On Automated Generation of Behavioral Parameterized Macromodels

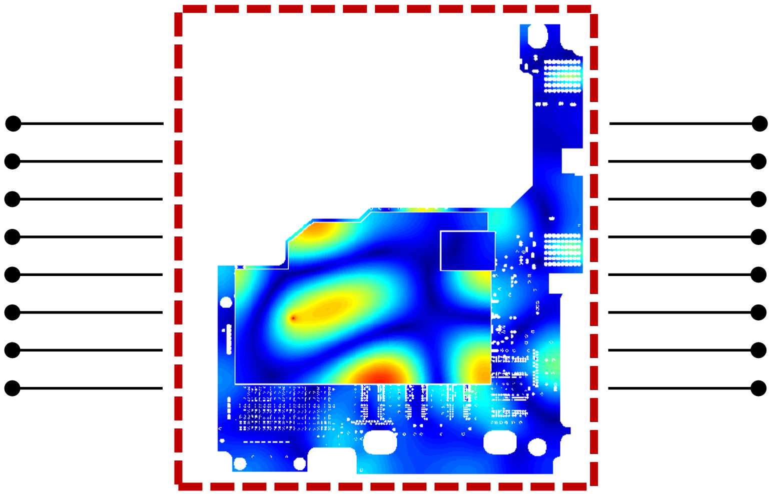
Part I: Algorithmic aspects

T. Bradde, S. Grivet-Talocia, M. De Stefano, A. Zanco

Dept. of Electronics and Telecommunications
Politecnico di Torino, Italy

European IBIS Summit, 25/05/2018, Brest, France

The Starting Point: Macromodeling



Macromodeling techniques

$$\mathbf{H}(s) = \frac{\mathbf{N}(s)}{\mathrm{D}(s)}$$

$$\dot{\boldsymbol{w}} = \boldsymbol{A}\boldsymbol{w} + \boldsymbol{B}\boldsymbol{u}$$
$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{w} + \boldsymbol{D}\boldsymbol{u}$$

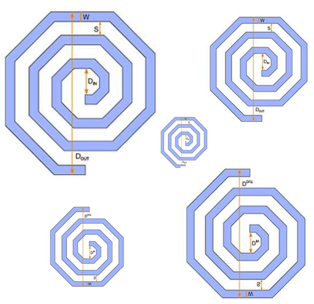
Gather I/O data from measurements or Maxwell equation solution → Obtain a transfer function representation (Usually in terms of S parameters) → Get the state space and a SPICE netlist

Advantage: allows to perform fast time-domain transient simulations that includes non-linear components in circuit simulation software

Example of Application: signal and power integrity analysis of RF systems

A Step Further: Parameterized Macromodeling

Our Goal: to obtain compact macromodels able to describe the behavior of the system with respect to parameters variations (geometric quantities, temperature, device operating point…)



EM Simulations

Parameterized macromodeling techniques

$$\dot{\boldsymbol{w}} = \boldsymbol{A}(\boldsymbol{\vartheta})\boldsymbol{w} + \boldsymbol{B}(\boldsymbol{\vartheta})\boldsymbol{u}$$
$$\boldsymbol{y} = \boldsymbol{C}(\boldsymbol{\vartheta})\boldsymbol{w} + \boldsymbol{D}(\boldsymbol{\vartheta})\boldsymbol{u}$$

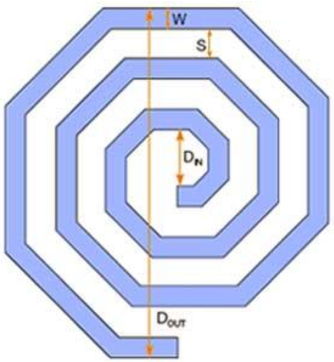
$$\Updownarrow$$

$$\mathbf{H}(s) = \frac{\mathrm{N}(s;\boldsymbol{\vartheta})}{\mathrm{D}(s;\boldsymbol{\vartheta})}$$

Model defined over a **continuous parameter interval**

Advantages: Perform circuit design optimization and worst-case scenario analysis faster and directly in circuit simulation software

Parameterized Macromodeling



Parametric sweep

Simulation or Measurement

Multiple S-parameters

$$\breve{\mathbf{H}}_{k;m} = \breve{\mathbf{H}}(j\omega_k; \boldsymbol{\vartheta}_m)$$
$$k = 1, \dots, K; \; m = 1, \dots, M$$

IN

.snp

Multivariate Rational Fitting

Stability, Passivity Enforcement

$$\mathbf{H}(s) = \frac{\mathbf{N}(s; \boldsymbol{\vartheta})}{\mathrm{D}(s; \boldsymbol{\vartheta})}$$

Parameterized model

State-space realization

Circuit synthesis

OUT

SPICE netlist

$$\dot{\boldsymbol{w}} = \boldsymbol{A}(\boldsymbol{\vartheta})\boldsymbol{w} + \boldsymbol{B}(\boldsymbol{\vartheta})\boldsymbol{u}$$
$$\boldsymbol{y} = \boldsymbol{C}(\boldsymbol{\vartheta})\boldsymbol{w} + \boldsymbol{D}(\boldsymbol{\vartheta})\boldsymbol{u}$$

Rational Fitting

Fix the model structure

$$\mathbf{H}(s;\boldsymbol{\vartheta}) = \frac{\mathbf{N}(s;\boldsymbol{\vartheta})}{\mathrm{D}(s;\boldsymbol{\vartheta})} = \frac{1}{\mathrm{D}(s;\boldsymbol{\vartheta})}\begin{pmatrix} \mathrm{N}_{11}(s;\boldsymbol{\vartheta}) & \cdots & \mathrm{N}_{1\mathrm{n}}(s;\boldsymbol{\vartheta}) \\ \vdots & \ddots & \vdots \\ \mathrm{N}_{\mathrm{m}1}(s;\boldsymbol{\vartheta}) & \cdots & \mathrm{N}_{\mathrm{mn}}(s;\boldsymbol{\vartheta}) \end{pmatrix};$$

$$\mathrm{N}_{\mathrm{ij}}(s;\boldsymbol{\vartheta}) = \boldsymbol{a_0}(\boldsymbol{\vartheta}) + \boldsymbol{a_1}(\boldsymbol{\vartheta})\boldsymbol{s} + \cdots + \boldsymbol{a_m}(\boldsymbol{\vartheta})\boldsymbol{s^m}$$

$$\mathrm{D}(s;\boldsymbol{\vartheta}) = \boldsymbol{b_0}(\boldsymbol{\vartheta}) + \boldsymbol{b_1}(\boldsymbol{\vartheta})\boldsymbol{s} + \cdots + \boldsymbol{b_{n-1}}(\boldsymbol{\vartheta})\boldsymbol{s^{n-1}} + \boldsymbol{s^n}$$

Enforce the fitting

$$\breve{\mathbf{H}}_{k;m} = \breve{\mathbf{H}}(s_k;\boldsymbol{\vartheta}_m); \quad for \quad k = 1,\ldots,K;\ m = 1,\ldots,M$$

$$\min \left\| \frac{\mathbf{N}(s_k;\boldsymbol{\vartheta}_m) - \mathrm{D}(s_k;\boldsymbol{\vartheta}_m)\breve{\mathbf{H}}(s_k;\boldsymbol{\vartheta}_m)}{\mathrm{D}(s_k;\boldsymbol{\vartheta}_m)} \right\|_2^2$$

NON-LINEAR LS PROBLEM!

