REALIZATIONS OF SPACE MAPPING BASED NEUROMODELS OF MICROWAVE COMPONENTS

J.W. Bandler, J.E. Rayas-Sánchez, F. Wang* and Q.J. Zhang*

Simulation Optimization Systems Research Laboratory
and Department of Electrical and Computer Engineering
McMaster University, Hamilton, Canada L8S 4K1

* Department of Electronics
Carleton University, Ottawa, Canada K1S 5B6

bandler@mcmaster.ca
www.sos.mcmaster.ca

presented at

AP2000 Millennium Conference on Antennas and Propagation, Davos, Switzerland, April 13, 2000

Artificial Neural Networks (ANN) in Modeling

Artificial Neural Networks are suitable in modeling high-dimensional and highly nonlinear problems

ANN models are computationally efficient and can be more accurate than empirical models

multilayer feedforward networks can approximate any measurable function to any desired level of accuracy, provided a deterministic relationship between input and target exists *(White et al., 1992)*

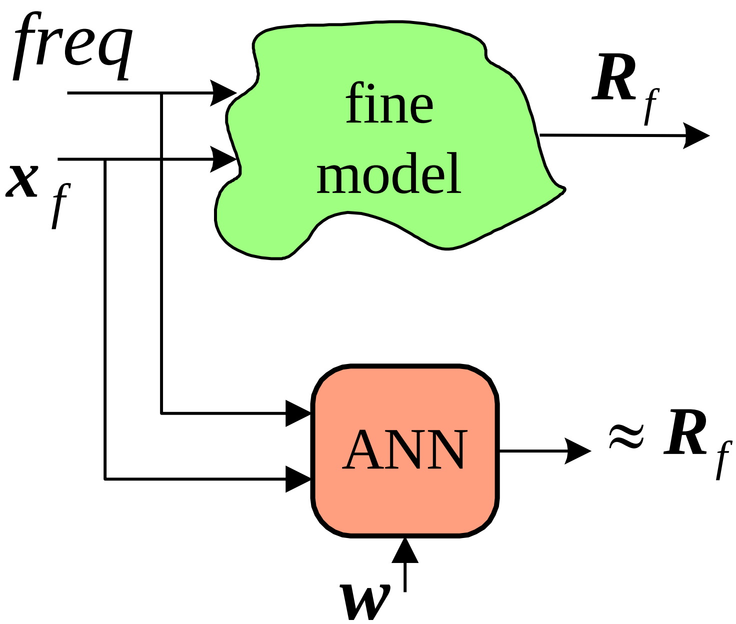
ANNs that are too small cannot approximate the desired input-output relationship

ANNs with too many internal parameters perform correctly in the learning set, but give poor generalization ability

ANNs are suitable models for microwave circuit optimization and statistical design *(Zaabab, Zhang and Nakhla, 1995, Gupta et al., 1996, Burrascano and Mongiardo, 1998, 1999)*

Space Mapping-based neuromodeling techniques significantly decrease the number of EM simulations needed for training *(Bandler et al., 1999)*

Conventional Neuromodeling of Microwave Components



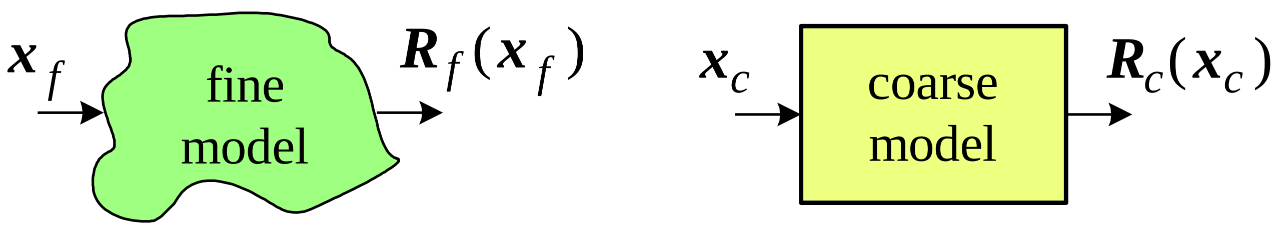
many learning samples are usually needed to ensure model accuracy

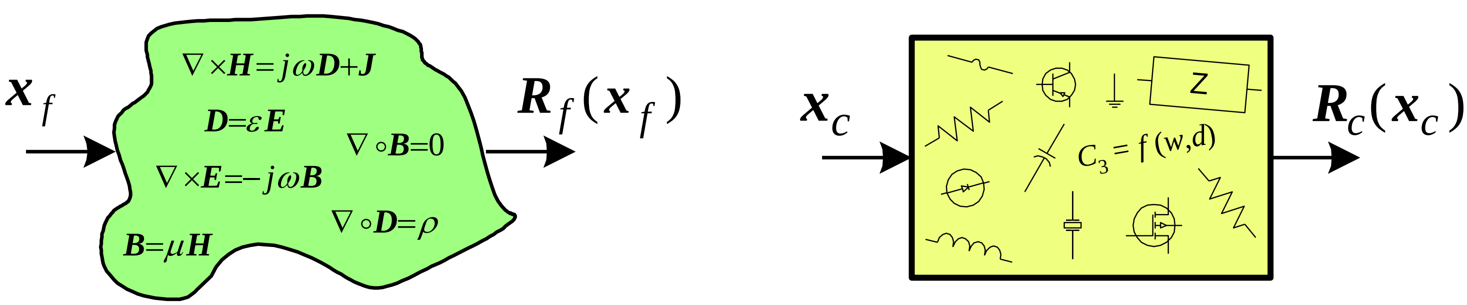
the number of learning samples needed to approximate a function grows exponentially with the ratio of the dimensionality to the function's degree of smoothness (*Stone, 1982*)

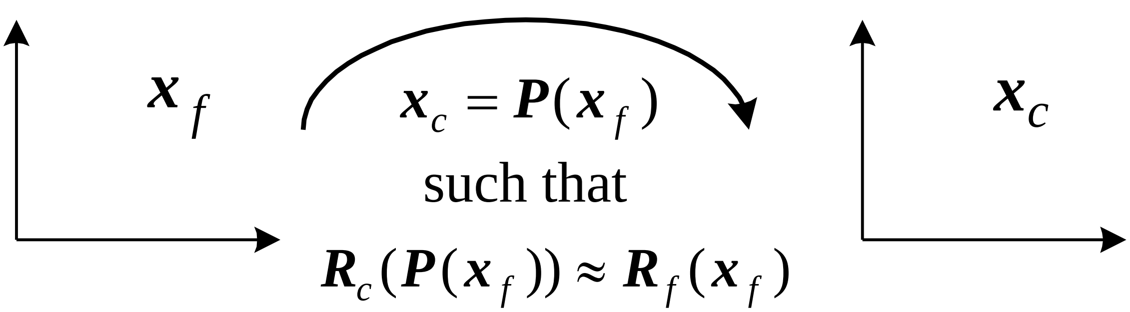
even with sufficient training data, the reliability of MLPs for extrapolation may be very poor

The Aim of Space Mapping

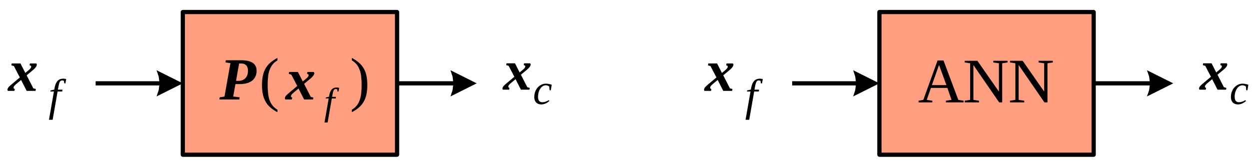
(Bandler et al., 1994-)



