From Brute Force Grid Search to Artificial Intelligence: Which Algorithms for Magnetics Optimization?

IEEE APEC 2020: PSMA Industry Session
Design of Magnetics for Different Circuit Topologies

Thomas Guillod and J. W. Kolar

Power Electronic Systems Laboratory
ETH Zurich, Switzerland

Acknowledgement

The authors would like to thank

- *P. Papamanolis*
- *Dr. D. Rothmund*
- *Dr. R. Burkart*

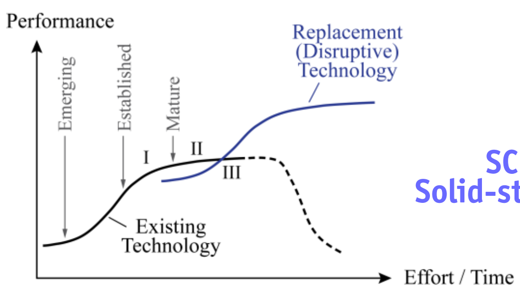
ETH zürich

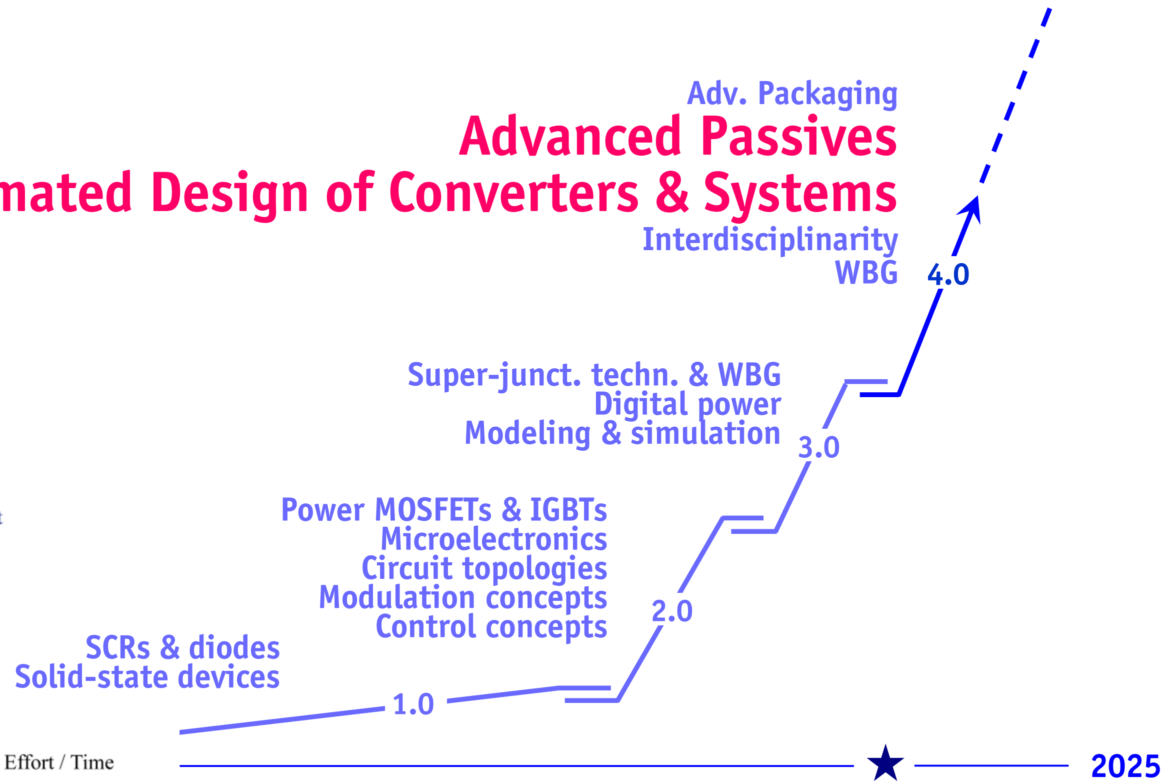
- *G. Mauro*
- *Dr. K. Leong*
- *Dr. M. Kasper*
- *Dr. G. Deboy*

infineon

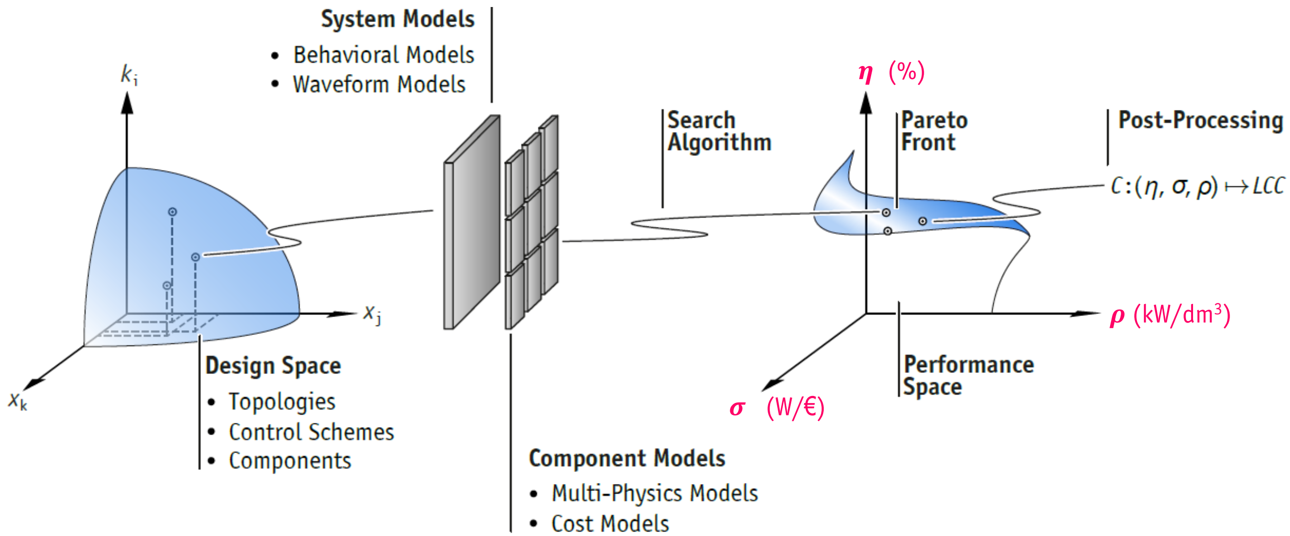
for their contributions.