Parameterized Sanathanan-Koerner

Model structure: $\mathbf{H}(s;\boldsymbol{\vartheta}) = \frac{\mathbf{N}(s;\boldsymbol{\vartheta})}{\mathrm{D}(s;\boldsymbol{\vartheta})} = \frac{\sum_n \sum_l \mathbf{R}_{n,l}\,\xi_l(\boldsymbol{\vartheta})\varphi_n(s)}{\sum_n \sum_l r_{n,l}\,\xi_l(\boldsymbol{\vartheta})\varphi_n(s)}$

Unknown Coefficients to be found

Basis Functions:

$\varphi_n(s)$: Partial Fractions

$\xi_l(\boldsymbol{\vartheta})$: Chebychev Polynomials

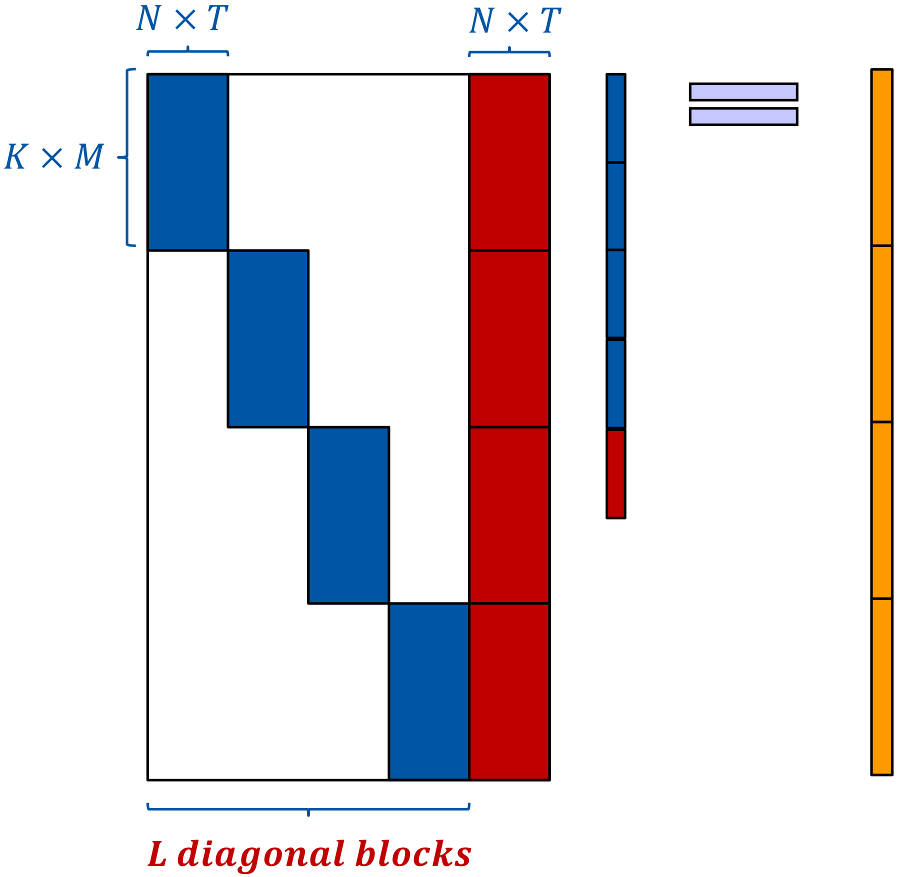
PSK scheme: $D_0 = 1$; for $\mu = 1,2,\ldots$

$$\min \left\| \frac{\mathbf{N}_\mu(s_k;\boldsymbol{\vartheta}_m) - \mathrm{D}_\mu(s_k;\boldsymbol{\vartheta}_m)\breve{\mathbf{H}}(s_k;\boldsymbol{\vartheta}_m)}{\mathrm{D}_{\mu-1}(s_k;\boldsymbol{\vartheta}_m)} \right\|$$

end

Known!

PSK Least Squares Problem



$K \times M$: Available data samples for each response

L: Number of responses

N: Cardinality of partial fractions

T: Cardinality of parameter basis

Denominator quantities

Numerator quantities

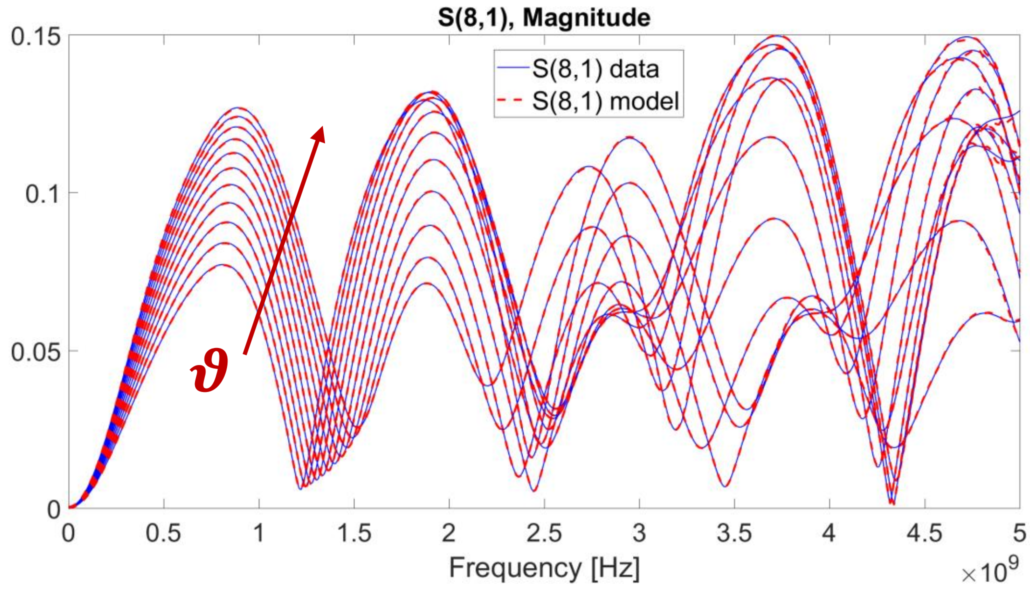
In white: Zeros

Memory requirements: $E_{PSK} = KMLNT(L+1)$

Flops requirements: $Fl_{PSK} \propto KMN^2T^2L^3$ → **CUBIC!**

An Example: 4-Port System

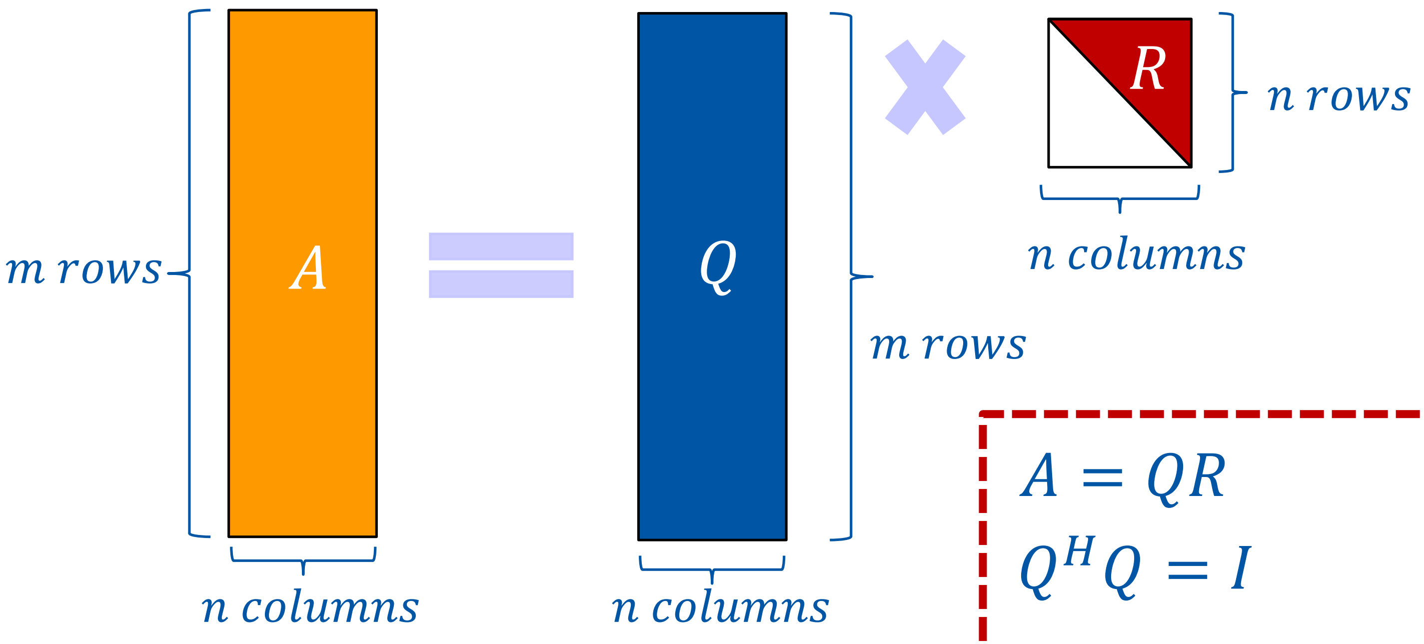
Accuracy looks ok but…



10 PSK iteration required 2140 seconds on a server machine (2.2 GHz) and 2.63 Gb of memory!

Deus Ex Algebra: QR factorization

We will exploit the reduced formulation…



The QR factorization finds an orthonormal basis (Q columns) that span the column space of A through the coefficients embedded in R