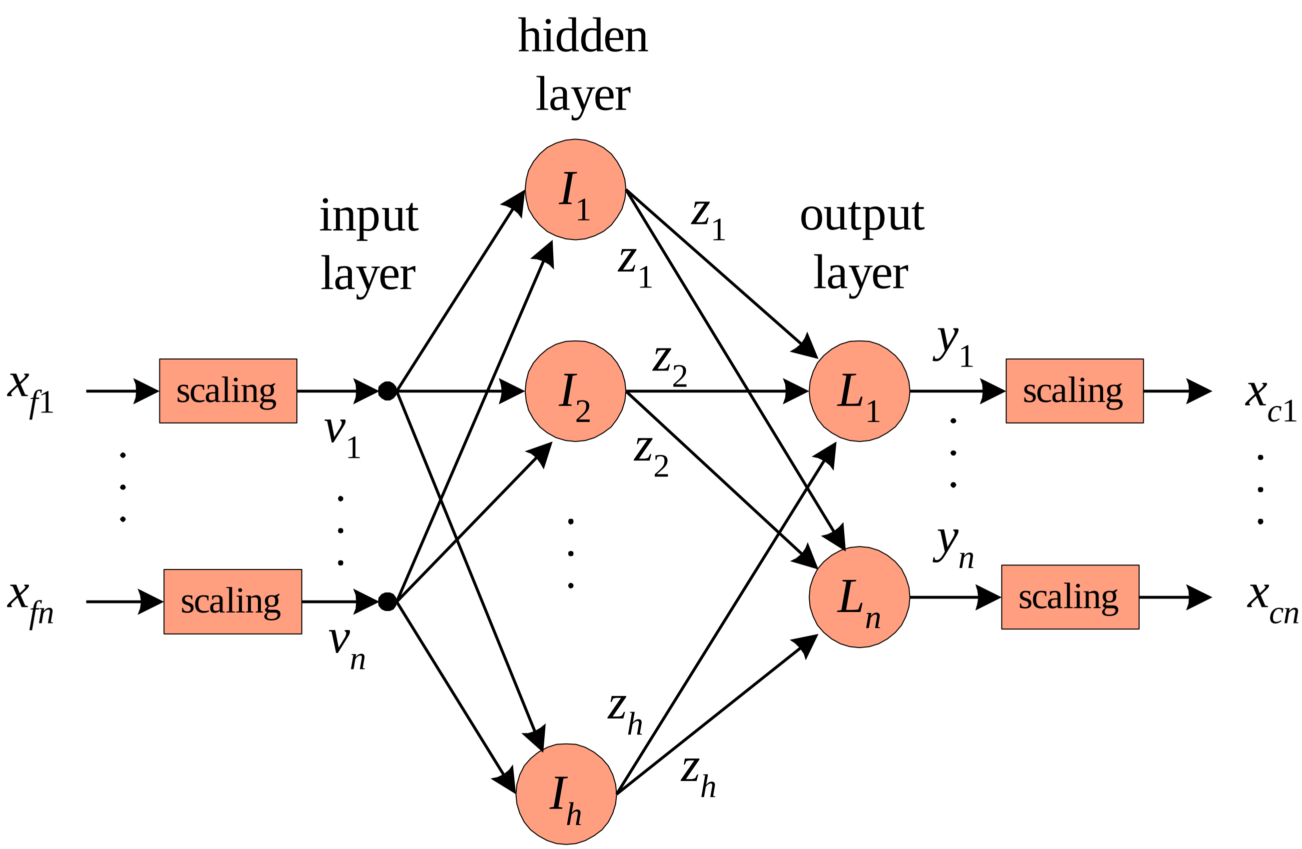




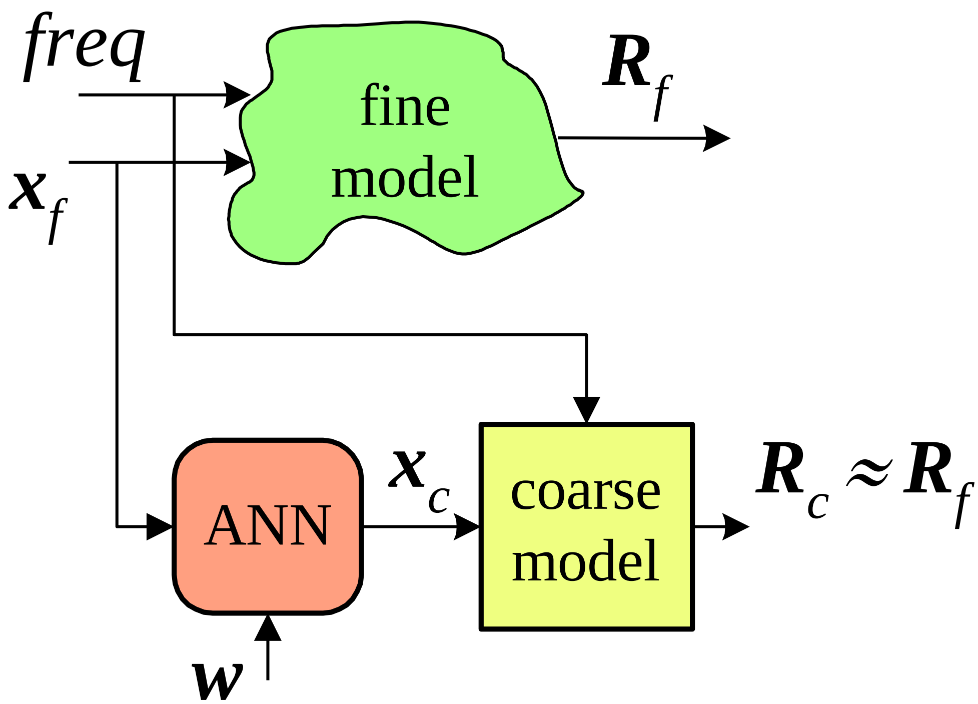
Neural Space Mapping



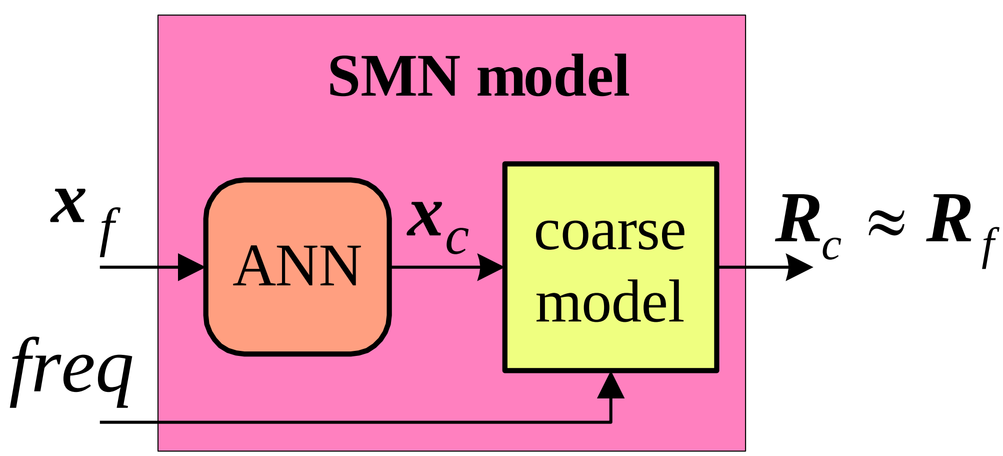
using a three layer perceptron (3LP)