▶ Design Automation in Power Electronics





▶ Multi-Objective Optimization



- **Advantages**
  - Efficiency, power density, costs, reliability, etc.
  - Virtual prototyping → time to market

- **Requirements**
  - Models & data
  - Algorithms & objectives

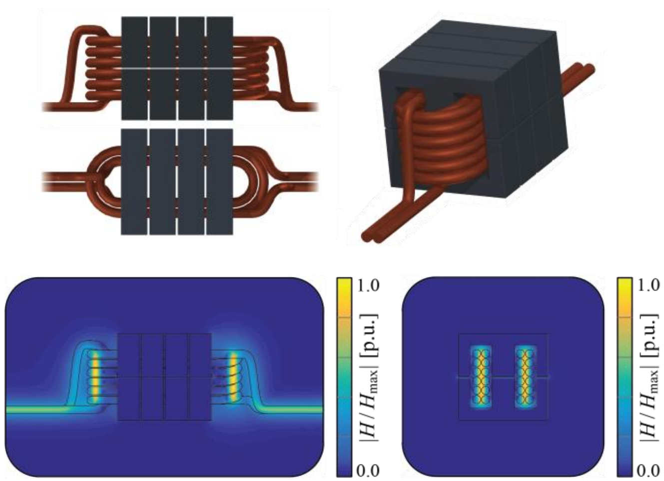
[Adapted from R. Burkart, PhD Thesis, ETHZ, 2016]

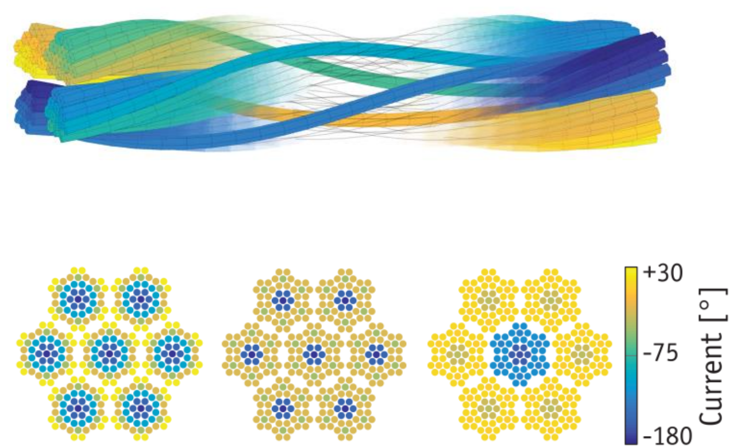
Modelling Magnetics

Accuracy
Complexity

▶ Full Numerical Model

- **Based on fundamental equations**
  - Maxwell
  - Heat transfer
  - Navier–Stokes
- **Methods**
  - FEM/FVM
  - FDM/FDTD
  - PEEC/MoM
- **Properties**
  - Highest accuracy
  - High modelling effort
  - High computational effort



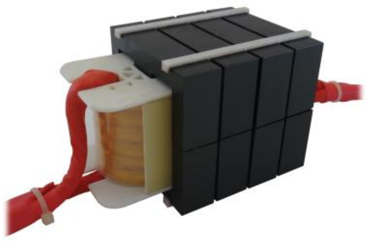


- **Useful for final validation**
- **Too time consuming for optimization**

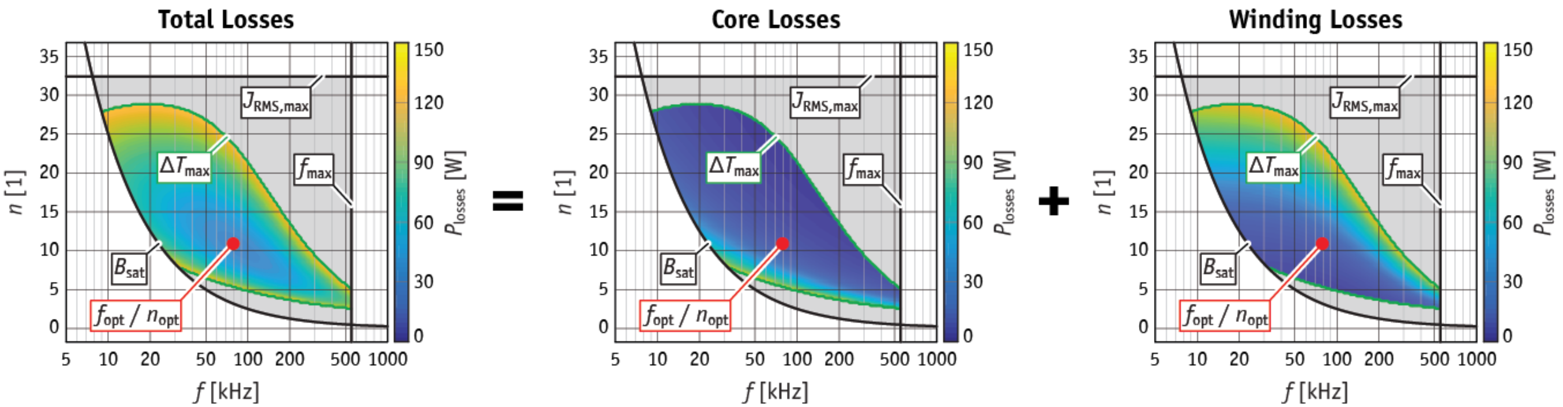
[Adapted from T. Guillod, PhD Thesis, ETHZ, 2018]

▶ Full Analytical Model

- **Modelling approach**
  - Simplified physics
  - Simple equations
  - Explicit solution
- **Properties**
  - Low accuracy
  - Low modelling effort
  - Low computational effort



- **MF transformer analytical model**

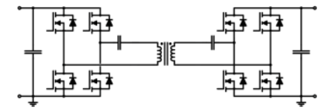


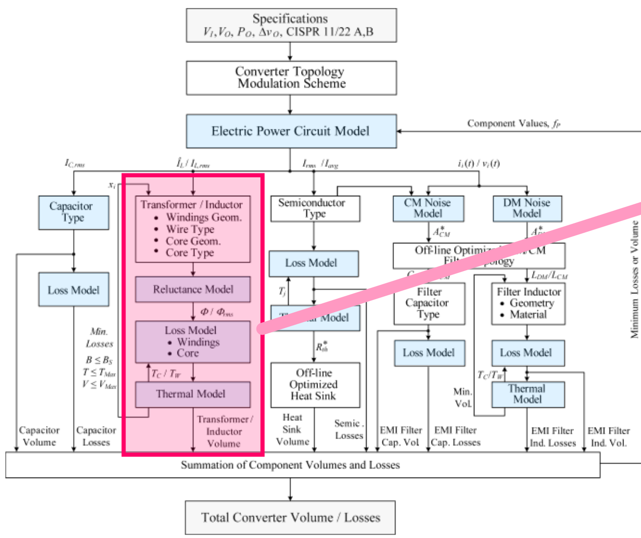
- **Useful for initial estimation & understanding**
- **Too many simplifications for virtual prototyping**

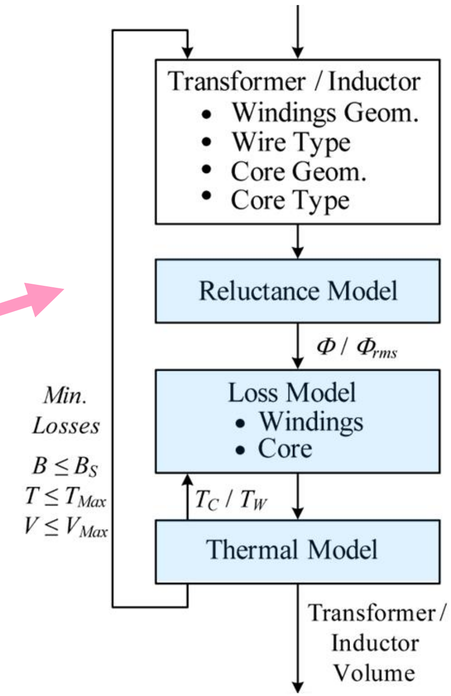
[Adapted from T. Guillod, IEEE CPSS, 2019]

▶ Semi-Numerical Model

- **Modelling approach**
  - Complex equations
  - Numerical solution
  - Thermal-loss coupling
- **Properties**
  - High accuracy
  - Medium modelling effort
  - Medium computational effort





