Fast parametric models for EM design using neural networks and space mapping

Rayas-Sánchez, José E.; Zhang, Qi J.
Fast Parametric Models for EM Design Using Neural Networks and Space Mapping

José E. Rayas-Sánchez
Research Group on Computer-Aided Engineering of Circuits and Systems (CAECAS)
Department of Electronics, Systems and Informatics
Instituto Tecnológico y de Estudios Superiores de Occidente (ITESO)
Guadalajara, Mexico, 45090
http://www.desi.iteso.mx/caecas/

Q.J. Zhang
Department of Electronics, Carleton University
Ottawa, Canada K1S 5B6
http://www.doc.carleton.ca/~qjz

Outline

- Brief introduction to ANNs
- EM-based statistical analysis
- Input Space Mapping
- Linear-Input Neural-Output Space Mapping (LINO-SM)
- LINO-SM approach to yield estimation
- Constrained Broyden-Based Space Mapping
- Training the Output Neuromapping
- Example
- Conclusions
Biological Neuron

![Biological Neuron Diagram](image)

(Kartalopoulos, 1996)

Artificial Neural Networks (ANNs)

- An Artificial Neural Network (ANN) is a massively parallel distributed processor made up of simple processing units, that is able of acquiring knowledge from its environment though a learning process
- ANNs are also information processing systems that emulate biological neural networks: they are inspired in the ability of human brain to learn from observation and generalize by abstraction
Artificial Neurons

- An artificial neuron is a simple processing unit that receives and combines signals from many other neurons
- Common types of artificial neurons are:
  - Linear Neurons
  - Inner-Product Nonlinear Neuron
  - Euclidean Distance Neuron

Linear Neuron

\[ v_k = [v_{k1}, \ldots, v_{kn}]^T \text{ vector of inputs} \]
\[ w_k = [w_{k1}, \ldots, w_{kn}]^T \text{ vector of weighting factors} \]
\[ b_k \text{ bias or offset term} \]
\[ z_k \text{ activation potential or induced local field, output signal} \]
\[ z_k = b_k + v_k^T w_k \]
**Inner Product Nonlinear Neuron**

\[ s_k = b_k + v_k^T w_k \]

\[ z_k = \varphi_k(s_k) \]

- \( v_k = [v_{k1} \ldots v_{kn}]^T \) inputs \( w_k = [w_{k1} \ldots w_{kn}]^T \) weighting factors
- \( b_k \) bias or offset term \( s_k \) activation potential
- \( \varphi_k(s_k) \) activation function or squashing function
- \( z_k \) output signal

**Typical Activation Functions**

**Sigmoid or logistic function**

\[ z_k = \varphi_k(s_k) = \frac{1}{1 + e^{-s_k}} \]

With a slope parameter

\[ z_k = \varphi_k(s_k) = \frac{1}{1 + e^{-a s_k}} \]

**Hyperbolic tangent function**

\[ z_k = \varphi_k(s_k) = \tanh(s_k) = \frac{e^{s_k} - e^{-s_k}}{e^{s_k} + e^{-s_k}} \]

\( a = 1 \) (---), \( a = 0.5 \) (×) and \( a = 2 \) (——)
Some ANN Paradigms

- Multilayer Perceptrons
- Radial Basis Functions
- Recurrent Neural Networks

3-Layer Perceptrons (3LP)

\[ y = b^o + W^o F(s) \]
\[ s = b^h + W^h v \]

where
\[ W^o = [w_{1}^{oT} \cdots w_{m}^{oT}]^{T} \]
\[ W^h = [w_{1}^{hT} \cdots w_{h}^{hT}]^{T} \]
\[ F(s) = [\varphi(s_1) \cdots \varphi(s_h)]^{T} \]
\[ b^h = [b_1^h b_2^h \cdots b_h^h]^{T} \]
\[ b^o = [b_1^o b_2^o \cdots b_h^o]^{T} \]
Automated Model Generation (AMG)

EM simulators (examples used: Ansoft HFSS, Sonnet em, Faustus MEFiSTo)

User-required accuracy

AMG
- Data sampling and generation
- ANN structure adaptation
- Over-learning detection
- Under-learning detection
- Other actions as needed

Fast neural model

AMG for Spiral Inductor

Data Generators: Sonnet em Ansoft-HFSS

AMG uses data sampling algorithm to sample data at critical locations. With the same amount of training data, AMG obtains better model accuracy than conventional training techniques.