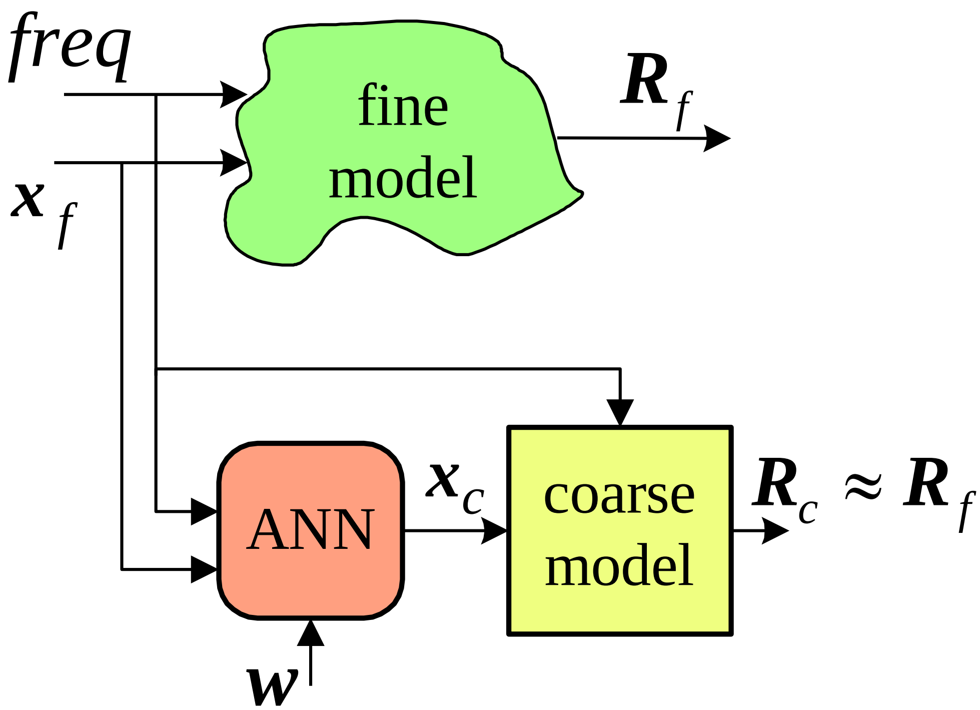
Space Mapped Neuromodeling (SMN) Concept



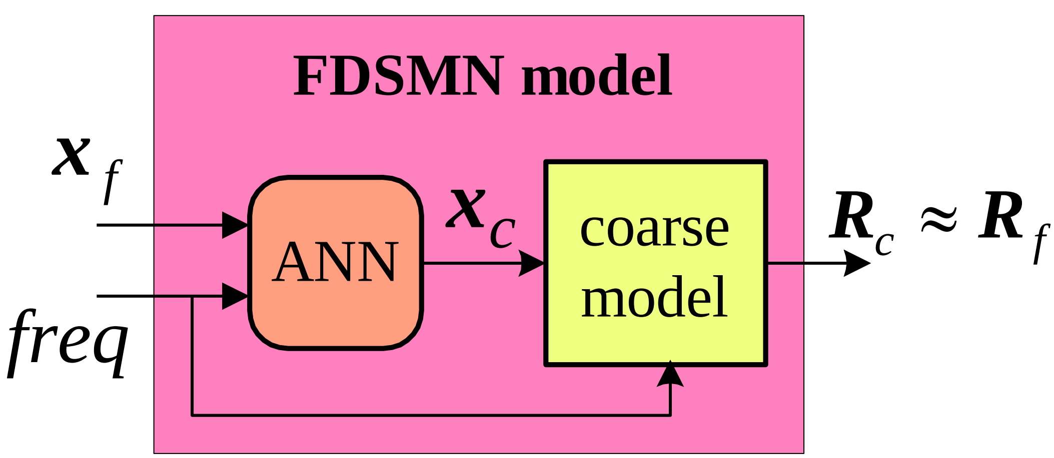
once the ANN is trained



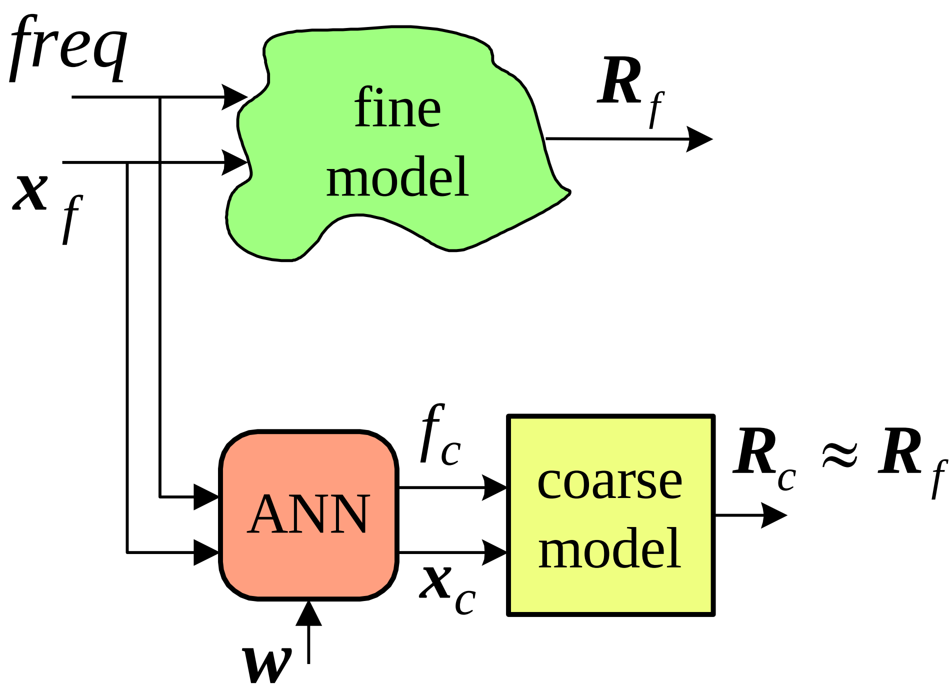
Frequency-Dependent Space Mapped Neuromodeling (FDSMN) Concept



