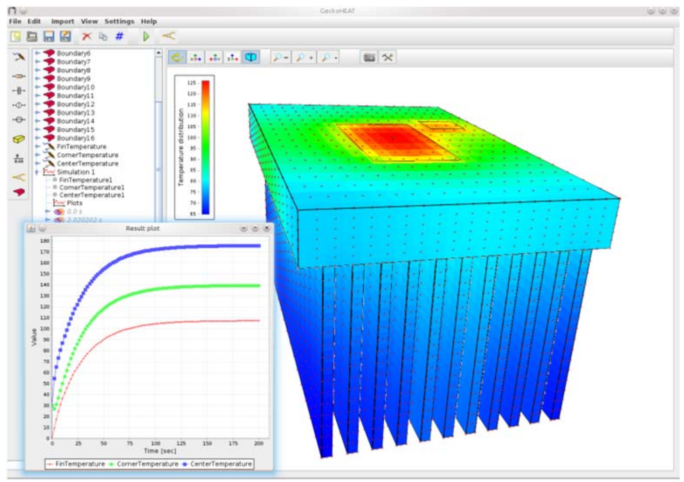
GECKO RESEARCH

Thermal Modeling & Simulation: GeckoHEAT

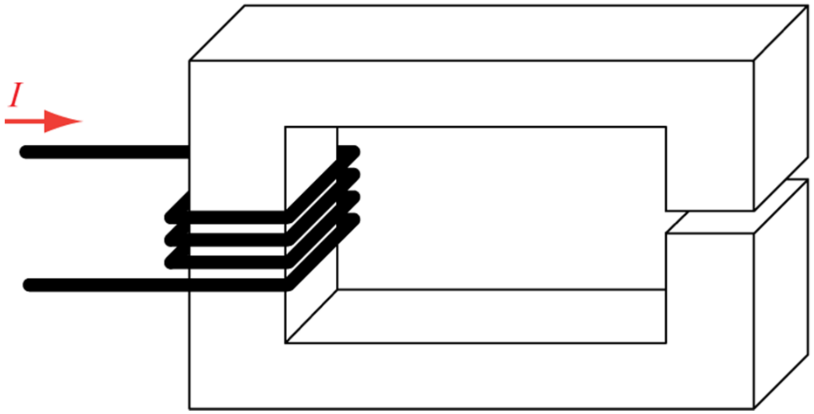
- Standard approach to thermal simulation: 3D-FEM simulation when necessary: slow and cumbersome
- GeckoHEAT: Finite-difference method (FDM) based approach to thermal modeling and simulation: thermal RC (impedance) circuits
- Easy-to-use, very fast
- Various boundary-conditions
  - Power loss density
  - Convection boundary
  - Fixed temperature
- Automatic extraction of thermal impedance network
- Conduction problems only: convection too complex
- Computation time reduction compared to 3D-FEM:
  hours → minutes,
  minutes → seconds