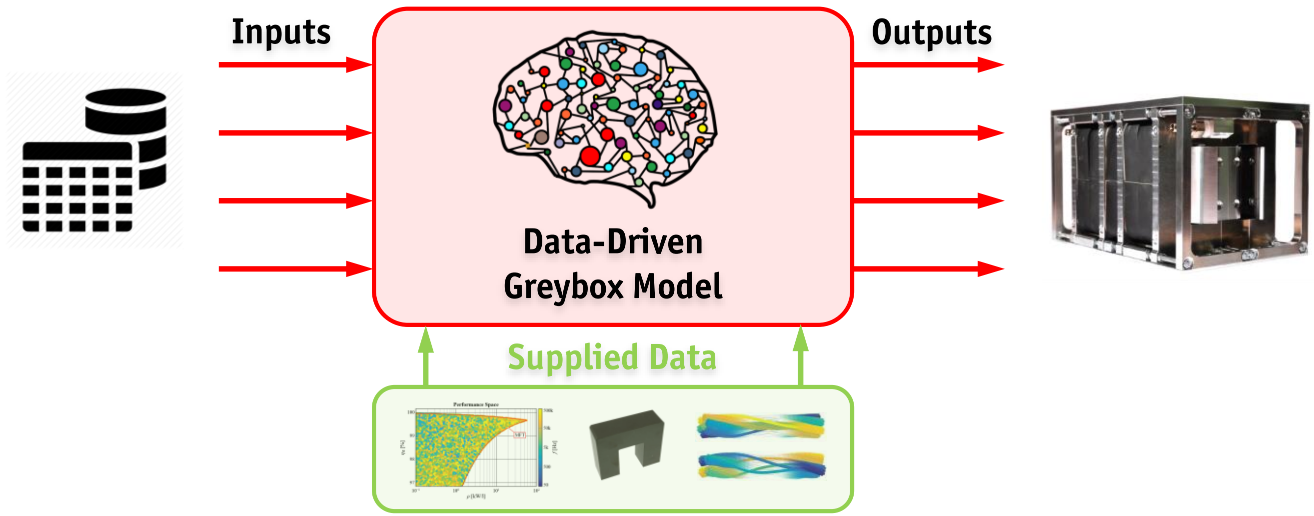


- **Easy to integrate in a full converter model**
- **Typical choice for optimization**

[Adapted from R. Burkart, PhD Thesis, ETHZ, 2016]

▶ Data-Driven Model

- **Modelling approach**
  - Limited physical meaning
  - From measured or simulated data
- **Methods**
  - Interpolation / regression
  - Artificial intelligence
- **Properties**
  - Versatile method
  - Limited validity range



- **New class of model for magnetic?**

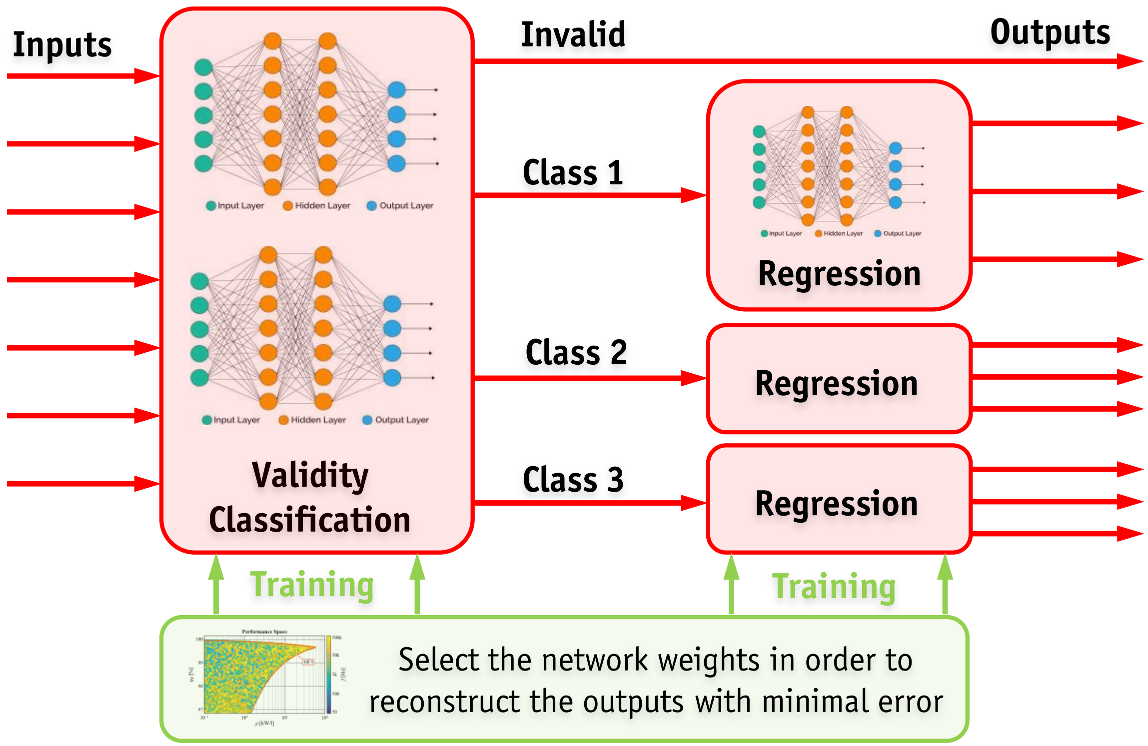
[Theoretical background: S. Skansi, Introduction to Deep Learning, 2018]

▶ Artificial Neural Networks

- **Artificial neural networks**
  - Machine learning
  - Input/output mapping

- **Advantages**
  - Versatile method
  - Very fast evaluation

- **Difficulties**
  - Choice of the network & data
  - Extrapolation is difficult

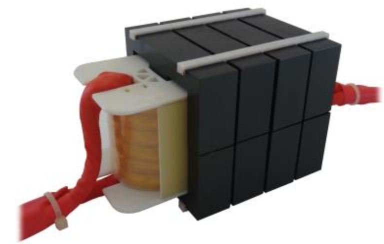


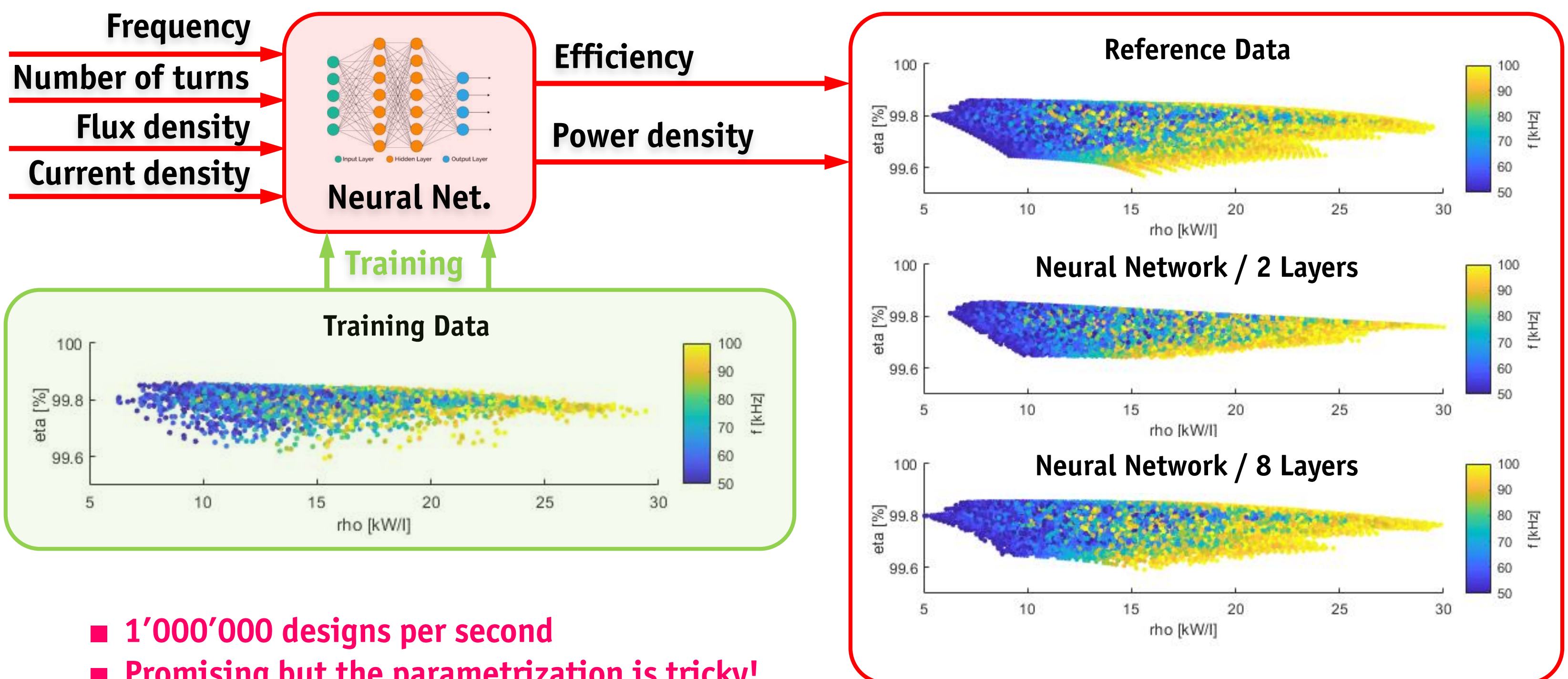
[Theoretical background: S. Skansi, Introduction to Deep Learning, 2018]