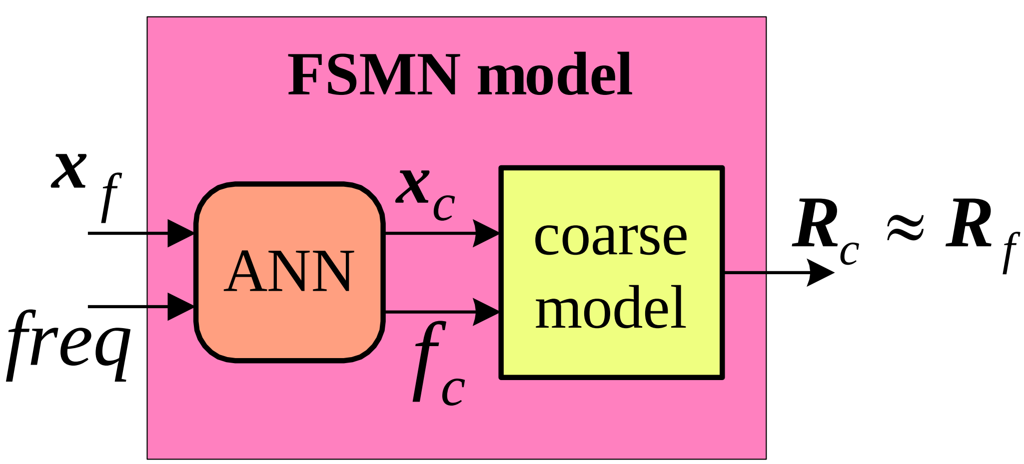
once the ANN is trained



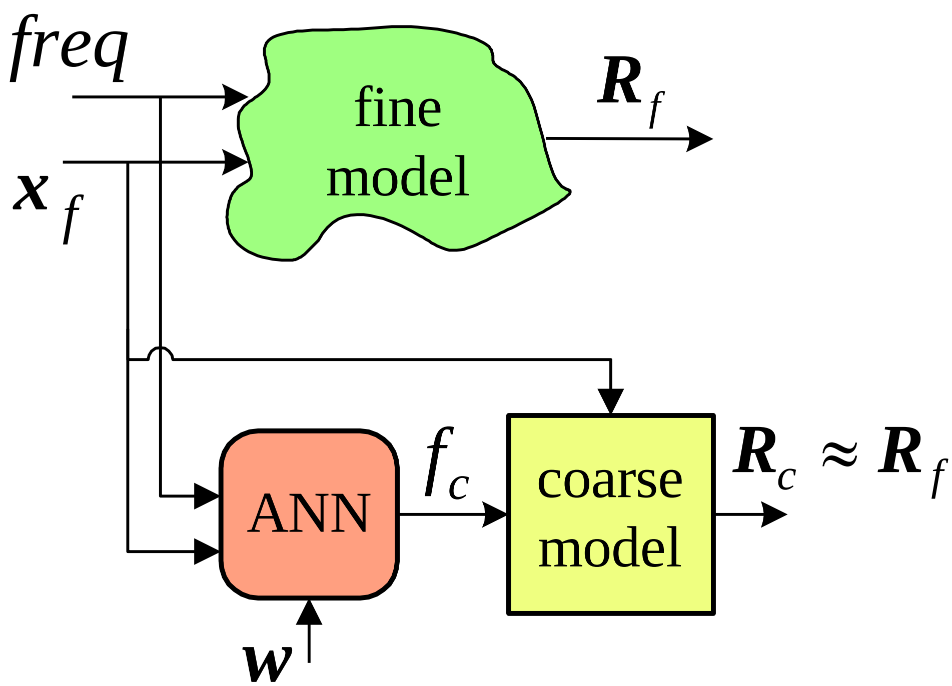
Frequency Space Mapped Neuromodeling (FSMN) Concept



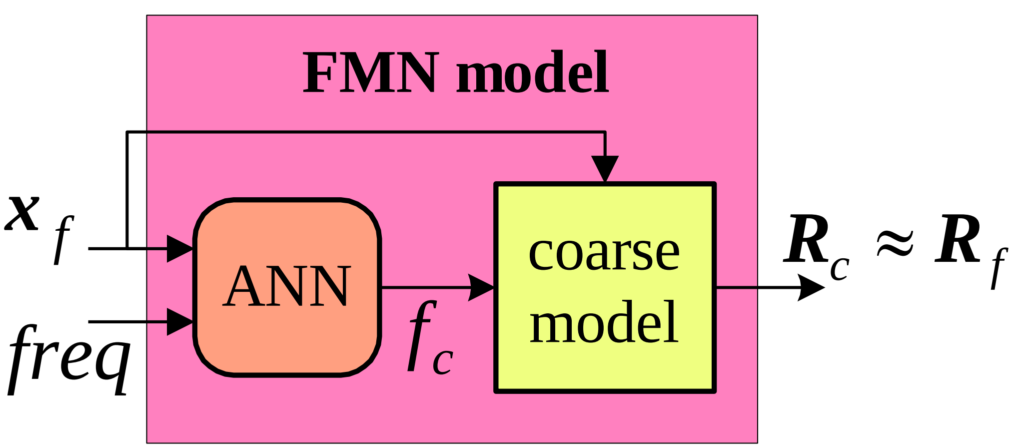
once the ANN is trained



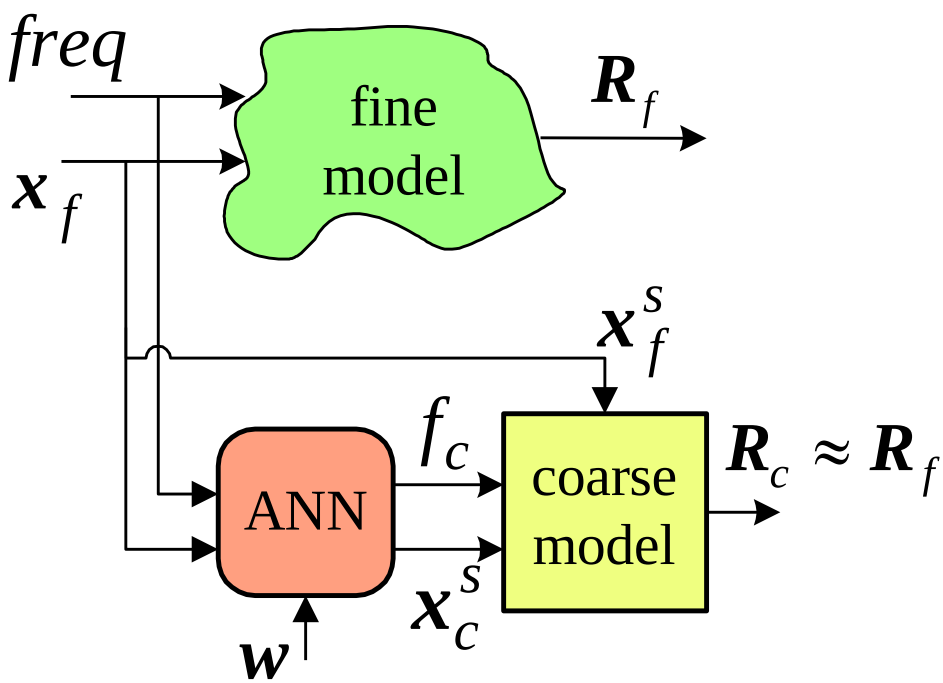
Frequency Mapped Neuromodeling (FMN) Concept



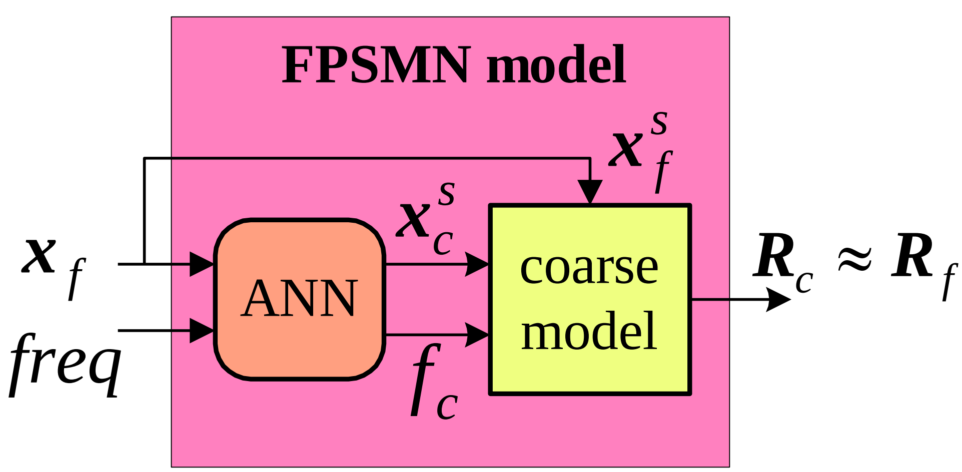
once the ANN is trained



Frequency Partial-Space Mapped Neuromodeling (FPSMN) Concept



once the ANN is trained



Training the ANN

the neuromapping can be found by solving the optimization problem

$$\min_{\boldsymbol{w}} \left\| [\boldsymbol{e}_1^T \quad \boldsymbol{e}_2^T \quad \cdots \quad \boldsymbol{e}_l^T]^T \right\|$$

$\boldsymbol{w}$ contains the internal parameters of the ANN (weights, bias, etc.) selected as optimization variables

l is the total number of learning samples

$\boldsymbol{e}_k$ is the error vector given by

for SMN

$$\boldsymbol{e}_k = \boldsymbol{R}_f(\boldsymbol{x}_{f\,i},\ freq_j) - \boldsymbol{R}_c(\boldsymbol{x}_c,\ freq_j)$$

$$\boldsymbol{x}_c = \boldsymbol{P}(\boldsymbol{x}_{f\,i})$$

for FDSMN

$$\boldsymbol{e}_k = \boldsymbol{R}_f(\boldsymbol{x}_{f\,i},\ freq_j) - \boldsymbol{R}_c(\boldsymbol{x}_c,\ freq_j)$$

$$\boldsymbol{x}_c = \boldsymbol{P}(\boldsymbol{x}_{f\,i},\ freq_j)$$

for FSMN

$$\boldsymbol{e}_k = \boldsymbol{R}_f(\boldsymbol{x}_{f\,i},\ freq_j) - \boldsymbol{R}_c(\boldsymbol{x}_c,\ f_c)$$