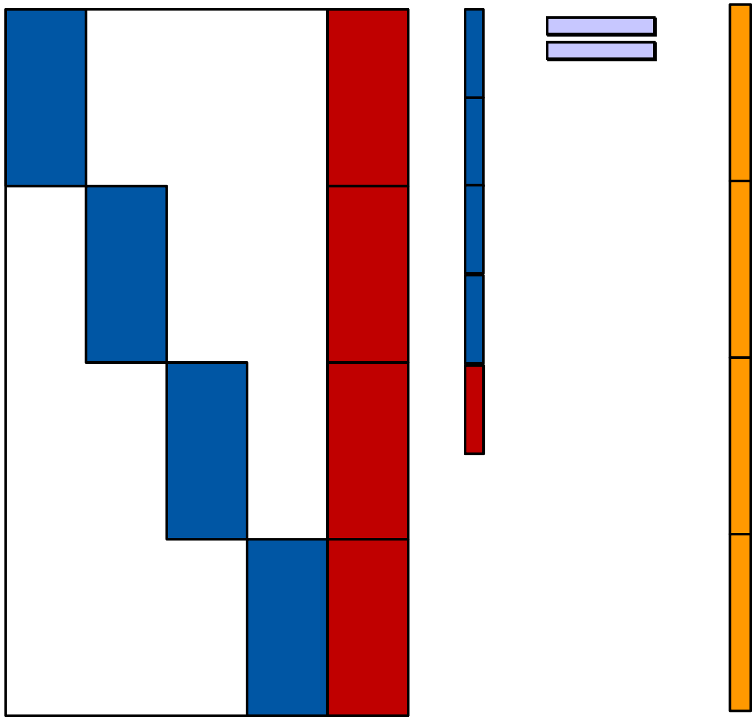
$$A = QR$$

$$Q^H Q = I$$

In our case: $m \gg n$

R is a small upper triangular matrix

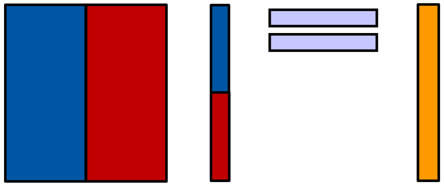
System Decoupling



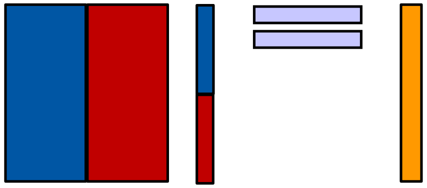
We start from the PSK LS system…

$$Ax = b$$

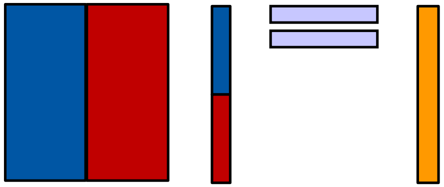
System Decoupling



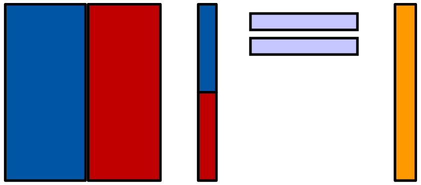
$$A_1 \begin{pmatrix} x_1 \\ x_d \end{pmatrix} = \widetilde{h_1}$$



$$A_2 \begin{pmatrix} x_2 \\ x_d \end{pmatrix} = \widetilde{h_2}$$



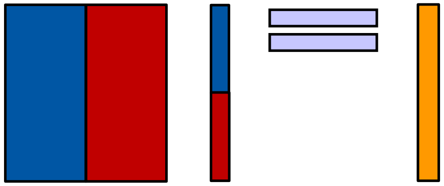
$$A_3 \begin{pmatrix} x_3 \\ x_d \end{pmatrix} = \widetilde{h_3}$$



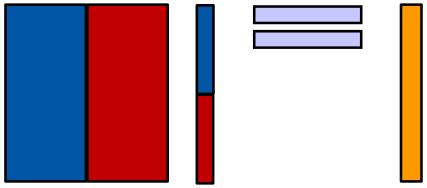
$$A_4 \begin{pmatrix} x_4 \\ x_d \end{pmatrix} = \widetilde{h_4}$$

…and we try to decouple the responses.

System Decoupling

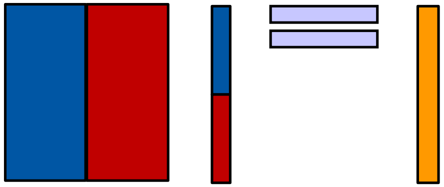


$$A_1 \begin{pmatrix} x_1 \\ x_d \end{pmatrix} = \widetilde{h_1}$$

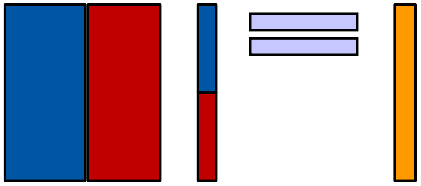


$$A_2 \begin{pmatrix} x_2 \\ x_d \end{pmatrix} = \widetilde{h_2}$$

Denominator unknowns are still common to all the responses!

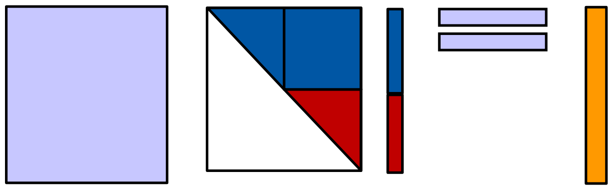


$$A_3 \begin{pmatrix} x_3 \\ x_d \end{pmatrix} = \widetilde{h_3}$$

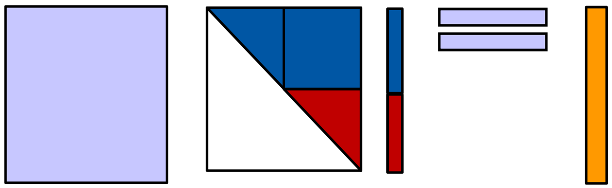


$$A_4 \begin{pmatrix} x_4 \\ x_d \end{pmatrix} = \widetilde{h_4}$$

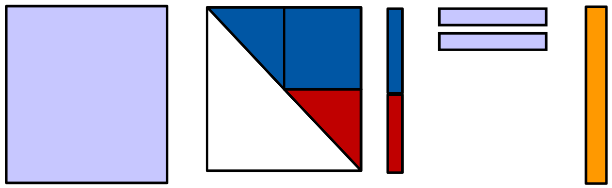
Denominator Responses Isolation



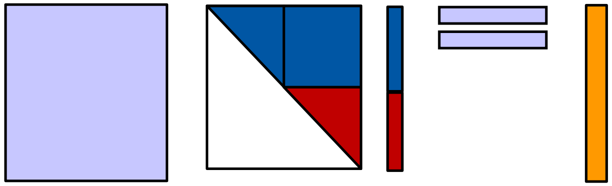
$$Q_1 R_1 \begin{pmatrix} x_1 \\ x_d \end{pmatrix} = \widetilde{h_1}$$



$$Q_2 R_2 \begin{pmatrix} x_2 \\ x_d \end{pmatrix} = \widetilde{h_2}$$



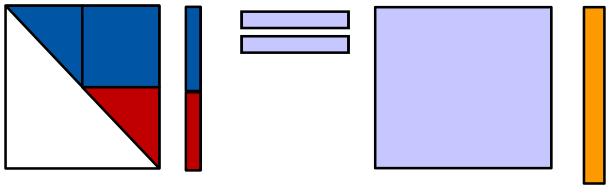
$$Q_3 R_3 \begin{pmatrix} x_3 \\ x_d \end{pmatrix} = \widetilde{h_3}$$