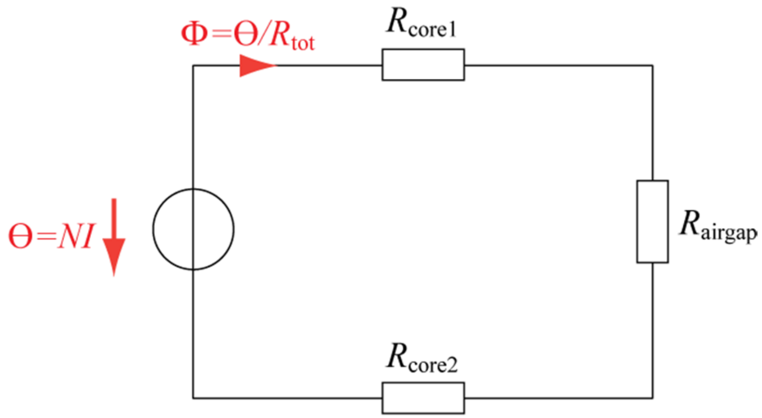






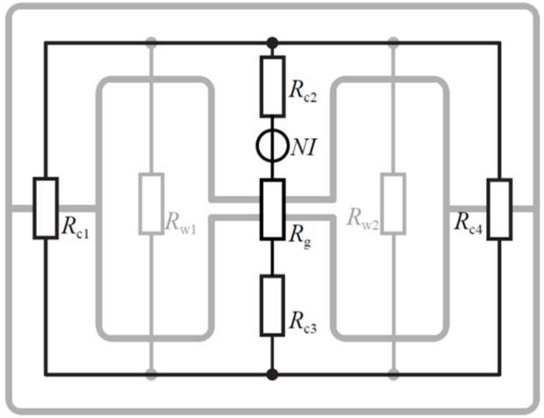
Inductor Modeling: Reluctance model





	Electric Network	Magnetic Network
Conductivity	κ	μ
Resistance	$R = l \,/\, \kappa A$	$R_{m} = l \,/\, \mu A$
Voltage	$V = \int_{P_1}^{P_2} \vec{E}\, d\vec{s}$	$V_{m} = \int_{P_1}^{P_2} \vec{H}\, d\vec{s}$
Current / Flux	$I = \iint_A \vec{J}\, d\vec{A}$	$\Phi = \iint_A \vec{B}\, d\vec{A}$

E-Core Reluctance model



GECKO RESEARCH

Inductor Loss Modeling

- Winding losses: analytic formulae well known and reasonably accurate
- Problem: **Core losses**: Improved generalized Steinmetz eqn.:

$$P_{\mathrm{v}} = \frac{1}{T}\int_0^T k_i \left|\frac{\mathrm{d}B}{\mathrm{d}t}\right|^{\alpha} \left(\Delta B\right)^{\beta-\alpha} \mathrm{d}t$$

- DC bias not considered!

- Relaxation effect not considered

- Steinmetz parameters are valid only in a limited flux density and frequency range

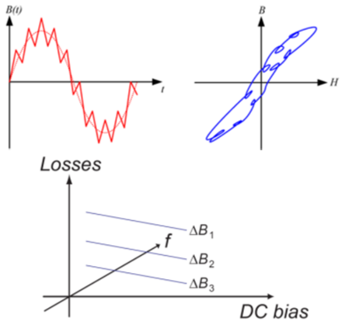
Core Loss Modeling including DC Bias

- Further improved generalized Steinmetz Equation:

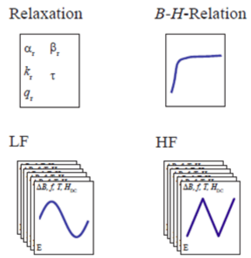
$$P_v = \frac{1}{T}\int_0^T k_i \left|\frac{dB}{dt}\right|^{\alpha} (\Delta B)^{\beta-\alpha}\, dt + \sum_{l=1}^{n} Q_{rl} P_{rl}$$

Simulated by reluctance model

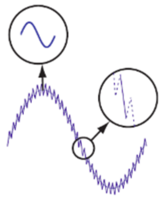
- Must measure core losses to parameterize the equation!
- Need database of core material measurements in simulation tool



Loss measurements



"Loss map" database



Simulated flux waveform

$k, k_i, \alpha, \beta, \alpha_r, \beta_r, q_r, \tau$ Equation parameters

Accurate core loss calculation

- Experimentally verified

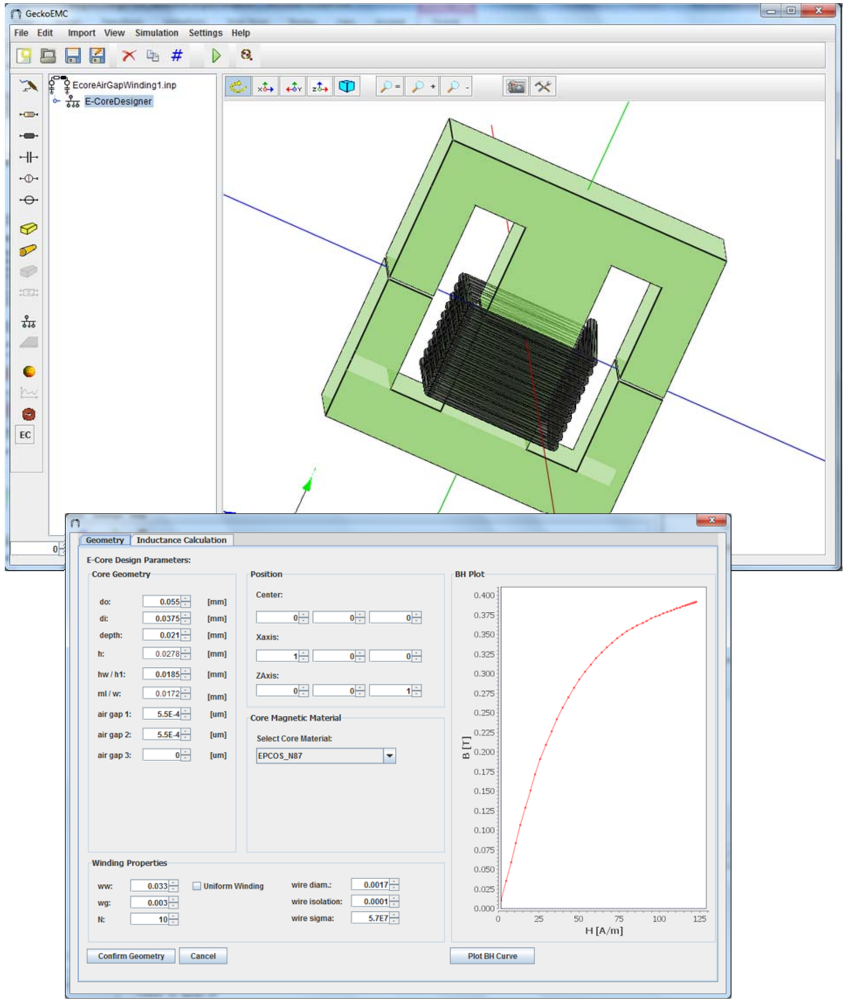
papers: J. Muehlethaler, J. W. Kolar, et al., ICPE 2011, APEC 2011, IPEC 2010