Training the ANN (continued)

$$\begin{bmatrix} \boldsymbol{x}_c \\ f_c \end{bmatrix} = \boldsymbol{P}(\boldsymbol{x}_{f_i},\ freq_j)$$

for FMN

$$\boldsymbol{e}_k = \boldsymbol{R}_f(\boldsymbol{x}_{f_i},\ freq_j) - \boldsymbol{R}_c(\boldsymbol{x}_{f_i},\ f_c)$$

$$f_c = P(\boldsymbol{x}_{f_i},\ freq_j)$$

for FPSMN

$$\boldsymbol{e}_k = \boldsymbol{R}_f(\boldsymbol{x}_{f_i},\ freq_j) - \boldsymbol{R}_c(\boldsymbol{x}^s_{f_i},\ \boldsymbol{x}^s_c,\ f_c)$$

$$\begin{bmatrix} \boldsymbol{x}^s_c \\ f_c \end{bmatrix} = \boldsymbol{P}(\boldsymbol{x}_{f_i},\ freq_j)$$

with

$$i = 1, \ldots, B_p$$

$$j = 1, \ldots, F_p$$

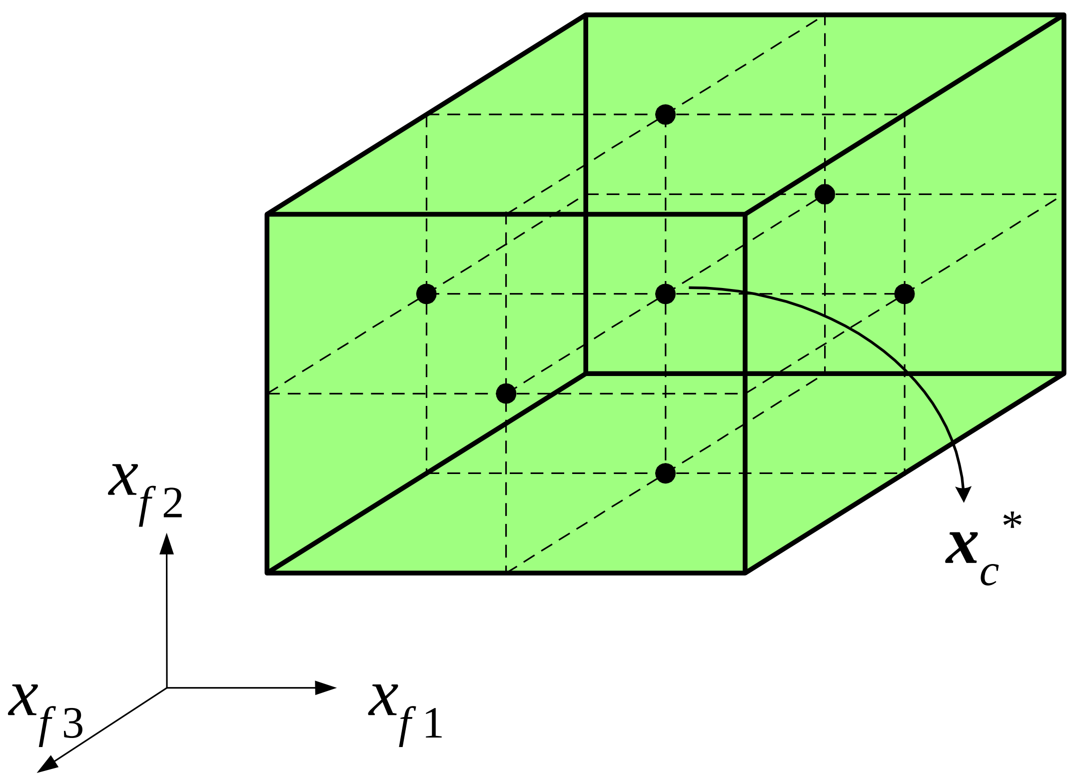
$$k = j + F_p(i-1)$$

Starting Point and Learning Samples

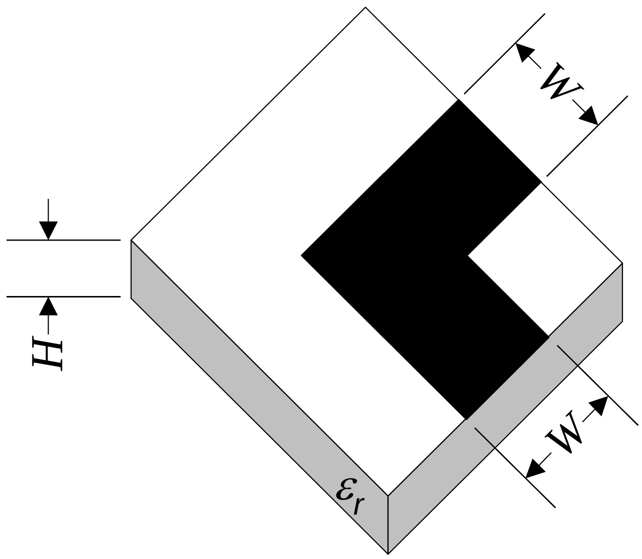
we chose a unit mapping ($\boldsymbol{x}_c \approx \boldsymbol{x}_f$ and $f_c \approx freq$) as the starting point for the optimization problem

to keep a reduced set of learning data samples, we consider an n-dimensional star distribution for the learning base points *(Bandler et al., 1989)*

the number of learning base points for a microwave circuit with n design parameters is $B_p = 2n + 1$



Microstrip Right Angle Bend



region of interest

$$20\text{mil} \le W \le 30\text{mil}$$
$$8\text{mil} \le H \le 16\text{mil}$$
$$8 \le \varepsilon_r \le 10$$
$$1\text{GHz} \le freq \le 41\text{GHz}$$