▶ Artificial Neural Networks

- **MF transformer model**
  - Semi-numerical model
  - Thermal-loss coupling

- **Artificial neural networks**
  - 5’000 designs for training
  - Prediction of 130’000 designs





- **1’000’000 designs per second**
- **Promising but the parametrization is tricky!**

[Theoretical background: S. Skansi, Introduction to Deep Learning, 2018]

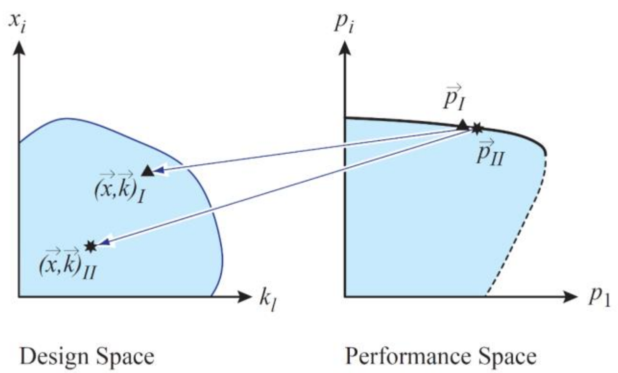
Optimizing Magnetics

Accuracy
Complexity

▶ Model Properties

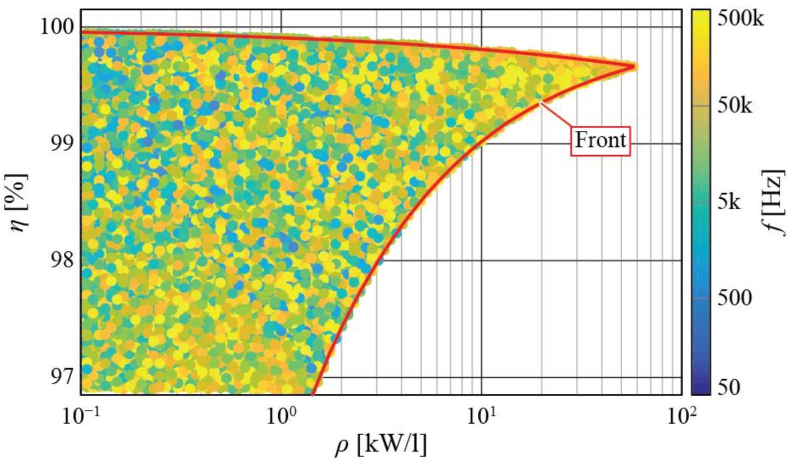
- **Properties (worst case)**
  - Multivariable (input/output)
  - Non-linear
  - Non-convex
  - Non-continuous
  - No (explicit) gradient
  - Constrained (explicit/implicit)
  - Mixed-integer (discrete variables)

- **Which optimization method?**
- **The perfect solution does not exist**



- **Design space to performance space**
  - No clear trends
  - No clear mapping
  - No clear optimum
  - Analytical opt. are not sufficient

- **Design space diversity**

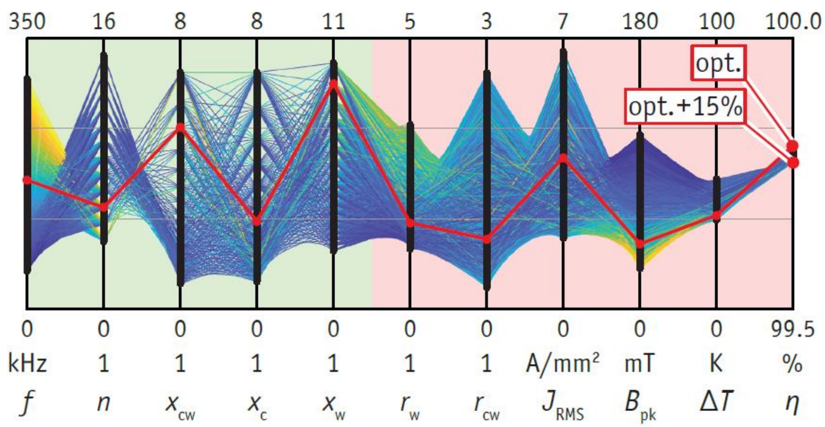


[Adapted from T. Guillod, IEEE CPSS, 2019]

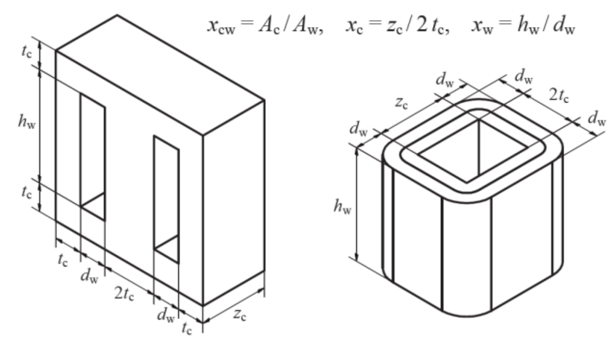
▶ Design Space Diversity

- **MF transformer semi-numerical model**
  - Fixed power: 20kW
  - Fixed volume: 1dm^3
  - Loss range: [P_{opt}, P_{opt} +15%]
- **300'000 designs with similar performances**
  - Frequency: [50, 300] kHz
  - Flux density: [25, 120] mT
  - Current density: [1.8, 6.5] A/mm^2

Quasi-Optimal Designs



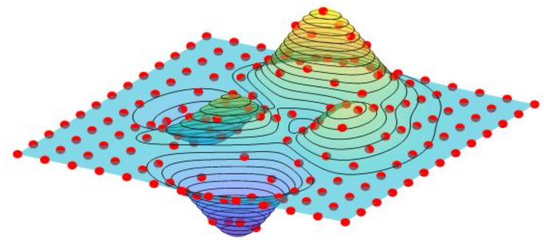
Geometrical Aspect Ratio



- **Local optima and/or flat optima**
- **Robustness of optimization algorithms?**
- **Opportunities for additional constraints?**