GeckoMAGNETICS: 3D Tool for Inductor Loss Calculations

Currently in Development

Inputs:

- Core Dimensions
- Winding properties (round conductor, Litz Wire, Foil Conductors & arragement)
- Material Database (B-H curve, Steinmetz paramters, loss map)
- Current/Flux waveforms (e.g. from GeckoCIRCUITS, FFT)
- Inductor thermal model



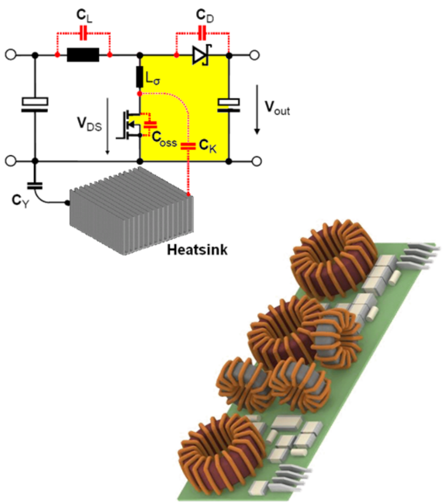
Output:

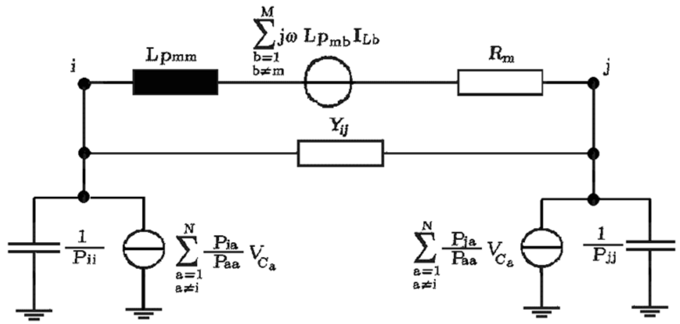
- Total losses & loss distribution
- Inductances
- Field distribution

Electromagnetic Modeling: GeckoEMC

- 3D electromagnetic modeling and simulation
  - Parasitics in modules, components
  - Layout parasitics
  - EMI filters
- Can be done with 3D FEM/FDM → usually very slow
- Solution: Partial Element Equivalent Circuit Method (PEEC)

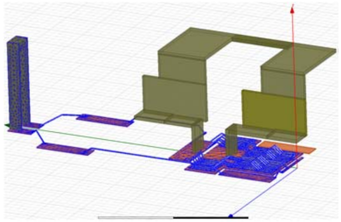
→ Model EM properties as a circuit, utilize fast circuit solver



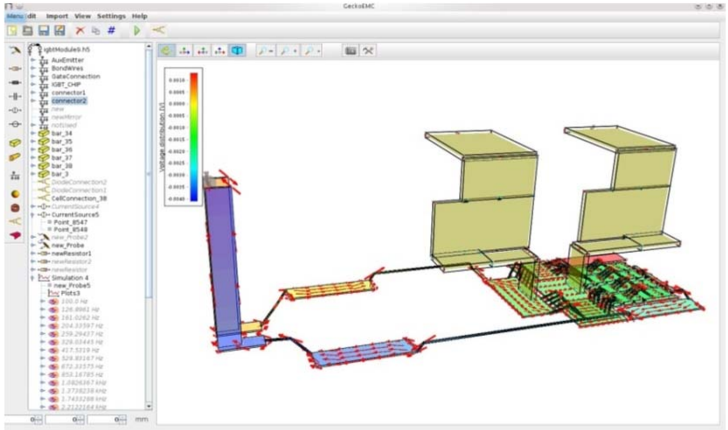


Electromagnetic Modeling: GeckoEMC

- Module modeling:

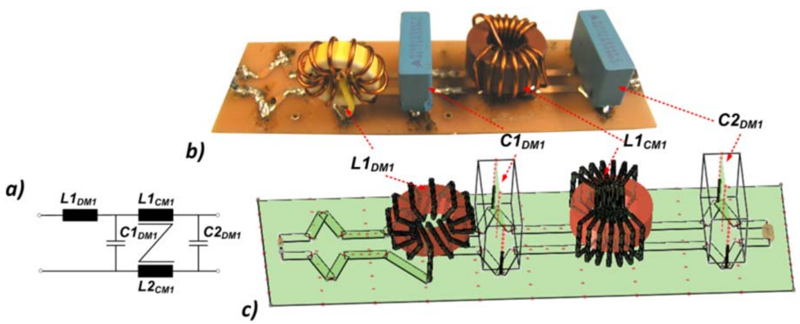


Maxwell 3D: 1 h 20 min

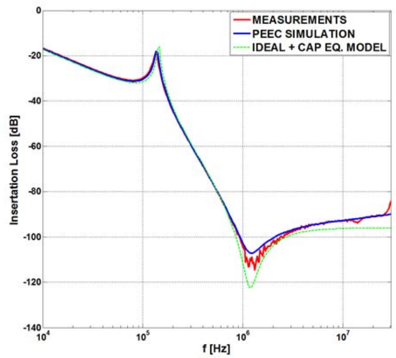


GeckoEMC: 30 sec

- EMI Filter modeling: Currently works only with toroidal inductors
  - Coupling effects considering geometric arrangement