"coarse" model: Gupta model *(Gupta, Garg and Bahl, 1979)*

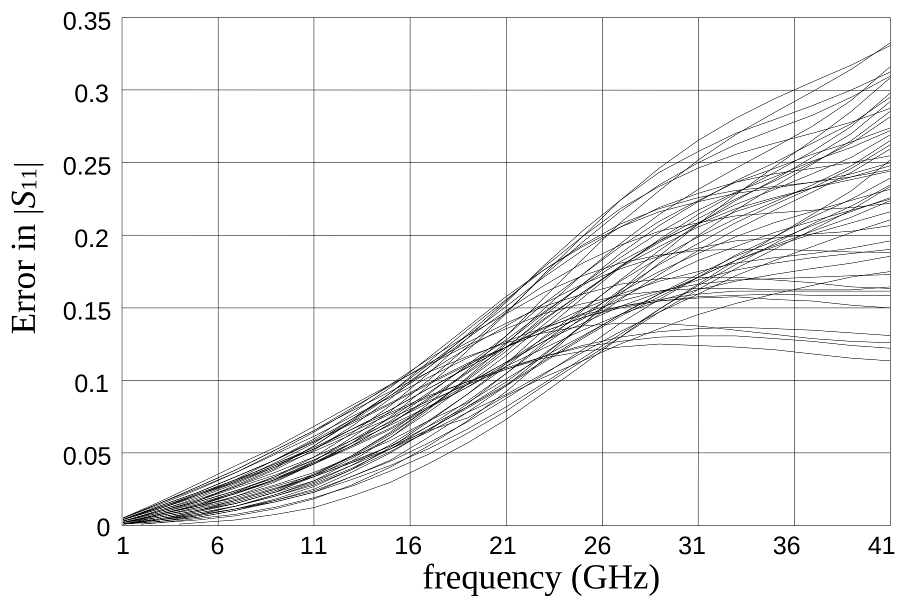
"fine" model: Sonnet's ***em***™

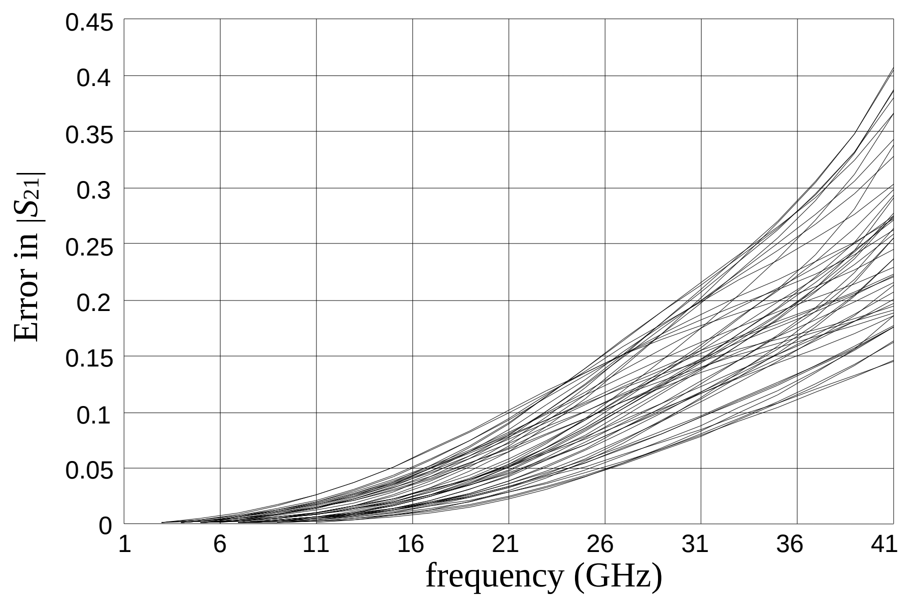
learning set: 7 base points with "star" distribution

testing set: 50 random base points in the region of interest

Microstrip Right Angle Bend Response Errors

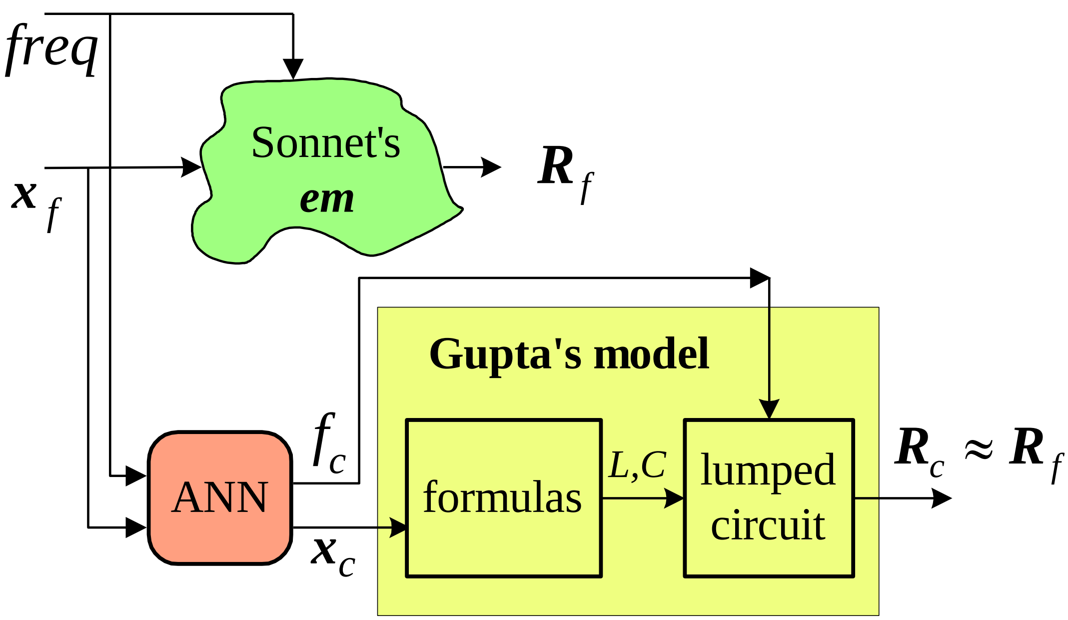
comparison before neuromodeling between ***em***™ and Gupta model at 50 random test points





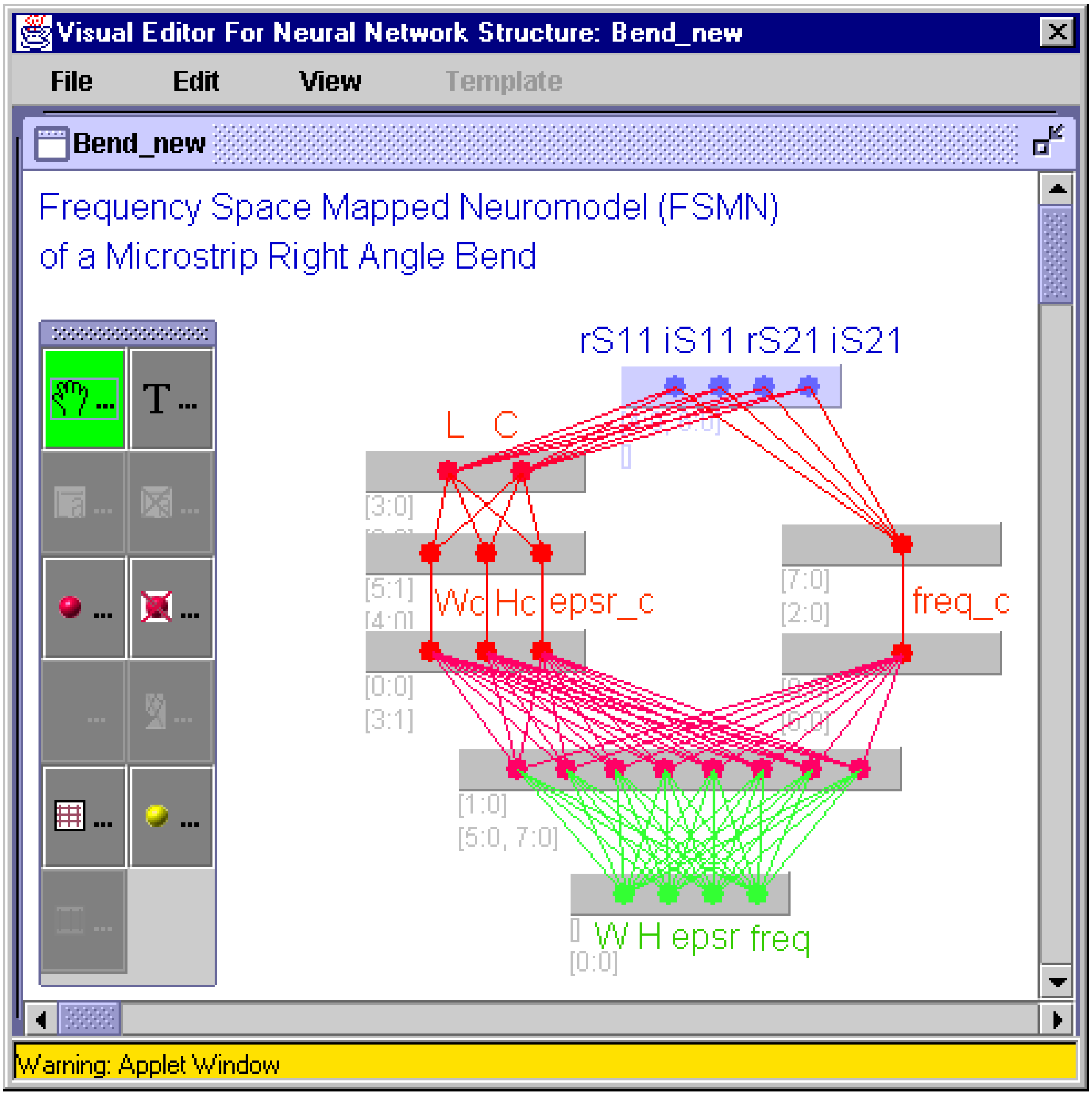
FSMN Model for the Right Angle Bend (3LP:4-8-4)

strategy implemented in *NeuroModeler*



FSMN Model for the Right Angle Bend (3LP:4-8-4)

implementation in *NeuroModeler* Version 1.2b (1999)