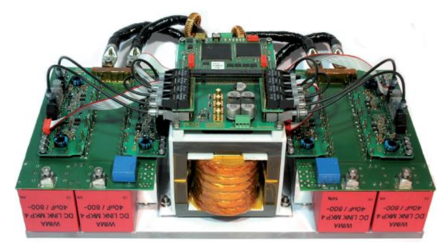
[Adapted from T. Guillod, IEEE CPSS, 2019]

► Brute Force Grid Search



- **Algorithms properties**
  - Extremely robust
  - No restriction on the model
  - Exponential scaling
  - Relatively slow but parallelizable
  - Can be combined with heuristics

- **A desktop computer makes 25-400 billion floating point operations per second!**
- **A cloud computing server cost 5-10¢ per hour!**



- **DC-DC resonant converter**
  - Semi-numerical model
  - Accurate thermal-loss coupling
  - Vectorized, parallel, and optimized
  - 100’000 designs per second

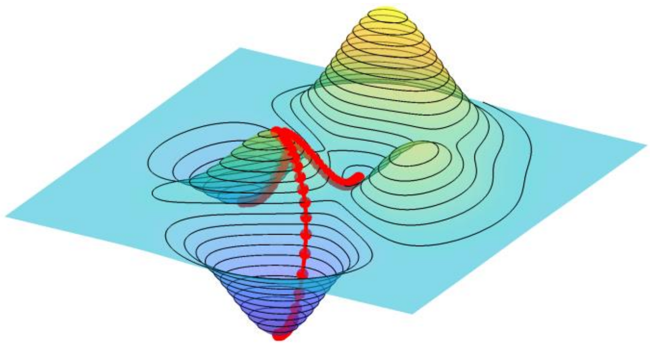
- **Brute force is (whenever possible) the best solution**

[Theoretical background: S. Rao, Engineering Optimization, 2009]

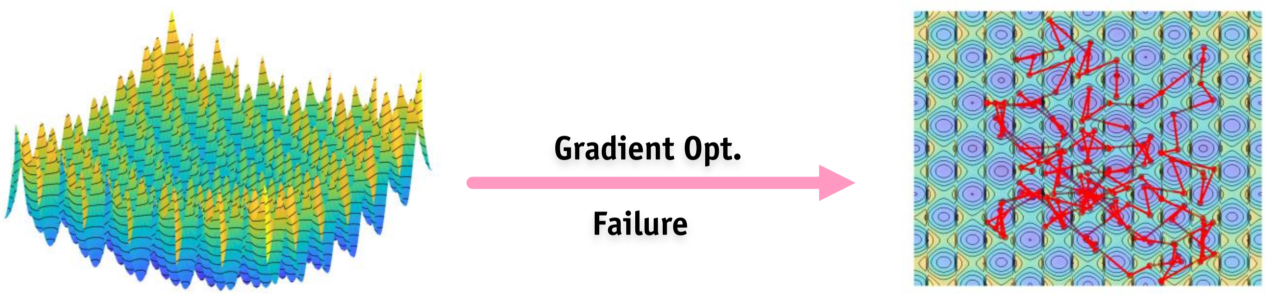
► Gradient, Simplex, Geometric Programming

- **Algorithms properties**
  - Extremely fast convergence
  - Problems with local minima
  - Problems with design space diversity

- **Restrictions on the model**
  - Smooth function (gradient opt.)
  - Posynomial function (geom. prog.)
  - No discrete variables (various alg.)
  - No complex constraints (various alg.)



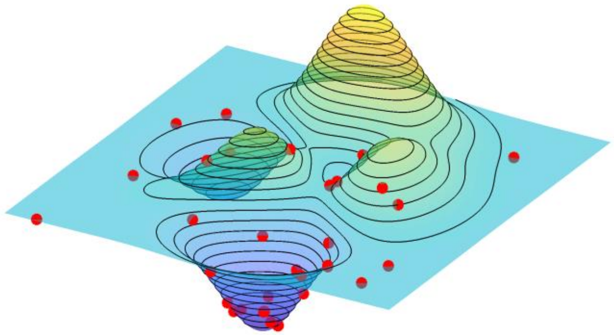
- **Restricted to problems with compatible models and constraints**
- **Can be combined with other approaches (e.g. brute-force)**



[Theoretical background: S. Rao, Engineering Optimization, 2009]

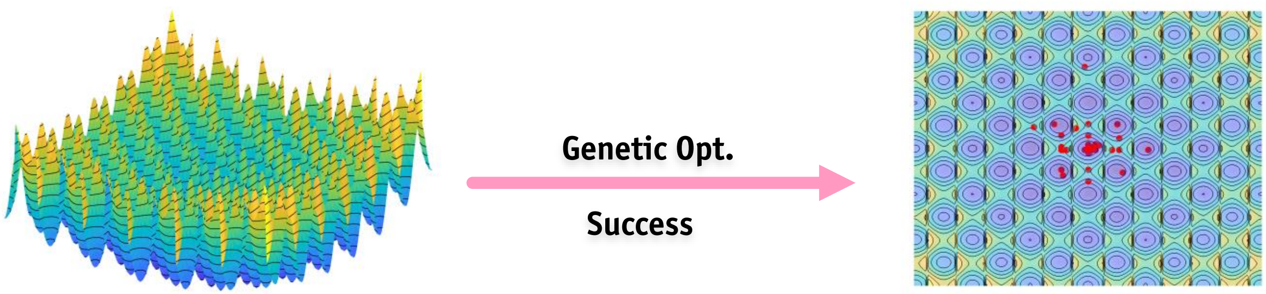
► Genetic Optimization, Particle Swarm, Simulated Annealing

- **Algorithms properties**
  - Stochastic approach
  - Slower convergence
  - Compatible with local minima
  - Compatible with design space diversity
  - Few restrictions on the model



- **Genetic algorithm**
  - Initial population
  - Fitness / selection
  - Crossover / mutation

- **Good trade-off between robustness and speed**

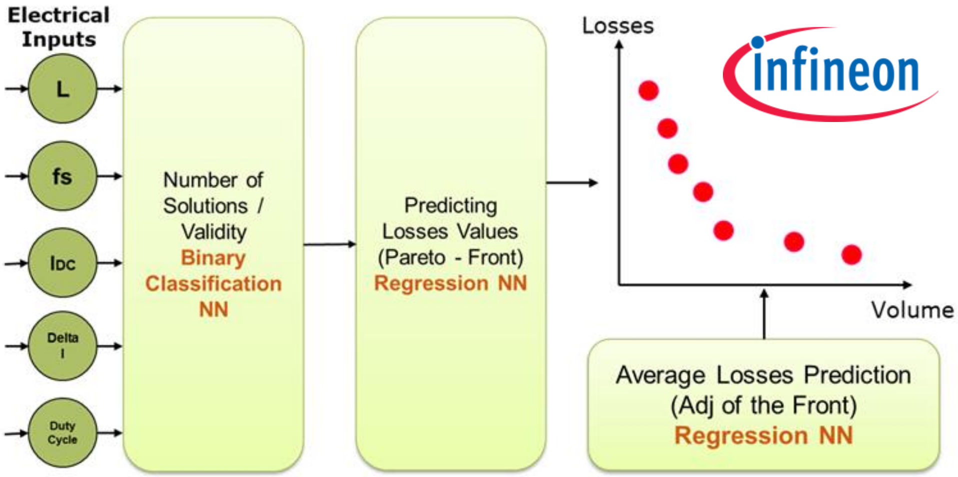


[Theoretical background: S. Rao, Engineering Optimization, 2009]