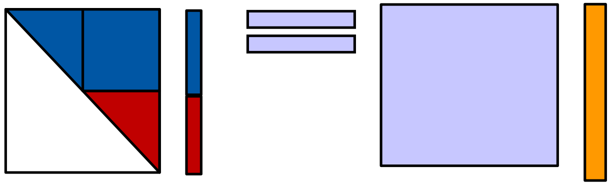
$$Q_4 R_4 \begin{pmatrix} x_4 \\ x_d \end{pmatrix} = \widetilde{h_4}$$

We perform the QR factorizations of the decoupled regressors…

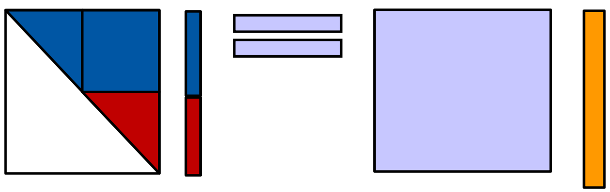
Denominator Responses Isolation



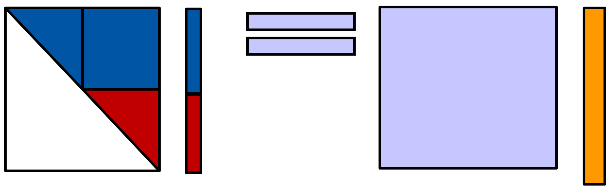
$$R_1 \begin{pmatrix} x_1 \\ x_d \end{pmatrix} = Q_1^H \widetilde{h_1}$$



$$R_2 \begin{pmatrix} x_2 \\ x_d \end{pmatrix} = Q_2^H \widetilde{h_2}$$



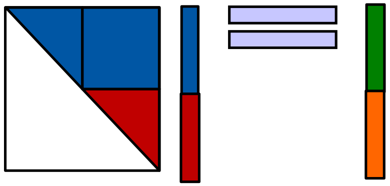
$$R_3 \begin{pmatrix} x_3 \\ x_d \end{pmatrix} = Q_3^H \widetilde{h_3}$$



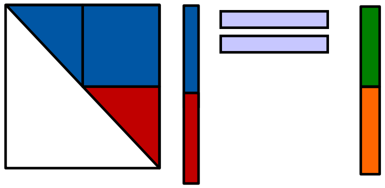
$$R_4 \begin{pmatrix} x_4 \\ x_d \end{pmatrix} = Q_4^H \widetilde{h_4}$$

…and we project the known terms on the new orthonormal basis.

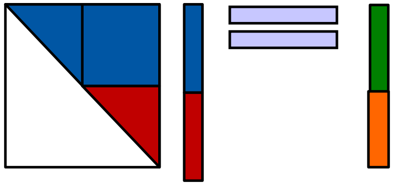
Denominator Responses Isolation



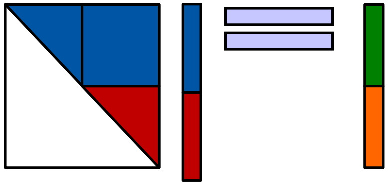
$$R_1 \begin{pmatrix} x_1 \\ x_d \end{pmatrix} = \begin{pmatrix} b_1 \\ b_{d1} \end{pmatrix}$$



$$R_2 \begin{pmatrix} x_2 \\ x_d \end{pmatrix} = \begin{pmatrix} b_2 \\ b_{d2} \end{pmatrix}$$



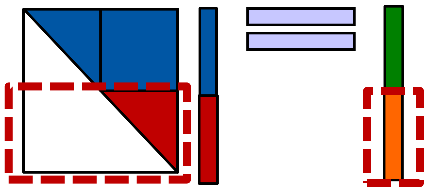
$$R_3 \begin{pmatrix} x_3 \\ x_d \end{pmatrix} = \begin{pmatrix} b_3 \\ b_{d3} \end{pmatrix}$$



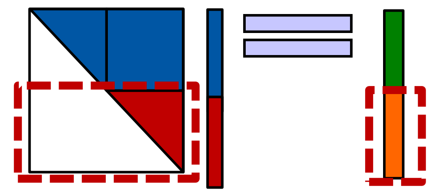
$$R_4 \begin{pmatrix} x_4 \\ x_d \end{pmatrix} = \begin{pmatrix} b_4 \\ b_{d4} \end{pmatrix}$$

We observe how the last rows of each regressor are now related only to the denominator…

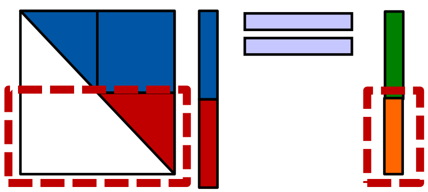
Denominator Responses Isolation



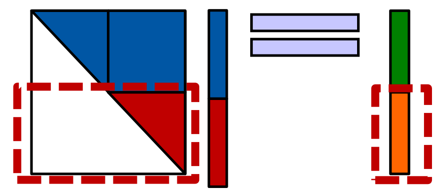
$$R_1 \begin{pmatrix} x_1 \\ x_d \end{pmatrix} = \begin{pmatrix} b_1 \\ b_{d1} \end{pmatrix}$$



$$R_2 \begin{pmatrix} x_2 \\ x_d \end{pmatrix} = \begin{pmatrix} b_2 \\ b_{d2} \end{pmatrix}$$



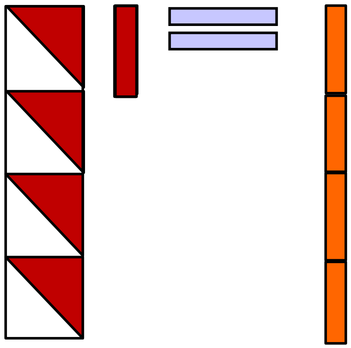
$$R_3 \begin{pmatrix} x_3 \\ x_d \end{pmatrix} = \begin{pmatrix} b_3 \\ b_{d3} \end{pmatrix}$$



$$R_4 \begin{pmatrix} x_4 \\ x_d \end{pmatrix} = \begin{pmatrix} b_4 \\ b_{d4} \end{pmatrix}$$

We observe how the last rows of each regressor are now related only to the denominator…

