Machine Learning Applied to the Blind Identification of Multiple Delays in Distributed Systems

Felipe Treviso, Riccardo Trinchero, Flavio G. Canavero
Dept. Electronics and Telecommunications, Politecnico di Torino, Italy
felipe.treviso@polito.it
https://emc.polito.it
Electrical interconnects are responsible for a considerable part of signal degradation [1].

- **Distortion** (bits propagate and get distorted)
- **Reflection** (bits bounce back at all discontinuities)
- **Crosstalk** (the transmitted bits appear also on the other traces)

**Interconnect models** are essential to predict **signal integrity** of the channel during the **design phase** (via simulations) without requiring expensive prototyping→ **We need a model!!!!**
Usually, interconnects are characterized by tabulated frequency data obtained from electromagnetic simulations or measurements.

Why do we need a model?
We already have a characterization of the linear interconnects.
The link is made of *wires/PCB traces (linear)*, while the *terminations* contain drivers, receivers, LDO and other *nonlinear components*.

Due to *nonlinear* elements, *signal integrity simulations* must be carried out in *time-domain*.

*Interconnect models* obtained from *frequency-domain data* should also be compatible with *time-domain circuit simulations*. 
Rational Models are naturally adopted to model linear structures

\[ H(j\omega) \approx \tilde{H}(j\omega) = \sum_{j=1}^{n_p} \frac{r_j}{j\omega - p_j} + r_0 \]

- It is a linear expansion of rational basis functions
  - Linear w.r.t. residues and nonlinear w.r.t. poles → iterative pole reallocation is used to select the optimal poles
  - An equivalent circuital representation is available

N.B. 1 pole = 1 dynamic element in the circuit (capacitor/inductor)
Circuital models sought for should be **accurate and fast to simulate!!!**
Supposing a simulation of 100 interconnects that need 161 poles each to achieve an accurate model:

- 100 x 161 poles = 16,100 poles
- 16,100 dynamic elements in the simulation (capacitors/inductors)
- 16,100 additional states in the system

If the model requires too many poles, its simulation will be inefficient!!!
Delayed Rational Model (DRM)

\[ \tilde{H}(j\omega) = \sum_{i=1}^{n_T} \sum_{j=1}^{n_p} r_{ij} + \sum_{i=1}^{n_T} e^{-j\omega \tau_i} + r_0 \]

- **Advantages:**
  - Generally a **lower number** of poles w.r.t. the RM is required → **faster simulation time**
  - **Causality** of the system is guaranteed by making \( \tau_i > 0 \)
  - **Linear** with respect to the residues
  - **Explicit representation** of the **delayed behavior** of the transfer function

- **Disadvantages:**
  - **Unpractical** to estimate both the poles and the delays together → usually delays are estimated first and poles afterwards
  - Generally requires **optimization** of the parameters to obtain a good model

Refs. [3]-[5]
How to find the appropriate poles and delays?

- Let us consider, as a **test function**, the transfer function
  \[ H(j\omega) = \frac{1}{j\omega + 3} e^{-j3\omega} \]

- We can **build a DRM** with **poles and delays** chosen on a **grid** in a \( p - \tau \) plane (\( p \) is restricted to be real, for the sake of visualization)

- The **nodes** of the grid provide the candidate **poles and delays** to be considered with the **delayed rational model**:

  \[ \tilde{H}(j\omega) = \sum_{i=1}^{n_p} \sum_{j=1}^{n_\tau} \frac{r_{ij}}{j\omega - p_{ij}} e^{-j\omega \tau_{ij}} + r_0 \]

  - To be estimated
  - From the grid
  - Linear parameter
  - Nonlinear parameters

  Basis functions:
  \[ \varphi(\omega; p, \tau) = \frac{c_{ij}}{j\omega - p_{ij}} e^{-j\omega \tau_{ij}} \]
Let us try to fit \( H(j\omega) = \frac{1}{j\omega+3} e^{-j3\omega} \) by making a grid in a \( p - \tau \) plane:

- If the grid captures **exact pole and delay**
  - (a) - Approx. model is **essentially perfect**
Let us try to fit $H(j\omega) = \frac{1}{j\omega + 3}e^{-j3\omega}$ by making a grid in a $p - \tau$ plane:

- If the grid captures **only the exact delay**
  - (b) – Approx. model is still **very good**
Let us try to fit \( H(j\omega) = \frac{1}{j\omega+3} e^{-j3\omega} \) by making a grid in a \( p - \tau \) plane:

- If the grid captures only the exact pole
  - (c) – Approx. model is clearly inaccurate
Let us try to fit $H(j\omega) = \frac{1}{j\omega+3}e^{-j3\omega}$ by making a grid in a $p - \tau$ plane:

Three cases considered:

- (a) – Approx. model is **essentially perfect**
- (b) – Approx. model is still **very good**
- (c) – Approx. model is clearly **inaccurate**

A larger number of poles can compensate a non-exact estimation of the poles, but a wrong delay estimation generates an inaccurate model, even if it uses the right poles.

An **accurate delay estimation** is essential to obtain an **accurate delayed rational model**
The DRM should contain the **exact delay** of the transfer function it approximates.

\[ H(j\omega) = \sum_{i=1}^{\infty} \sum_{j=1}^{n_{p,i}} r_{ij} e^{-j\omega \tau_i} + r_0 \]

The only way to ensure that an unknown delay is included in the model is by considering an infinite number of delays. How can we estimate a model with an **infinite number of terms**?
Support Vector Machines (SVMs) [6][7]

Model:

\[ \hat{H}(j\omega) = \sum_{k=1}^{K} \alpha_k k(\omega, \omega_k) + b \]

- Kernel is linked to a vector with the basis functions of a regression model → vector can be infinite dimensional!!! [6]

Historical applications:

- Handwriting recognition
- Face detection

Inner product

\[ k(\omega, \omega_k) = \langle \varphi(\omega; p, \tau), \varphi^*(\omega_k; p, \tau) \rangle \]

Handwriting recognition image

Face detection image
The Least Squares Support Vector Machine (LS-SVM) regression has two equivalent formulations [7]:

**Dual Space**

\[
\tilde{H}(j\omega) = \sum_{k=1}^{K} \alpha_k k(\omega, \omega_k) + b
\]

\[
w = \sum_{k} \alpha_k \varphi^*(\omega_k)
\]

**Primal Space:**

\[
\hat{H}(j\omega) = \langle w, \varphi(\omega; p, \tau) \rangle + b
\]

\[
\tilde{H}(j\omega) = \sum_{i,j} w_{ij} \varphi(\omega; p_{ij}, \tau_i) + b
\]

- **Non-parametric model** → number of terms equal to the number of samples
- **Parametric model** → number of terms equal to the number of basis functions
Weights \( w \) are proportional to the residues of a delayed-rational model.

**ML model:**
\[
\tilde{H}(j\omega) = \sum w(\tau_i, p_{ij}) \varphi(\omega, p_{ij}, \tau_i) + b
\]

**DRM:**
\[
\tilde{H}(j\omega) = \sum_{i=1}^{n_\tau} \sum_{j=1}^{n_{p,i}} \frac{r_{ij}}{j\omega - p_{ij}} e^{-j\omega\tau_i} + r_0
\]

By looking at the values of \( w \) as a function of \( \tau \), we are able to see for which values of \( \tau \) the \( w \) is larger, i.e., the dominant propagation delays of the system.
The identified propagation delays can be employed to build **low-order delayed rational models**

\[
\tilde{H}(j\omega) = \sum_{i=1}^{n_\tau} \sum_{j=1}^{n_{p,i}} \frac{r_{ij}}{j\omega - p_{ij}} e^{-j\omega \tau_i} + r_0
\]
Known transfer function:

\[ H(j\omega) = \left( \frac{1}{j\omega + 60 + 20j} + \frac{1}{j\omega + 60 - 20j} \right) e^{-0.1\omega} + \frac{0.075}{j\omega + 100} e^{-0.3\omega} \]

- **Real** and complex-conjugate poles
- **First delay** is the same as identified by Hilbert transform [8]
- Overall curve is like the one obtained by the Gabor transform [3]
- Circuit with **multiple transmission lines**:
  
  \[ H(j\omega) = \frac{V_o(j\omega)}{V_i(j\omega)} \]

- Three paths from input to output, **wave reflection** at the **discontinuities**

- **Multiple delays** expected in \( H(j\omega) \)

- **Accuracy similar** to Hilbert (1\(^{\text{st}}\) delay) and Gabor transforms (overall shape)
Circuit with multiple transmission lines:

\[ H(j\omega) = \frac{V_o(j\omega)}{V_i(j\omega)} \]

- Three paths from input to output, wave reflection at the discontinuities
- Multiple delays expected in \( H(j\omega) \)
- Accuracy similar to Hilbert (1st delay) and Gabor transforms (overall shape)
- Good performance with noisy data
Circuit with multiple transmission lines:

- Five delays and 20 poles distributed among them are sufficient to accurately model the original transfer function.
- A rational model with similar accuracy requires 54 poles.