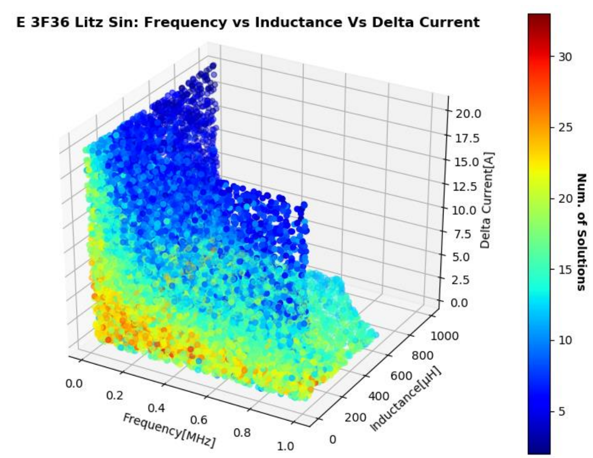
▶ Artificial Neural Networks

- **Deep learning**
  - Given specifications
  - Extract Pareto Front
  - Within seconds
- **Artificial neural networks**
  - Prediction the number of sol.
  - Predicting the losses
  - Adjusting the Pareto front
- **Difficulties**
  - Choice of the network & data
  - Handling discrete data
  - Scaling to large problems

Neural Network for Inductor Pareto Fronts



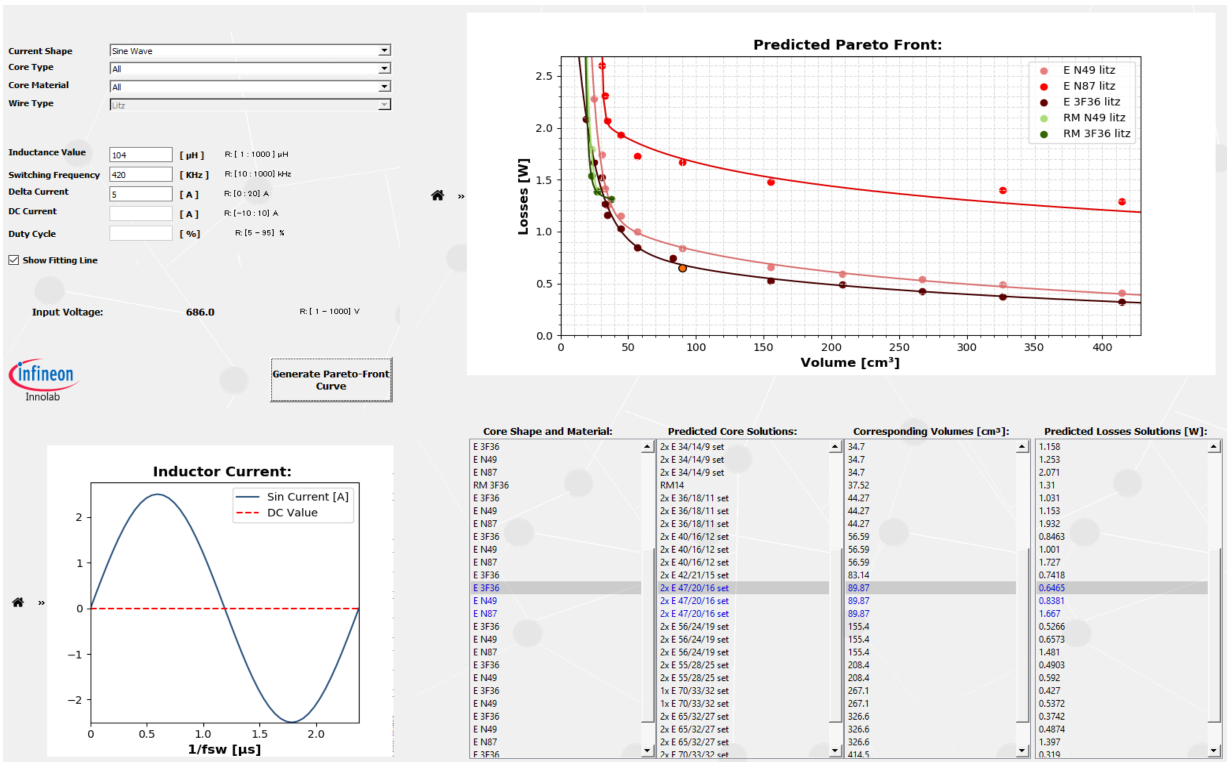
Training Data from Genetic Alg.



[Infineon Technologies, Villach, Austria]

▶ Artificial Neural Networks

- **Generates inductor Pareto fronts in less than 5 seconds!**



- **For quick comparison between technologies**
- **For getting a good initial design guess**

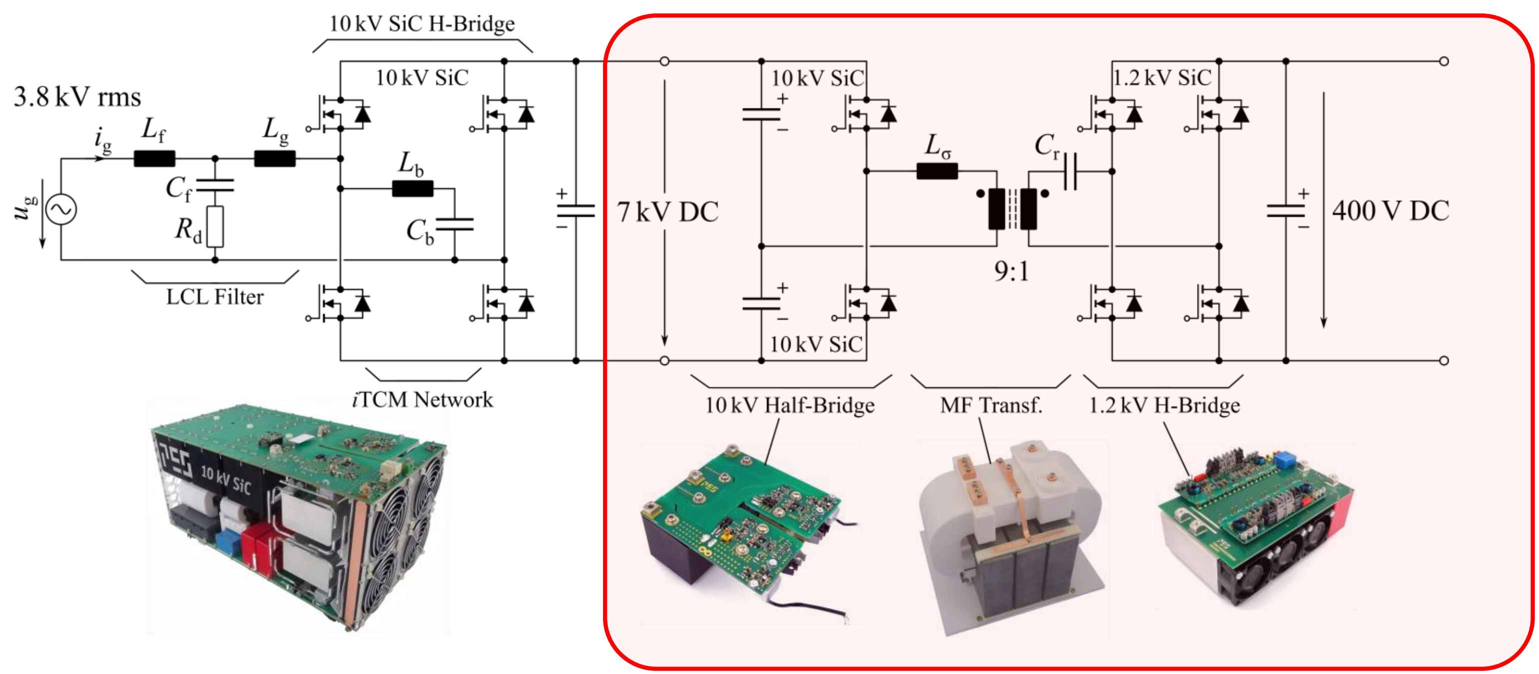
[Infineon Technologies, Villach, Austria]