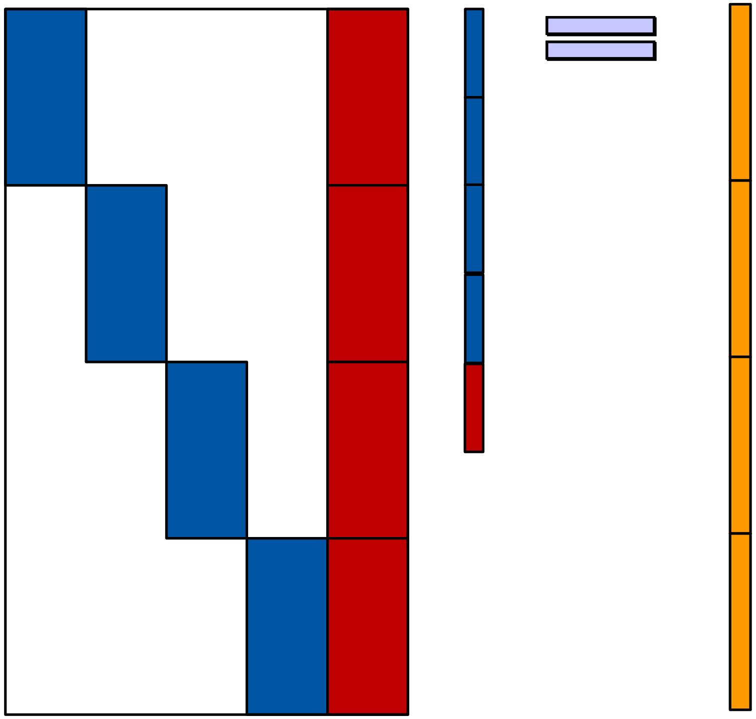
Denominator Least Squares Problem



$$R_d x_d = b_d$$

… so we build a single LS problem to solve for the denominator unknowns

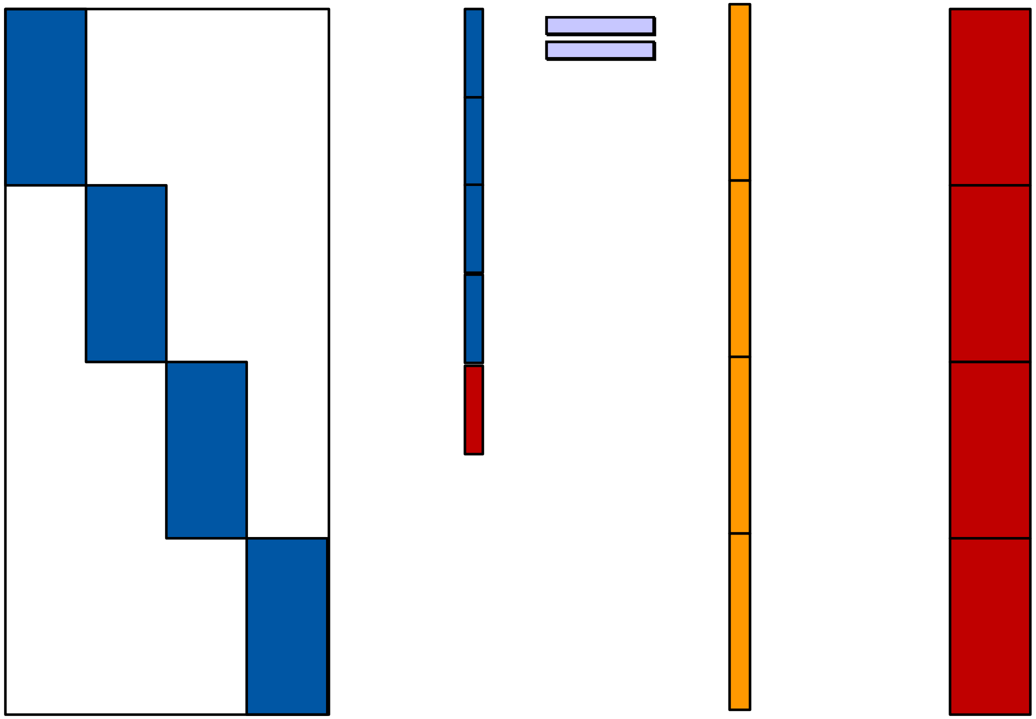
Achieve the Diagonal Block Form



$$\boldsymbol{Ax} = \boldsymbol{b}$$

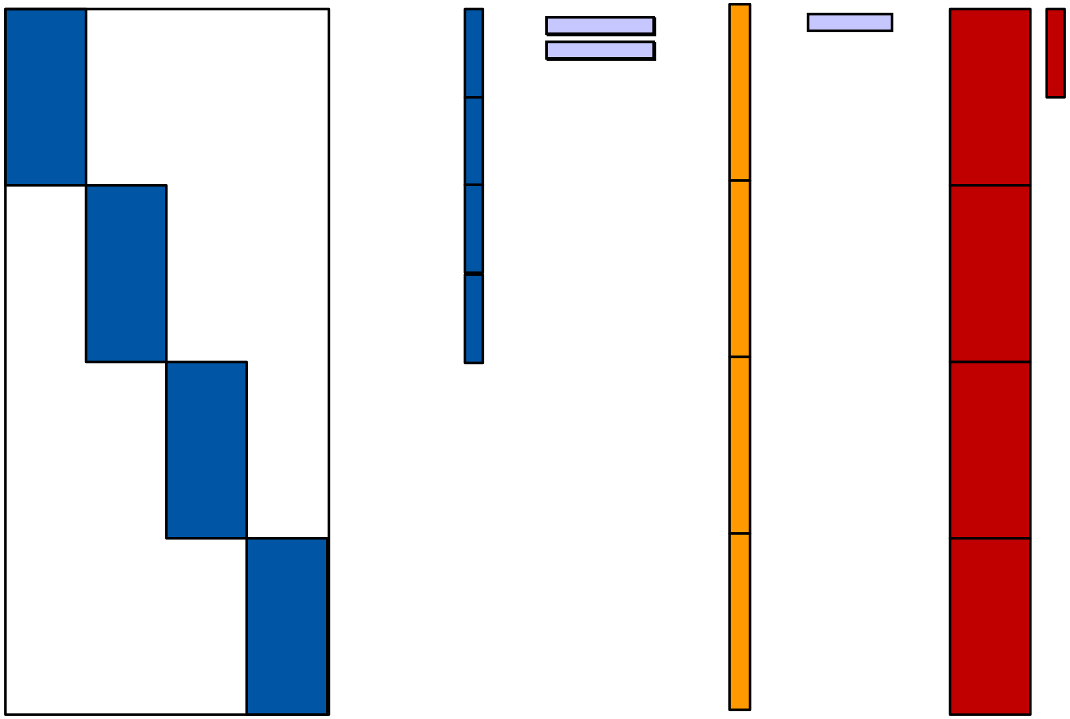
We consider again the initial system …

Achieve the Diagonal Block Form



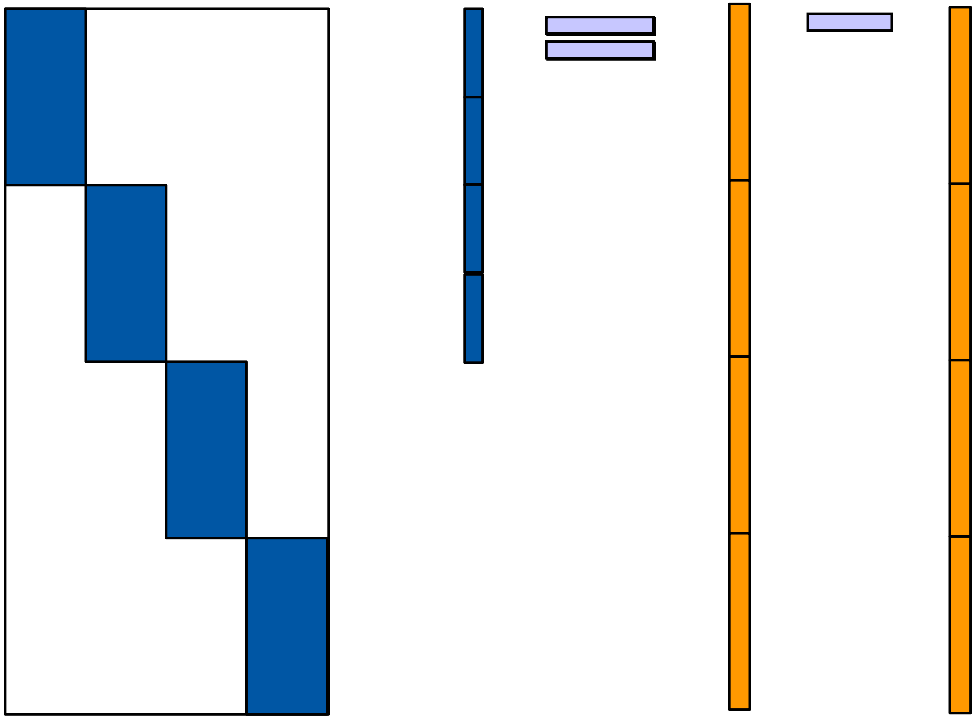
…and exploit the found coefficients to eliminate the columns that couple the responses

Achieve the Diagonal Block Form



…and exploit the found coefficients to eliminate the columns that couples the responses