<table>
<thead>
<tr>
<th>ANN Training Technique</th>
<th>Testing Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional training</td>
<td>6.25%</td>
</tr>
<tr>
<td>AMG</td>
<td>2.34%</td>
</tr>
<tr>
<td>Advanced AMG (KAMG-PKI)</td>
<td>0.85%</td>
</tr>
</tbody>
</table>

Model accuracy under limited training data
Inverse Modeling by Neural Network

(Waveguide Filter Example, H. Kabir, Y. Wang, M. Yu and Q. Zhang, 2006)

Data Generator: Ansoft-HFSS (3D EM)

Filter Dimensions By Neural Net vs Measurement

<table>
<thead>
<tr>
<th></th>
<th>Neural Model (inch)</th>
<th>Measurement (inch)</th>
<th>Difference (inch)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O irises</td>
<td>0.405</td>
<td>0.405</td>
<td>0</td>
</tr>
<tr>
<td>M13 iris</td>
<td>0.299</td>
<td>0.297</td>
<td>-0.002</td>
</tr>
<tr>
<td>M24 iris</td>
<td>0.212</td>
<td>0.216</td>
<td>0.004</td>
</tr>
<tr>
<td>M17/M34 tuning screws</td>
<td>0.045</td>
<td>0.005</td>
<td>-0.040</td>
</tr>
<tr>
<td>M22/M33 tuning screws</td>
<td>0.133</td>
<td>0.135</td>
<td>0.002</td>
</tr>
<tr>
<td>M12/M34 coupling screws</td>
<td>0.111</td>
<td>0.115</td>
<td>0.004</td>
</tr>
<tr>
<td>Cavity length</td>
<td>1.865</td>
<td>1.864</td>
<td>-0.001</td>
</tr>
</tbody>
</table>
Time-Domain Modeling by Neural Networks

WR-28 Rectangular Waveguide Ka-band (26.5 to 40Ghz)

![Diagram showing location of conducting posts (d) controls pass band behavior]

**Data Generator:** MEFiSTo 3D Professional

**Recurrent neural network (RNN)**

**EM and RNN Transient Responses**

- **f_1**
  - d = 3.88 mm
  - d = 4.53 mm
  - d = 5.17 mm

- **f_21**
  - d = 3.88 mm
  - d = 4.53 mm
  - d = 5.17 mm

- EM test data (MEFiSTo)
  - RNN
EM-based Interpolating Surrogates
for Yield Estimation using
Neural Space Mapping Methods

EM-based Statistical Analysis

- Statistical analysis and yield prediction are crucial for manufacturability
- Reliable yield prediction typically requires massive amount of high-fidelity simulations (full-wave EM simulations)
- Performing Monte Carlo yield analysis by directly using EM simulations is not feasible for most practical problems
- Using an interpolating surrogate based on linear-input neural-output space mapping can be a solution
Input Space Mapping

\[ R_f(x_f, \psi) \]

\[ x_f \]

\[ x_c \]

\[ R_c(x_c, \psi) \]

\[ R_c(P(x_f), \psi) \]

\[ R_f(x_f, \psi) \]

\[ P \]

\[ x_c \]

\[ x_f \]

\[ x_f^{SM} \]

\[ \approx \]

\[ R_c(x_c^*, \psi) \]

\[ R_c(P(x_f^{SM}), \psi) \]

\[ Q(R_c(P(x_f), \psi), x_f, \psi, w) \]

\[ Q(R_c(Bx_f + c, \psi), x_f, \psi, w^*) = R_f(x_f, \psi) \]

for all \( x_f \) and \( \psi \) in the training region

Linear-Input Neural Output Space Mapping

Fast Parametric Models for EM Design Using Neural Networks and Space Mapping
Q.J. Zhang, Carleton University, Canada; José E. Rayas-Sánchez, ITESO, Mexico
LINO-SM approach to Yield Estimation

Start

Obtain $x_c^*$ by optimizing the coarse model

Calculate $x_f^{SM}$ and $P$ through input space mapping

Generate learning and testing data around $x_f^{SM}$ according to tolerances

Obtain $Q$ by training an output neuro mapping

Calculate yield using $Q(R_c(P(x_f,\psi),x_f,\psi))$

End

Constrained Broyden-Based SM

Begin
find $x_c^*$ solving (1)
i = 0, $x_f^{(i)} = x_c^*$, $B^{(i)} = I$, $\delta = 0.3$
$f^{(i)} = P(x_f^{(i)}) - x_c^*$ using (2)
repeat until stopping_criterion
solve $B^{(i)}h^{(i)} = -f^{(i)}$ for $h^{(i)}$
$x_f^{(test)} = x_f^{(i)} + h^{(i)}$
while $x_f^{(test)} < x_f^{\min} \lor x_f^{(test)} > x_f^{\max}$
$h^{(i)} = \delta h^{(i)}$
x_f^{(test)} = x_f^{(i)} + h^{(i)}$
end
$x_f^{(i+1)} = x_f^{(test)}$
f^{(i+1)} = P(x_f^{(i+1)}) - x_c^*$ using (2)
$B^{(i+1)} = B^{(i)} + \frac{f^{(i+1)}h^{(i)}f^{(i)} - h^{(i+1)}f^{(i)}} {h^{(i)}f^{(i)}h^{(i)}}, i = i + 1$
end