Case Study MV Converter

Solid-State Transformer Design Space Diversity

▶ Case Study: Solid-State Transformer for Datacenter

- **Single-stage SST for datacenters presented by ETH zürich**
  - 3.8kV AC input
  - 400V DC output
  - 25kW
  - 10kV SiC technology



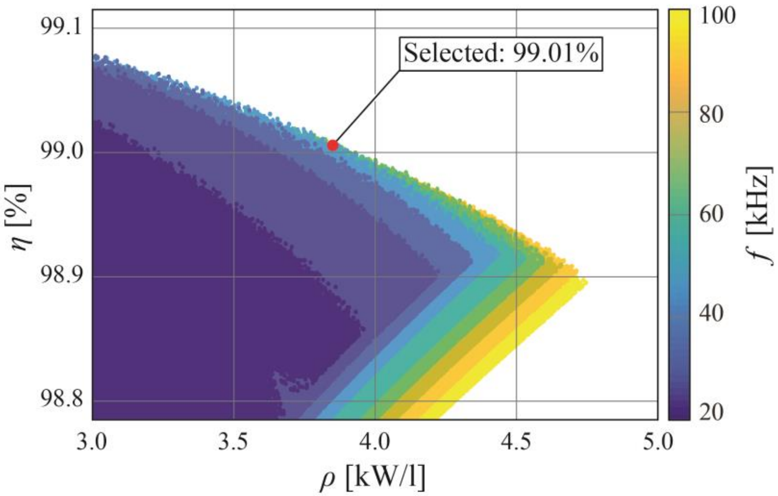
- **DC-DC perf. target: 99% & 3kW/dm³ & single hardware iteration**
- **How to optimize using the design space diversity?**

[Adapted from D. Rothmund, IEEE JESTPE, 2018]

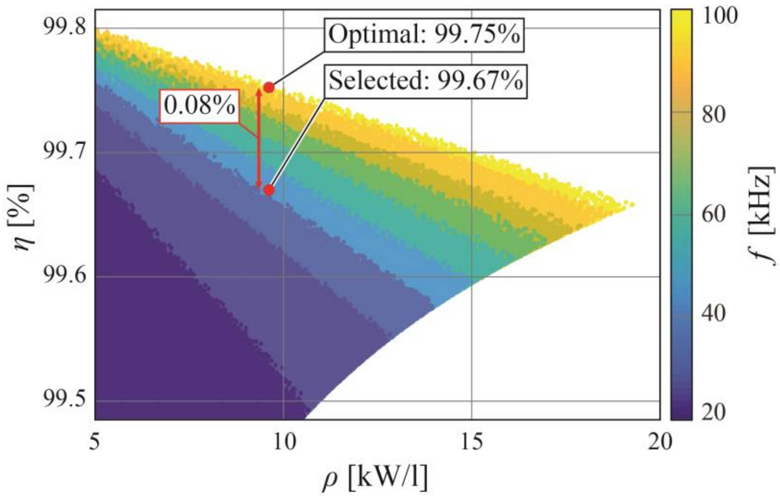
► Converter Pareto Front

- **Trade-off: switching frequency**
  - Transformer: reduced volt-second product
  - Semiconductors: switching losses
- **Selected frequency: 48kHz**
  - System optimum: 48kHz
  - Transformer optimum: 100kHz

DC-DC Pareto Front



Transformer Pareto Front



- **Global optimum is composed of sub-optimal components**
- **Design space diversity?**

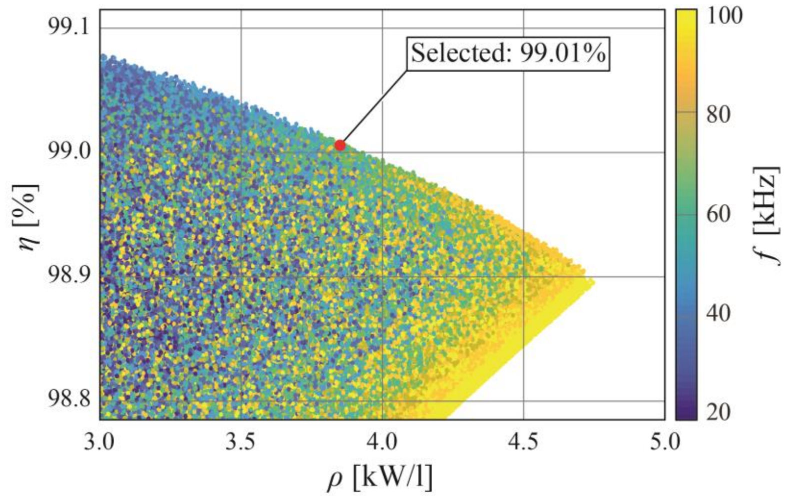
[Adapted from T. Guillod, PhD Thesis, ETHZ, 2018]

▶ Converter Pareto Front

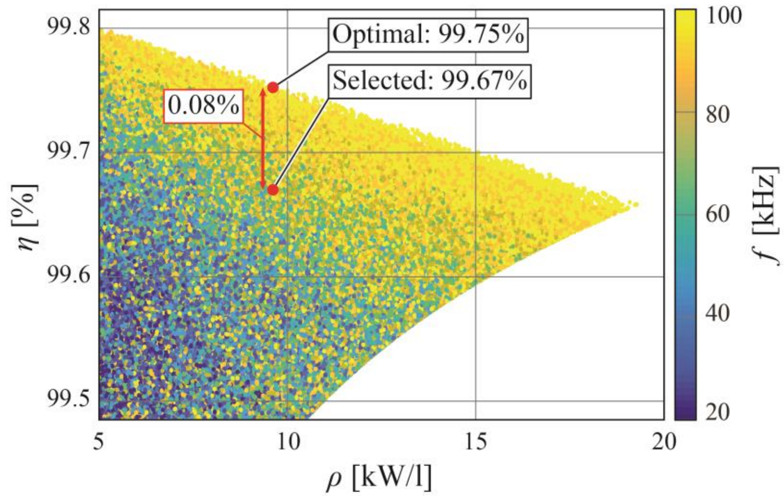
- **Trade-off: switching frequency**
  - Transformer: reduced volt-second product
  - Semiconductors: switching losses

- **Selected frequency: 48kHz**
  - System optimum: 48kHz
  - Transformer optimum: 100kHz