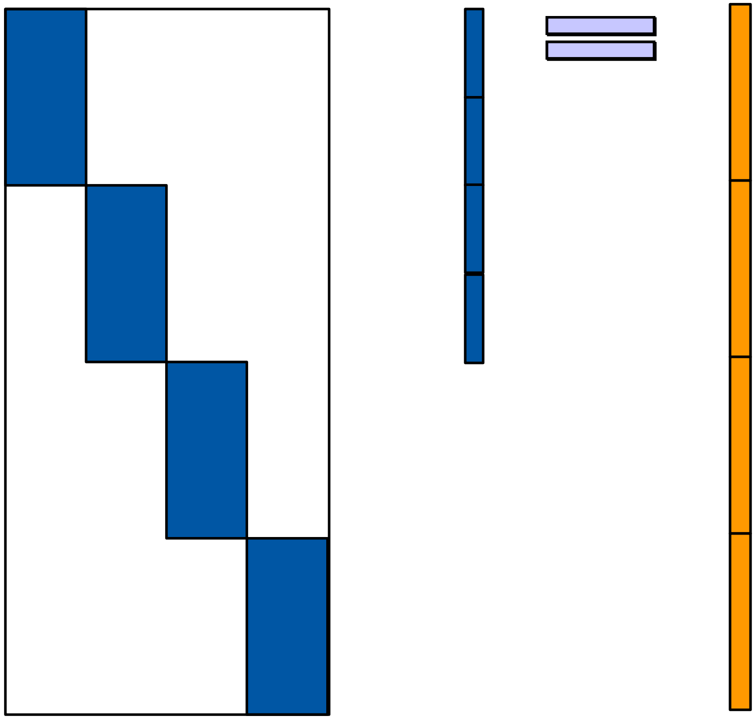
Achieve the Diagonal Block Form



…and exploit the found coefficients to eliminate the columns that couples the responses

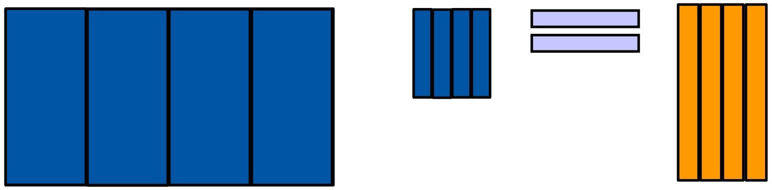
Achieve the Diagonal Block Form

$$A_n x_n = b_n$$



…and exploit the found coefficients to eliminate the columns that couples the responses

Achieve the Diagonal Block Form

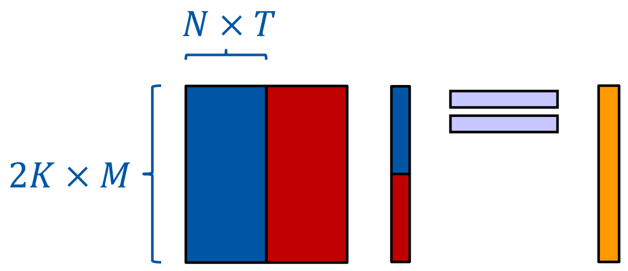


$$A_n X_n = B_n$$

We solve a multiple right-hand side LS problem and we are done!

Computational Requirements of the Fast-PSK

Dominated by the QR factorization that we perform over each decoupled response…



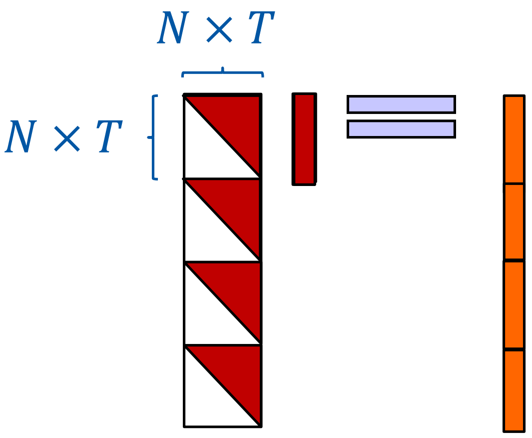
The scaling with the responses is now linear!

$$Fl_{QR} = 2mn^2 \quad \xrightarrow{\text{In our case…}} \quad Fl_{FPSK} \propto KMN^2T^2 \times L$$

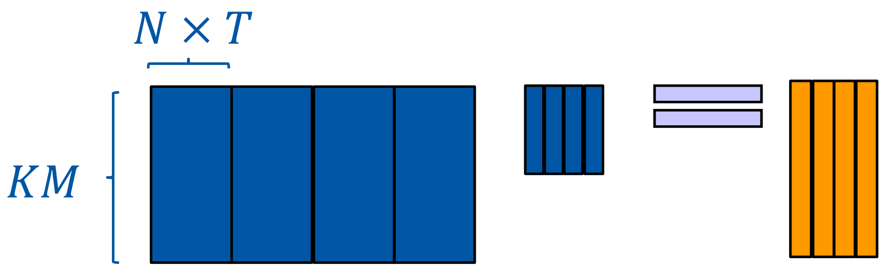
Memory Requirements of the FPSK

For the denominator:

$$E_{Den} = N^2T^2L + NTL$$

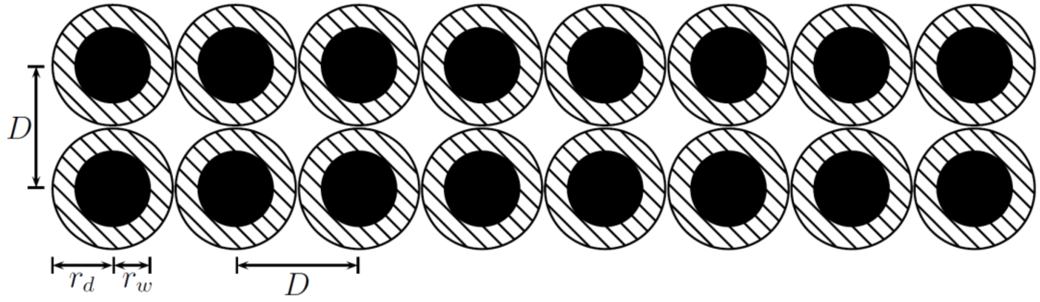


For the numerator:



$$E_{Num} = KMNTL + KML$$

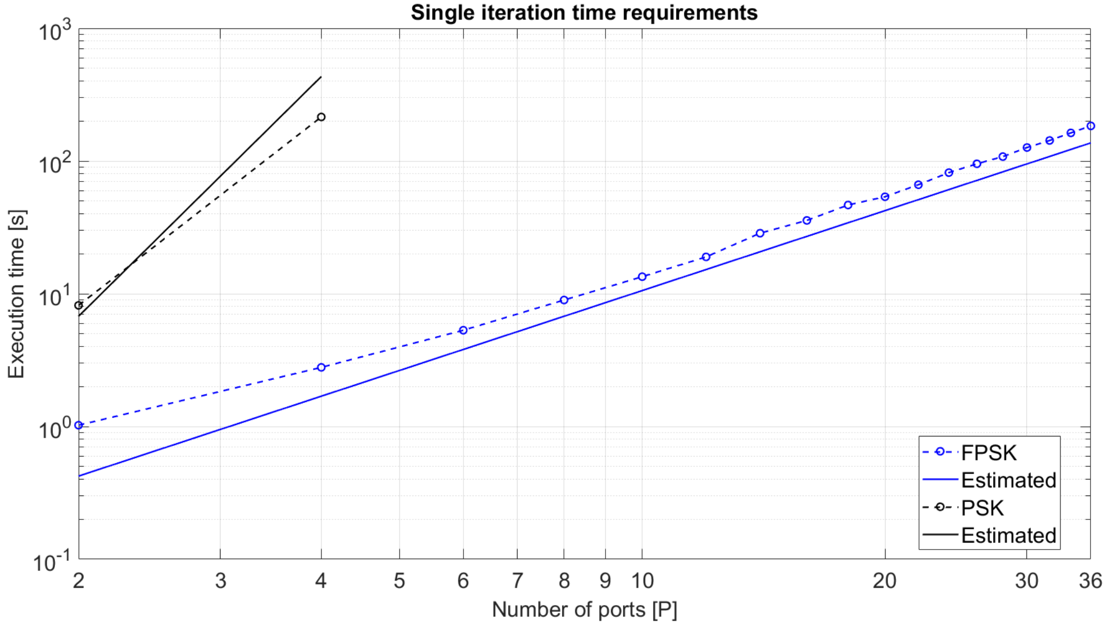
A Test Case: Coupled Transmission Lines



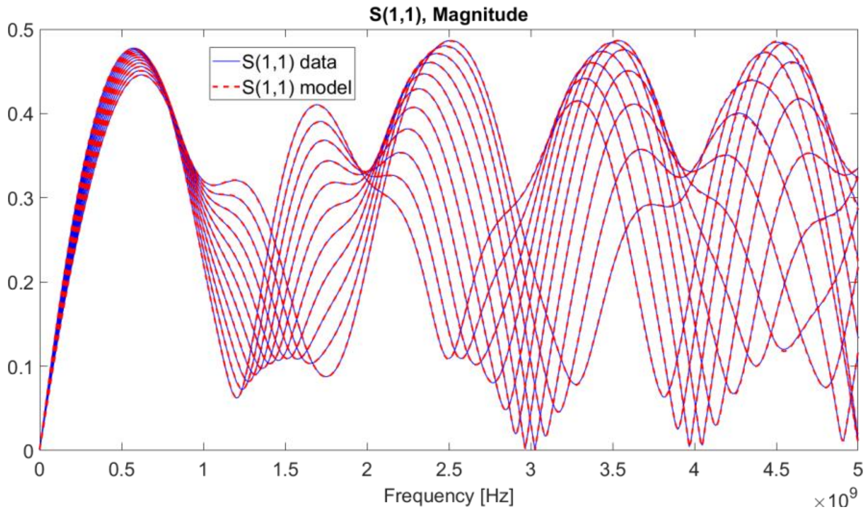
A 16-port system

Free parameter: length of the coupling

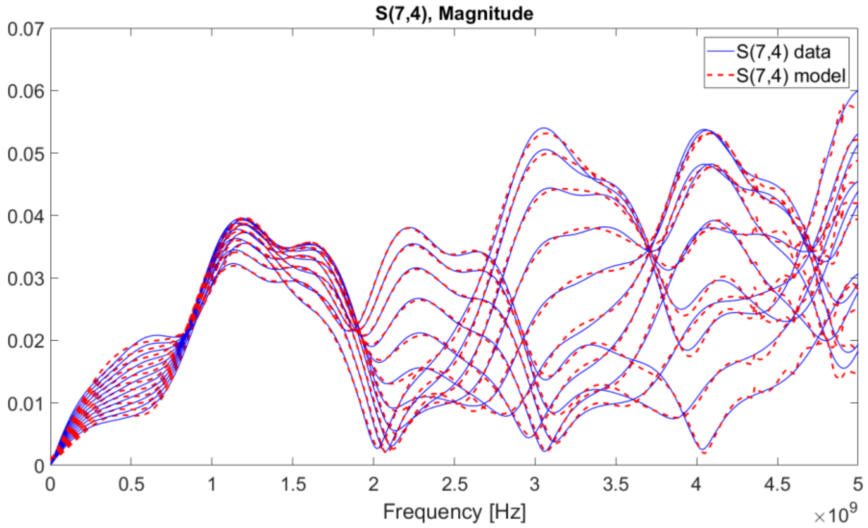
We modeled the same system with a number of coupled conductors ranging from 1 to 18 to test the port scaling of time requirements.



Accuracy of the Fitting (12-port case)



Accuracy is good for non-attenuating responses…



…fair for highly attenuating ones!

Minimization of the Relative Error

Achieved through a weighting scheme…

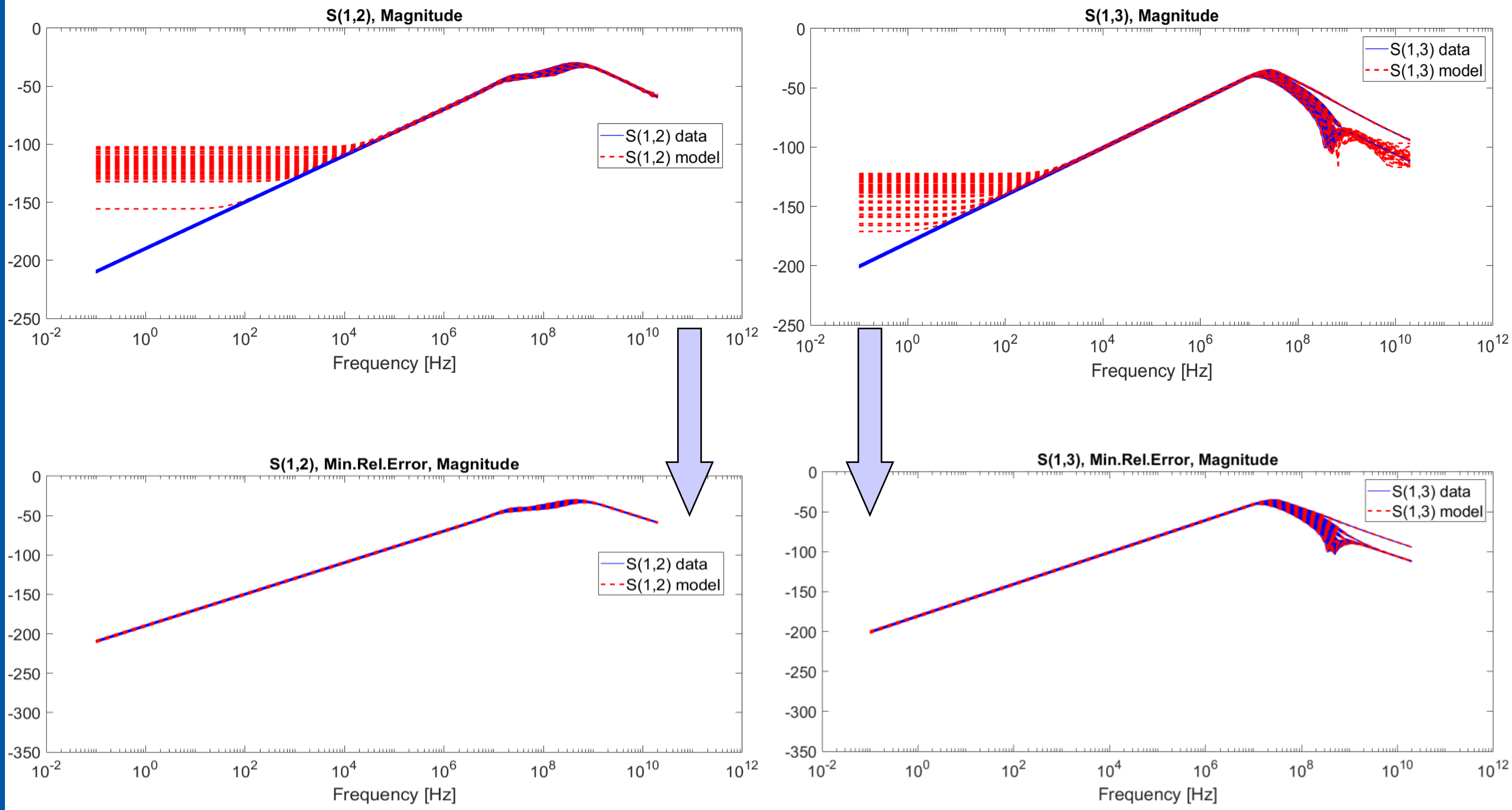
For every data sample and for every i-th response, minimize…

$$r^{i}_{k,m} = \left| \frac{N^{i}(s_k;\boldsymbol{\vartheta}_m) - D(s_k;\boldsymbol{\vartheta}_m)\breve{h}^{i}_{k,m}(s_k;\boldsymbol{\vartheta}_m)}{D(s_k;\boldsymbol{\vartheta}_m)\left|\breve{h}^{i}_{k,m}\right|^{\beta}} \right|^{2}$$