<table>
<thead>
<tr>
<th></th>
<th>Proposed model</th>
<th>VF model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error - $L_2$-norm</td>
<td>0.3344</td>
<td>0.5200</td>
</tr>
<tr>
<td>Error - $L_{\infty}$-norm</td>
<td>0.0261</td>
<td>0.0546</td>
</tr>
<tr>
<td>Order</td>
<td>20 total poles</td>
<td>54 total poles</td>
</tr>
</tbody>
</table>
SpaceWire (SpW) cable link:

Scattering matrix of two wires of the link is considered:

\[ S(j\omega) = \begin{bmatrix} S_{1,1}(j\omega) & S_{1,2}(j\omega) \\ S_{2,1}(j\omega) & S_{2,2}(j\omega) \end{bmatrix} \]

Cable channel linking a driver and a receiver through:

- Striplines
- 9-pin Micro-D connectors
- SpW cable
  - 4 twisted pairs of wires
  - 1 inner shield around each of the pairs
  - 1 outer shield

Credit: Rick Mastracchio/NASA/Twitter
SpaceWire cable link:
SpaceWire cable link:

![Diagram of SpaceWire cable link]

**Application Examples – III (S_{2,2})**

- **SpaceWire** cable link:

  ![Diagram of SpaceWire cable link]

  - **S_{2,2}**
  - **τ = 0.5001 ns**
  - **τ = 100.02 ns**
  - **τ = 111.52 ns**
  - **τ = 111.53 ns**

  ![Graph of magnitude and phase](graph.png)

  - **Original transfer function**
  - **Approximation (30 total poles)**

  **Magnitude**

  ![Magnitude graph](magnitude_graph.png)

  **Phase (degrees)**

  ![Phase graph](phase_graph.png)

  **Frequency (Hz)**

  ![Frequency graph](frequency_graph.png)
**SpaceWire** cable link:

- **Application Examples – III** ($S_{1,2}$)

- **Figure:**
  - **Graph:**
    - **Axes:**
      - Horizontal: Frequency (Hz) × 10^8
      - Vertical: Magnitude
    - **Curves:**
      - Original transfer function
      - Approximation (25 total poles)
    - **Marker:**
      - Points of interest

- **Key Points:**
  - Delay times:
    - $\tau = 56.008$ ns
    - $\tau = 53.508$ ns
    - $\tau = 158.52$ ns
    - $\tau = 41.506$ ns
  - Frequency range:
    - 1 to 10 GHz
Application Examples – III (Summary)

SpaceWire cable link:

- All the delayed-rational models built with the identified delays require less poles than a pure rational model with similar accuracy.

- Kernel depends only on frequency points, chosen poles, $\tau_m$ and $\tau_M$:

$$k(\omega, \omega_k; p, \tau_m, \tau_M)$$

- Same kernel for all the 3 terms of the matrix!

Around 5x less poles!
Conclusions

- **Delayed-rational models** allow reducing the complexity of models of distributed systems. Examples showed a reduction of 2.5-5 times in the total number of poles when comparing with rational models.

- **ML kernel-based regression** (e.g., Least-Squares Support Vector Machine (LS-SVM)) can be adopted for the estimation of the dominant delays in distributed systems.

- The **LS-SVM approach** provides a very accurate identification of the network delays (comparable with Hilbert transform – when applicable – and with Gabor transform method for multiple delays), and generates a rational approximation with a number of poles significantly reduced w.r.t. conventional fitting methods.

N.B: The proposed methodology for the delay estimation is extremely flexible, i.e., poles and delay interval can be changed as the knowledge about the system increases. E.g., the model can consider multiple delay intervals.
Thank you very much for the attention!

Questions?