DC-DC Pareto Front



Transformer Pareto Front

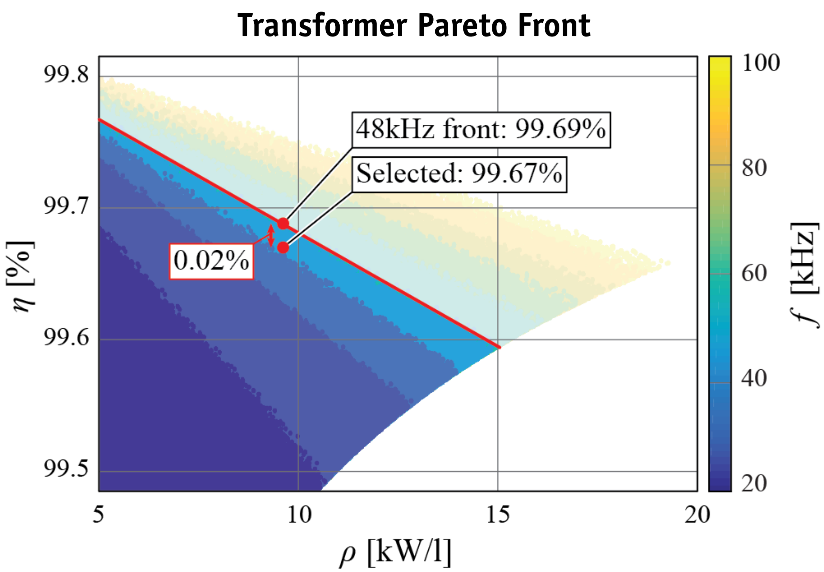


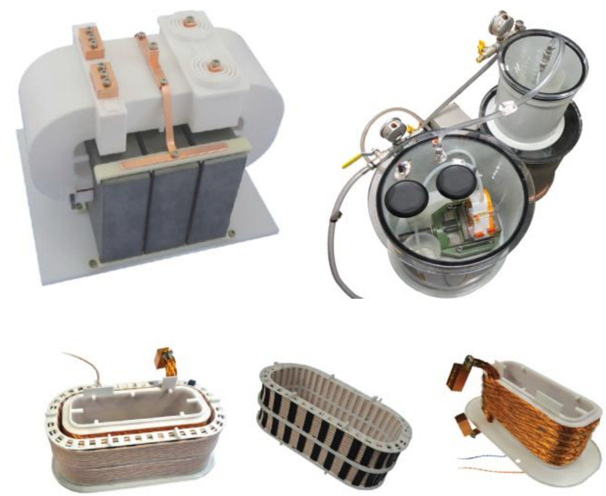
- **How to use the design space diversity?**
- **Brute force grid search / genetic alg.**

[Adapted from T. Guillod, PhD Thesis, ETHZ, 2018]

▶ Design Space Diversity: Accommodating Practical Constraints

- **Transformer optimization**
  - Every core geometry
  - Every litz wire stranding
- **Available parts**
  - Core & winding
  - Which impact?
- **Practical constraints**
  - Manufacturability
  - Which impact?





- **Accommodating available core & litz wires: 0.02% impact**
- **Design space diversity mitigates the impact**

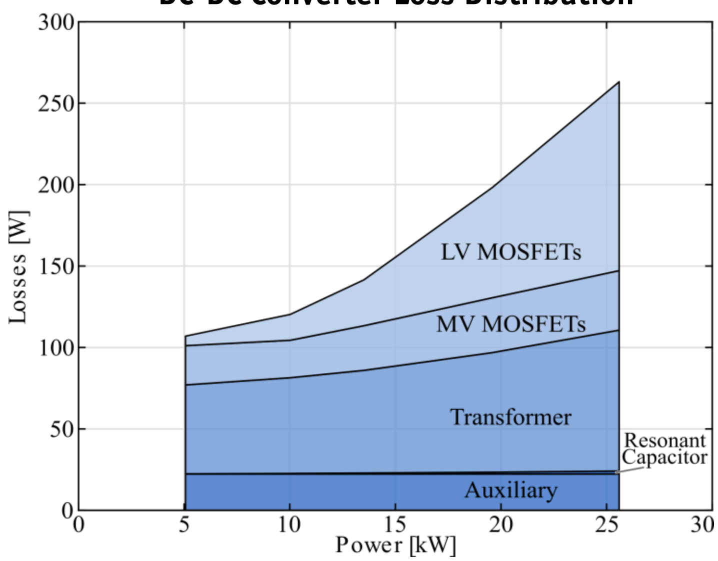
[Adapted from T. Guillod, PhD Thesis, ETHZ, 2018]

▶ Design Space Diversity: Adding a Secondary Goal

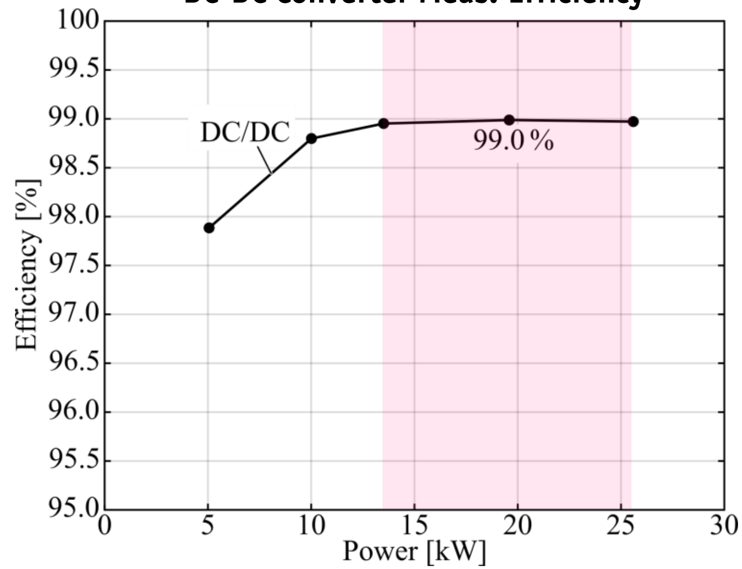
- **Partial load efficiency as an additional trade-off**
  - No-load losses (core)
  - Load losses (winding)
  - Negligible impact on the full-load efficiency

★ **99.0% @ 100% load**
★ **99.0% @ 50% load**
★ **3.8 kW/dm^3**

DC-DC Converter Loss Distribution



DC-DC Converter Meas. Efficiency



- **Design space diversity means that additional goals are achievable** ✔

[Adapted from D. Rothmund, IEEE JESTPE, 2018]

Conclusion & Outlook

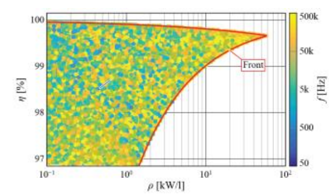
Model & Optimization
Future Research Areas

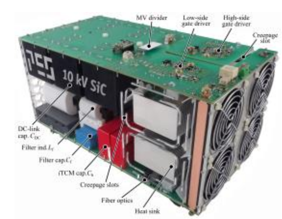
▶ Conclusion & Outlook

- **Models**
  - Analytical model for basic comparison
  - Semi-numerical model for optimization
  - Numerical model for verification
  - Data-driven model has potential
- **Design space diversity**
  - Different designs → same performances
  - Enable add. objectives and constraints
  - Should be checked → don’t miss opportunities
- **Optimization**
  - Brute force is robust and reasonably fast
  - Genetic, part. swarm, neural network, etc.
  - Care is required: no guarantee for global opt.
- **Remaining challenges**
  - Integration in industrial context
  - Readily available software, model, data, etc.

```
##Objective Function
fun = lambda x: (x[0] - 1)**2 + (x[1]-2.5)**2

#Constraints
cons = ({'type': 'ineq', 'fun': lambda x:  x[0] - 2 * x[1] + 2},
{'type': 'ineq', 'fun': lambda x: -x[0] - 2 * x[1] + 6},
{'type': 'ineq', 'fun': lambda x: -x[0] + 2 * x[1] + 2})
```





?

Thank You!

Questions? guillod@lem.ee.ethz.ch