Normalization

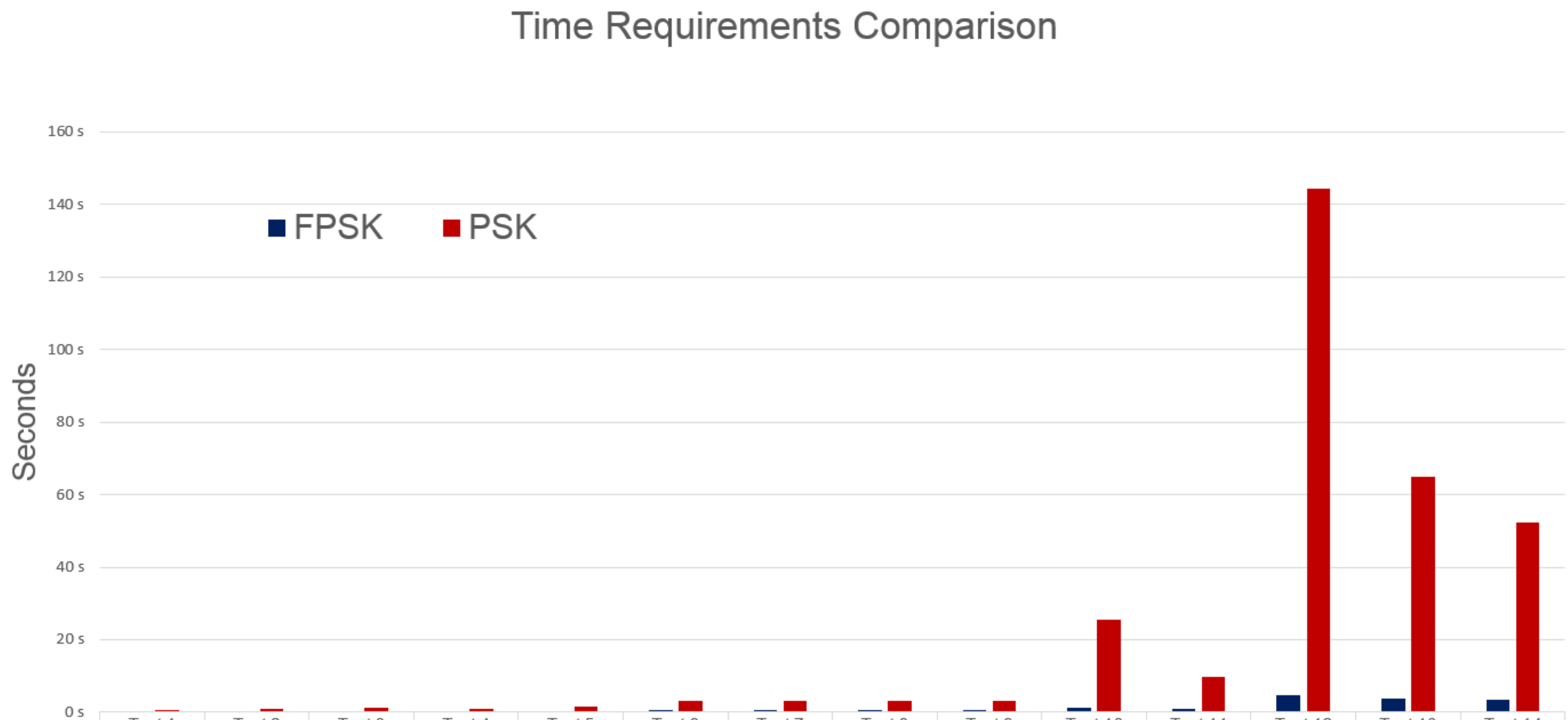
A free parameter β allows tuning the fitting procedure

Resulting Fitting on a Two Parameters Example



Other Examples Comparisons

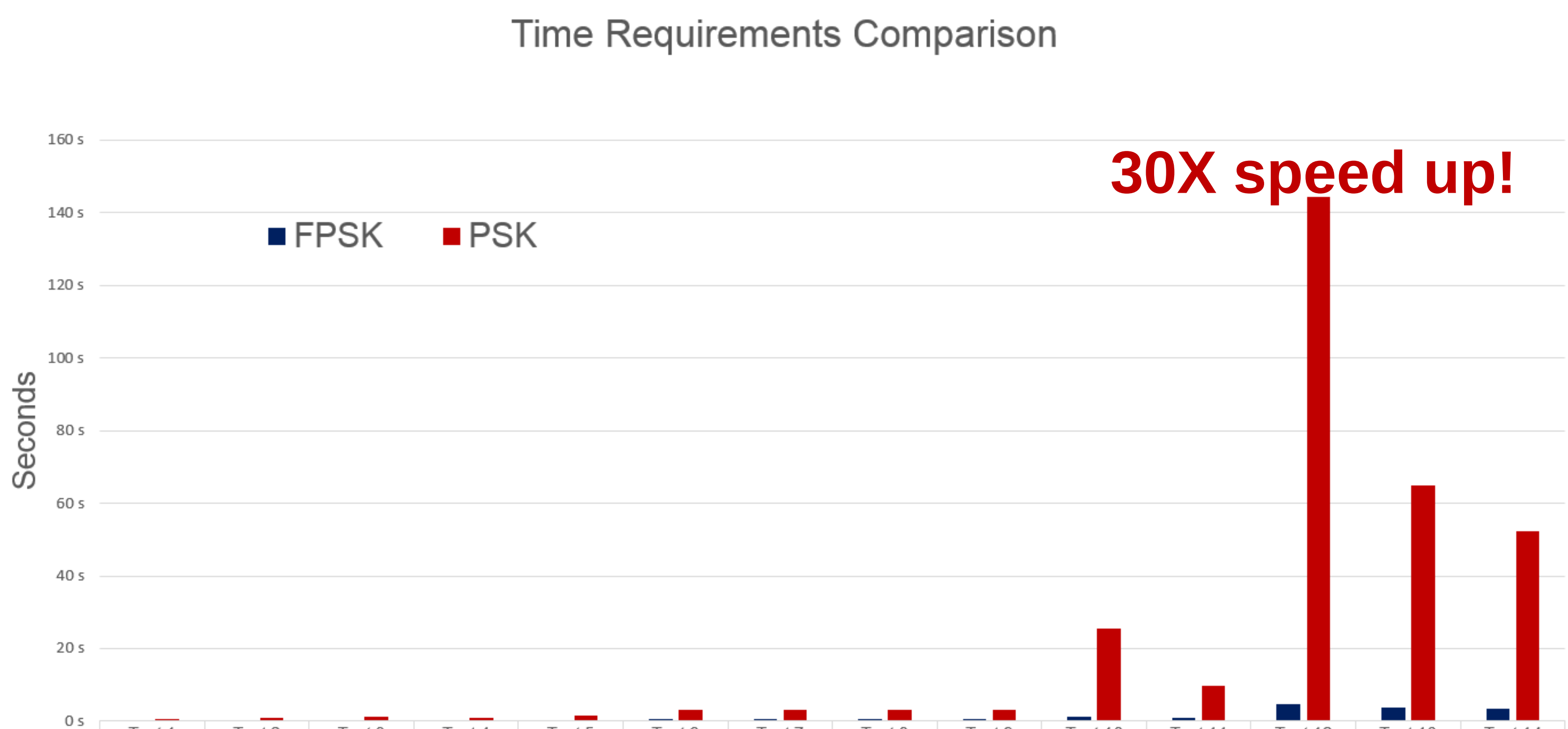
We compare the time requirements over other 14 test examples



	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8	Test 9	Test 10	Test 11	Test 12	Test 13	Test 14
FPSK	0,19 s	0,28 s	0,31 s	0,22 s	0,31 s	0,58 s	0,57 s	0,51 s	0,56 s	1,36 s	0,84 s	4,76 s	3,79 s	3,5 s
PSK	0,62 s	1,01 s	1,19 s	0,81 s	1,44 s	3,1 s	3,12 s	3,08 s	3,11 s	25,41 s	9,64 s	144,27 s	65,05 s	52,43 s

Other Examples Comparisons

We compare the time requirements over other 14 test examples



	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8	Test 9	Test 10	Test 11	Test 12	Test 13	Test 14
FPSK	0,19 s	0,28 s	0,31 s	0,22 s	0,31 s	0,58 s	0,57 s	0,51 s	0,56 s	1,36 s	0,84 s	4,76 s	3,79 s	3,5 s
PSK	0,62 s	1,01 s	1,19 s	0,81 s	1,44 s	3,1 s	3,12 s	3,08 s	3,11 s	25,41 s	9,64 s	144,27 s	65,05 s	52,43 s

Conclusions and Further Improvements

We applied a decouple-and-compress procedure to the PSK algorithm that grants:

- **Linear complexity scaling with the number of responses to be modeled**
- **Major reduction in memory requirements**

Additionally we allow:

- **Possibility to model high dynamic range systems**
- **Guaranteed uniformly stable models** → **In the next presentation**

Further Improvement: the algorithm is suitable to be easily parallelized.