Implementation in *NeuroModeler*

layer one, in green, has the input parameters of the neuromapping (W, H, ε_r, and *freq*) scaled to ±1

layer two corresponds to the hidden layer of the ANN implementing the mapping (8 hidden neurons with sigmoid non-linearities)

layer three is linear and contains the coarse design parameters $\boldsymbol{x}_c$ and the mapped frequency f_c before de-scaling

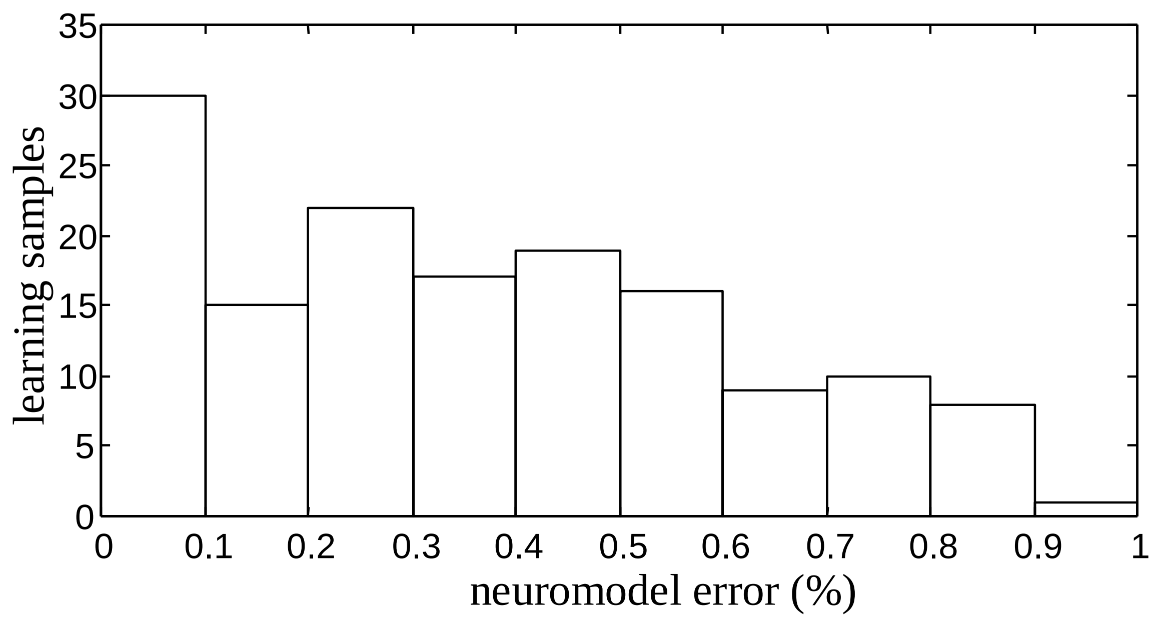
layer four de-scales the parameters

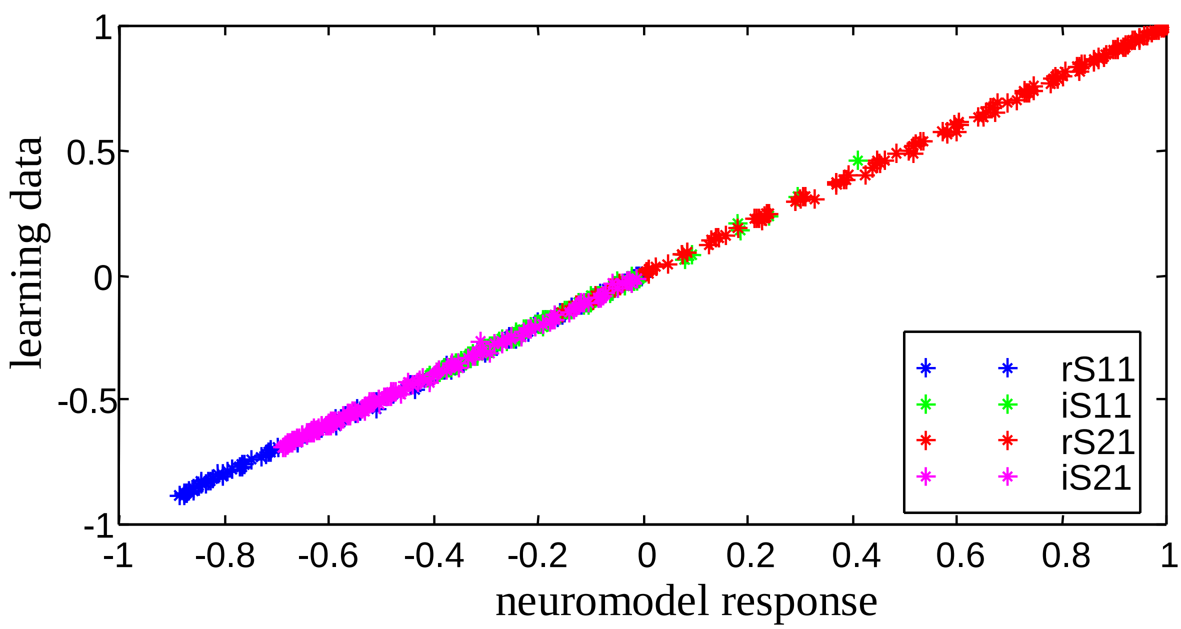
Gupta's formulas to calculate L and C are programmed as the internal analytical functions of the fifth hidden layer, using the built-in MultiSymbolicFixed function

the output layer, in blue, contains a simple internal circuit simulator that computes the real and imaginary parts of S_{11} and S_{21} for the lumped LC equivalent circuit (this layer uses the built-in CktSimulatorPS function)

FSMN Model Results for the Right Angle Bend

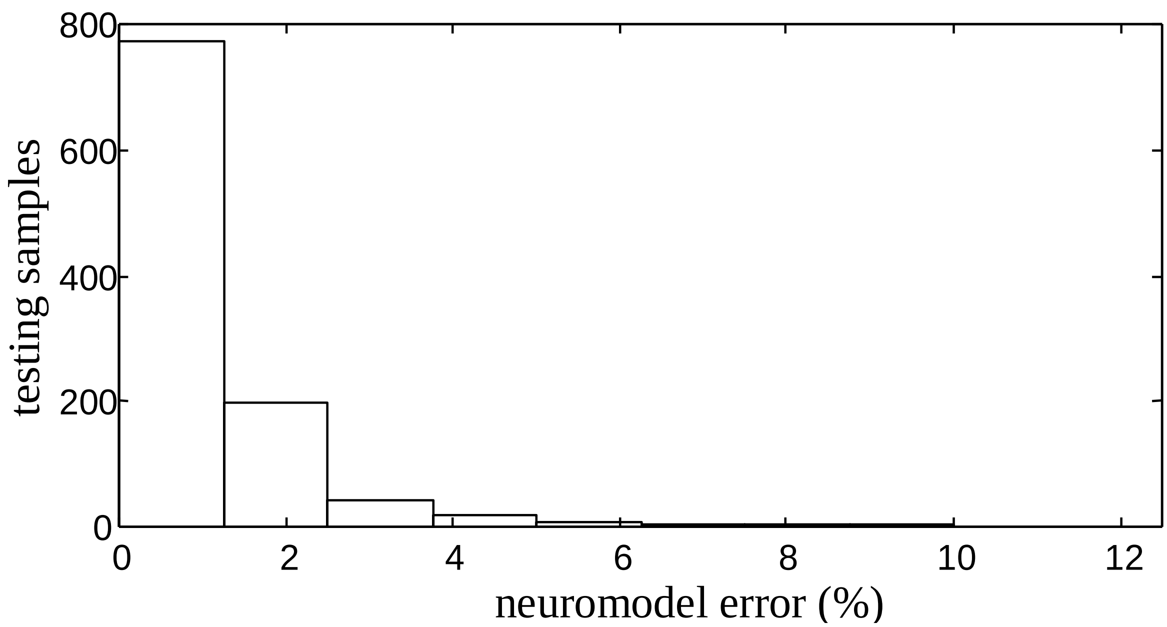
errors in the learning set after training (w.r.t. ***em***™)