(1) $x_c^* = \arg \min_{x_c} U(R_c(x_c,?))$

(2) $P(x_f) = \arg \min_{x_c} \begin{bmatrix} x_1^T & \ldots & x_2^T \end{bmatrix}$
$\epsilon_j(x_f) = R_j(x_f,?) - R_j(x_c,?)$

$x_f^{SM} = x_f^{(i)}$
P(x_f) = Bx_f + \epsilon$
where $B = B^{(i)}$ and $\epsilon = x_c^* - Bx_f^{SM}$
Generating Learning and Testing Points

- learning base point
- testing base point

2n+1 learning base points in a star distribution
2n testing base points in a rotated star distribution

Training the Output Neuro Mapping

Begin
Generate \( R_{CL}, R_{CT}, R_{FL}, \) and \( R_{FT} \)
\[ \varepsilon_L^{old} = \| R_{CL} - R_{FL} \|_F, \quad \varepsilon_T^{old} = \| R_{CT} - R_{FT} \|_F \]
\( h = m, i = 1 \)
\[ \omega^{(i)} = \arg \min_{\omega} \| E_L(\omega) \|_F \]
\[ \varepsilon_L = \| Q_L(\omega^{(i)}) - R_{FL} \|_F \]
\[ \varepsilon_T = \| Q_T(\omega^{(i)}) - R_{FT} \|_F \]
while \( \varepsilon_T^{old} \geq \varepsilon_T \lor \varepsilon_L^{old} \geq \varepsilon_T \)
\[ \varepsilon_T^{old} = \varepsilon_T, \quad \varepsilon_L^{old} = \varepsilon_L, \quad i = i + 1, \quad h = h + 1 \]
\[ \omega^{(i)} = \arg \min_{\omega} \| E_L(\omega) \|_F \]
\[ \varepsilon_L = \| Q_L(\omega^{(i)}) - R_{FL} \|_F \]
\[ \varepsilon_T = \| Q_T(\omega^{(i)}) - R_{FT} \|_F \]
end

\[ w^* = \omega^{(i-1)} \]

\( E_L(\omega) = R_{FL} - Q_L(\omega) \)

for all \( x_f \) and \( \omega \) in the training region
Microstrip Notch Filter

\[ H = 10\text{mil} \]
\[ W_{50} = 31\text{mil} \]
\[ \varepsilon_r = 2.2 \]
\[ \text{loss tan} = 0.0009 \]

(RT Duroid 5880)

\[ x_j = [L_c, L_o, S_g]^T \]

Specifications:
\[ |S_{21}| \leq 0.05 \text{ for } 13.19\text{GHz} \leq f \leq 13.21\text{GHz} \]
\[ |S_{21}| \geq 0.95 \text{ for } f \leq 13\text{GHz} \text{ and } f \geq 13.4\text{GHz} \]

Microstrip Notch Filter – Fine Model

\[ H_{\text{air}} = 60 \text{ mil} \]
\[ L_p = \frac{1}{2}(L_o + L_c) \]
\[ Y_{\text{gap}} = L_o \]
\[ \text{grid} = 0.5\text{mil} \times 0.5\text{mil} \]
Microstrip Notch Filter – Coarse Model

Optimization Variables:
- var Lc 143mil
- var Lo 158mil
- var Sg 8mil

Preassigned Parameters:
- var W50 31mil
- var Lp 31mil
- var Lzero 1e-9mil

- MSub RT-Duroid-5880
- H=10mil
- ER=2.2
- TAND=0.0009
- LEVEL=2

- W=W50 L=Lp
- W=W50 L=Lc
- W=W50 L=Lo

Microstrip Notch Filter – Starting Point

\[ x^* = [143, 158, 8]^T \text{ (mil)} \]

\[ R_c(x^*) \]

\[ R_f(x^*) \]

frequency (GHz)

Fast Parametric Models for EM Design Using Neural Networks and Space Mapping
Q.J. Zhang, Carleton University, Canada; José E. Rayas-Sánchez, ITESO, Mexico
Microstrip Notch Filter – SM Solution

Microstrip Notch Filter – Training $Q$

Fast Parametric Models for EM Design Using Neural Networks and Space Mapping
Q.J. Zhang, Carleton University, Canada; José E. Rayas-Sánchez, ITESO, Mexico
Fast Parametric Models for EM Design Using Neural Networks and Space Mapping
Q.J. Zhang, Carleton University, Canada; José E. Rayas-Sánchez, ITESO, Mexico
Microstrip Notch Filter – LINOSM Yield

![Graph showing yield versus frequency](image)

**Conclusions**

- We described a method for highly accurate EM-based statistical analysis and yield estimation of RF and microwave circuits.
- It consists of applying a constrained Broyden-based linear-input space mapping, followed by a neural-output space mapping, in which the responses, the design parameters and independent variable are mapped.
- The output neuromodel is trained using reduced sets of learning and testing samples.
- The resultant interpolating surrogate model is used as a very efficient vehicle for accurate statistical analysis and yield prediction.