PFC input filter stage

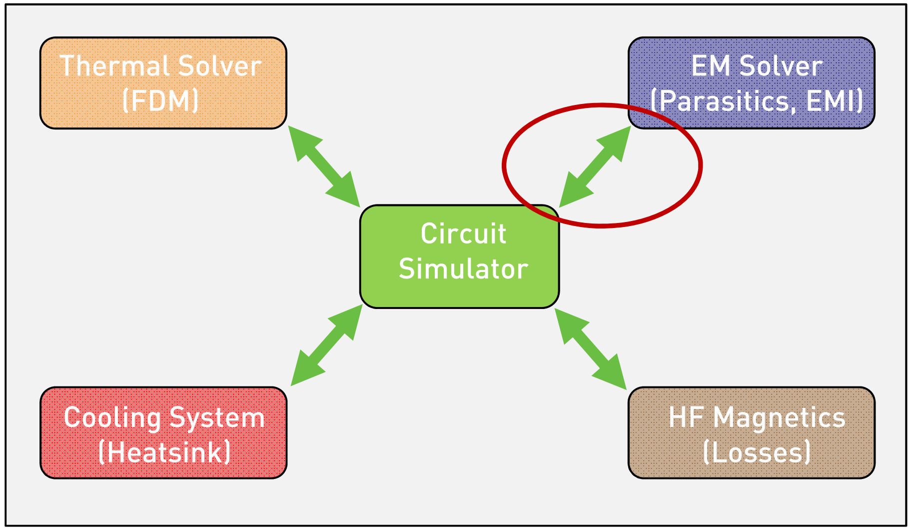


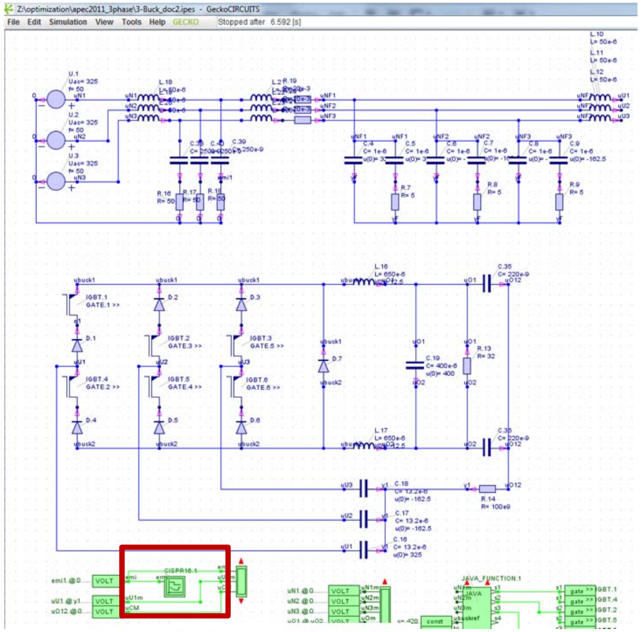
Measurements match simulation

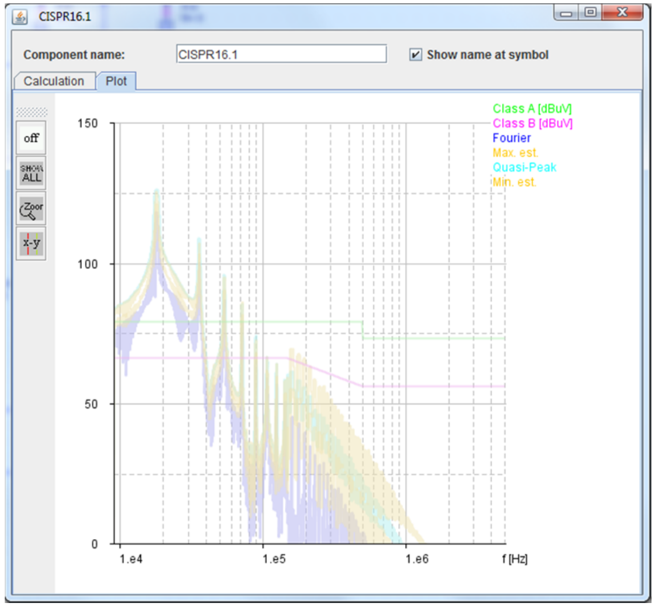
papers: I. Kovacevic, A. Muesing, J. W. Kolar, et al., CEFC 2010, IPEC 2010, COMPUMAG 2011

GECKO RESEARCH

Coupling GeckoCIRCUITS and GeckoEMC







- EMI analyzed in GeckoCIRCUITS (Test Receiver block)
- Waveform can be fed into GeckoEMC

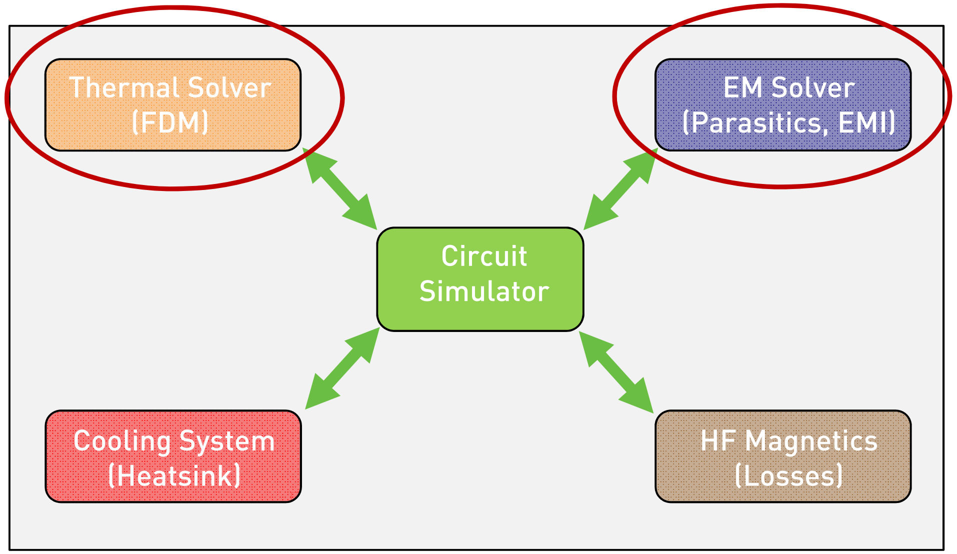
Combining Simulation Domains – MOR

Motivation: Finally, we want to include thermal models and electromagnetic models (parasitics) into a circuit simulation

- Typical: Thermal or EM solver contains > 10000 cells
- Circuit simulation: dt = 100 nsec, T = 1 sec

→ This is impossible to solve together

- Our future solution approach: Model Order Reduction (MORe)



MORe: Construct a simplified system to approximate the original system with reasonable accuracy.

Gecko-Research Software Overview