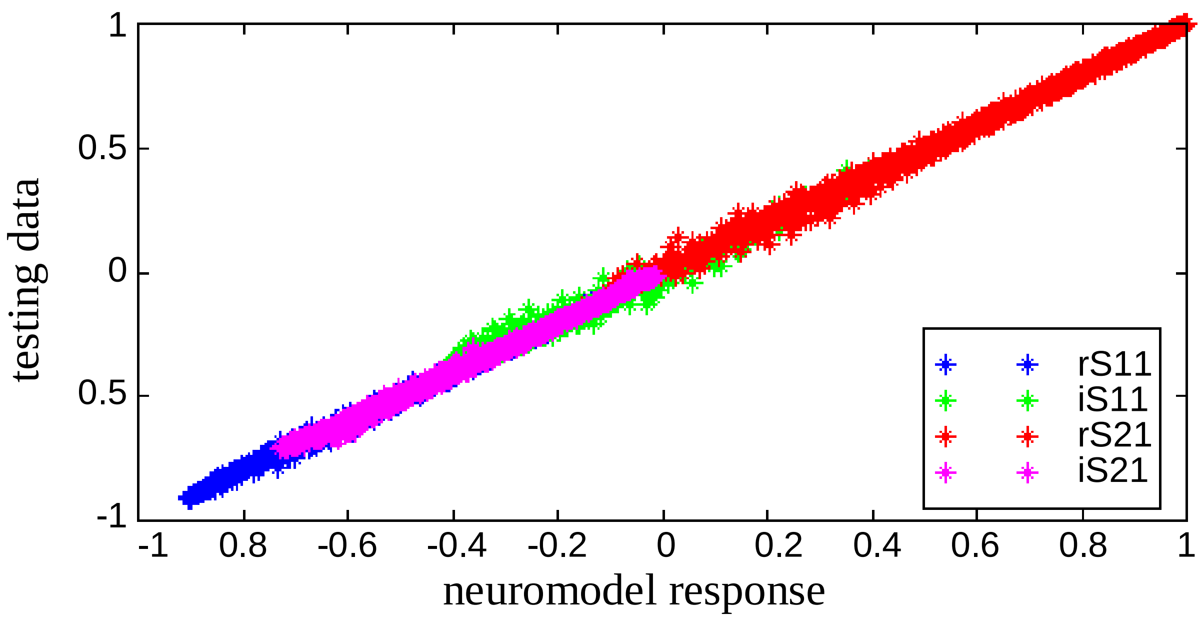




FSMN Model Results for the Right Angle Bend

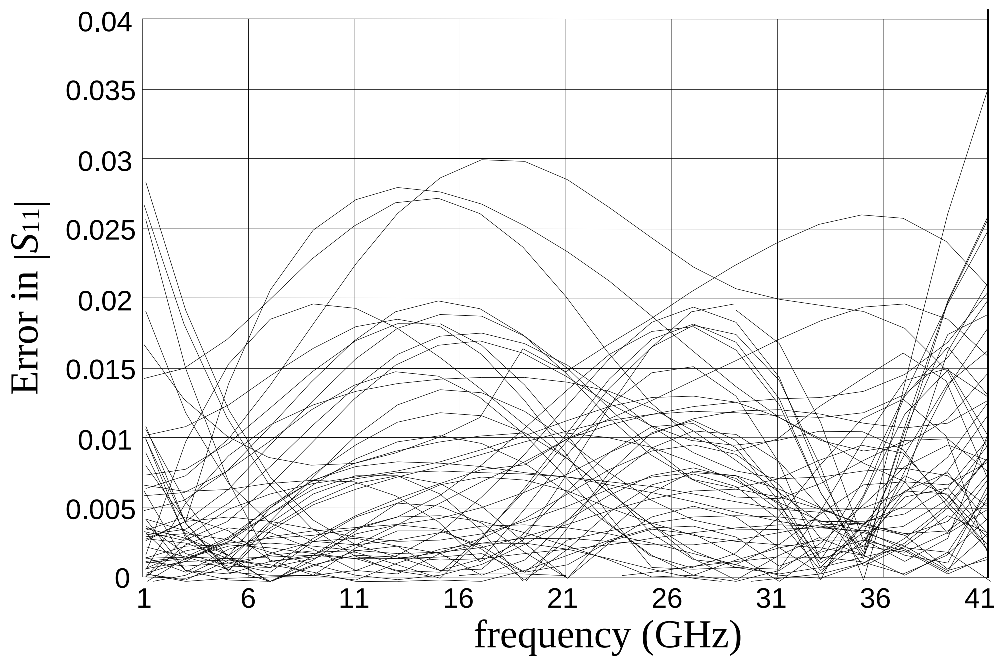
errors in the testing set after training (w.r.t. ***em***™)

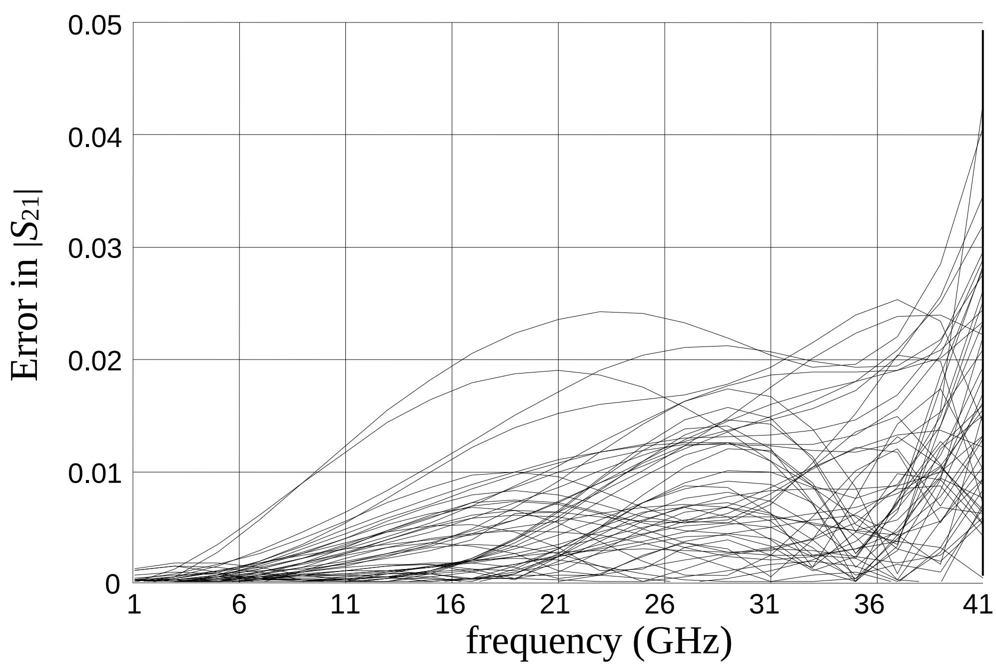




FSMN Model Results for the Right Angle Bend

comparison between ***em***™ and the FSMN model





Conclusions

we present novel realizations of Space Mapping based neuromodels of practical passive microwave components using available software

five powerful SM based neuromodeling techniques are described

these techniques

- exploit the vast set of empirical models already available
- decrease the fine model evaluations needed for training
- improve generalization ability
- reduce complexity of the ANN topology
  - w.r.t. the classical neuromodeling approach

frequency-sensitive neuromappings expand the usefulness of empirical quasi-static models

an SM based neuromodel of a microstrip right angle bend is implemented using *NeuroModeler* Version 1.2b (1999)

this model can be entered into Agilent ADS Version 1.1 (1999) as a library component through an ADS plugin module