Receiving Mode CST MWS/MATLAB Co-Simulations for Time and Frequency Domain

UWB Small Antenna Array Characterization

M. Gardill
markus.gardill@fau.de

CST European User Conference 2013

Institute for Electronics Engineering
Prof. Dr.-Ing. Dr.-Ing. habil. R. Weigel, Prof. Dr.-Ing. Georg Fischer
Cauerstr. 9, 91058 Erlangen, Germany
Outline

• Introduction/Motivation
• RX Mode UWB Antenna (Array) Characterization
• Simulation Framework and Simulation Results
• Summary & Outlook
Introduction and Motivation
Application Scenario: Industrial High-Precision Positioning

Existing System (Symeo GmbH)

- Reference units (RU): known position
- Measurement Unit (MU): position to be estimated
- Multilateration: FMCW; 5.8 GHz ISM

Current research: extension towards …

- UWB FMCW, 5 GHz – 8 GHz
  - High precision, multipath mitigation
- UWB DOA estimation
  - Instantaneous heading estimation
  - NLOS path detection

Focus today: UWB RX antenna array characterization and simulation using CST
Motivation: Precise Characterization of RX Antenna Arrays

Institute for Electronics Engineering

regular approach

proposed approach

Motivation:

Only use RX information: eliminate sources of error / proper excitation

Find LTI system formulation for UWB mutual coupling effects

UWB Antenna (Array) Characterization
UWB Antenna Characterization: Spatial LTI Filter

Realized Gain of isolated element:

\[ G_{\text{eff}}^{\text{iso}}(\omega, \theta, \phi) = \frac{\omega^2}{\pi c_0^2} |H^{\text{ISO}}(\omega, \theta, \phi)|^2 \]

IEEE Gain:

\[ G^{\text{ISO}}(\omega, \theta, \phi) = \frac{G_{\text{eff}}^{\text{ISO}}(\omega, \theta, \phi)}{1 - |S_{1,1}(\omega)|^2} \]

\[ v^{\text{Rx}}(t, \theta, \phi) = \sqrt{\frac{Z_C}{Z_F}} h^{\text{ISO}}(t, \theta, \phi) * E^{\text{inc}}(t - t_p, \theta, \phi) \]

\[ V^{\text{Rx}}(\omega, \theta, \phi) = \sqrt{\frac{Z_C}{Z_F}} H^{\text{ISO}}(\omega, \theta, \phi) E^{\text{inc}}(\omega, \theta, \phi) e^{-j \omega t_p} \]
UWB Arrays I: Only Consider Propagation Delay

Assumptions:
- Identical elements
- Identical orientation
- No mutual coupling

\[
V_{n}^{RX}(\omega, \theta, \phi) = \sqrt{\frac{Z_C}{Z_F}} H^{ISO}(\omega, \theta, \phi) E^{inc}(\omega, \theta, \phi) e^{-j \omega t_n}
\]

\[
\mathbf{V}_{n}^{RX}(\omega, \theta, \phi) = \sqrt{\frac{Z_C}{Z_F}} H^{ISO}(\omega, \theta, \phi) \mathbf{E}_{n}^{inc}(\omega, \theta, \phi)
\]
UWB Arrays II: Representation of Mutual Coupling

Assumptions:
- Identical elements
- Identical orientation
- No mutual coupling

\[ \tilde{V}_{n}^{RX}(\omega, \theta, \phi) = \sqrt{\frac{Z_C}{Z_F}} H_{n}^{MC}(\omega, \theta, \phi) E^{inc}(\omega, \theta, \phi) e^{-j \omega t_n} \]

- Exact representation of mutual coupling
- Generally: new effective height \(H_{n}^{MC}(\omega, \theta, \phi)\) unknown
- But: relation with \(H^{ISO}(\omega, \theta, \phi)\) expected \(\Rightarrow\) find model
UWB Arrays III: UWB Mutual Coupling Model

- Proposed by Wang for MC compensation [1]
- $G^{-1}$ for MC compensation
- Represent $H_{n}^{MC}(\omega, \theta, \phi)$ as linear combination of $H^{ISO}(\omega, \theta, \phi)$
- Incomplete representation of coupling [2]
- Serves as basis for our coupling model

---


UWB Arrays IV: Normalized Coupling Height Formalism

Modelled output voltage spectrum of element $n$ ...

$$\hat{V}_{n}^{RX}(\omega, \theta, \phi) = V_{n}^{RX}(\omega, \theta, \phi) + \sum_{m=1, m\neq n}^{N} G_{n,m} V_{m}^{RX}(\omega, \theta, \phi)$$

... expressed using the coupling effective height $C_{n}(\omega, \theta, \phi)$ [1]

$$\hat{V}_{n}^{RX}(\omega, \theta, \phi) = \sqrt{\frac{Z_{C}}{Z_{F}}} \left( (H^{ISO}(\omega, \theta, \phi) + C_{n}(\omega, \theta, \phi) \right) E^{inc}(\omega, \theta, \phi) e^{-j \omega t_{n}}$$

Simulation Framework
Simulation Framework Overview

- $C_{n}^{CO}(\omega, \theta_i, \phi_i)$ can be obtained from RX simulation (or measurement)

$$C_{n}^{CO}(\omega, \theta_i, \phi_i) = \sqrt{\frac{Z_C}{Z_F}} \frac{\tilde{V}_n^{RX}(\omega, \theta_i, \phi_i)}{E^{inc}(\omega, \theta_i, \phi_i)} e^{j\omega t_n} - H^{ISO,co}(\omega, \theta_i, \phi_i)$$

- CST / Matlab co-simulation framework for simulation control and data postprocessing; communication via COM objects and ResultReader DLL
- Transient solver -> time domain signals
Simulation Framework: CST Model and Planewave Setup

- model antenna array
- assign port (discrete or WG) to each element
- optional: replace antenna elements with E-Field Probes (phase calibration)

create plane wave
- currently: use linear polarization
- polarization and plane normal vectors set by MATLAB tool
Simulation Framework: Solver Setup

- set desired frequency range
- use appropriate BCs
  - all open / open add space
  - or: PEC/PMC plane (e.g. planar antenna arrays)
- apply TD solver parameters
  - accuracy -80 dB
  - source set by MATLAB
  - acceleration
  - no adaptive meshing

CST setup complete -> save project + close project
Simulation Framework: MATLAB Usage

Start MATLAB → call function CST_planewave

```matlab
function [CST_Planewave_RawData] = ...
CST_planewave(s_filename, range_theta,
range_phi, samples_theta, samples_phi,
offset_theta, offset_phi)
```

- `s_filename`: string containing path and filename to CST project
- `range_theta`: range in which the elevation is varied
- `range_phi`: range in which the azimuth is varied
- `samples_theta`: number of evaluations for variation of theta
- `samples_phi`: number of evaluations for variation of phi
- `offset_theta`: start position for theta variation
- `offset_phi`: start position for phi variation

CST is started → simulation runs automatically → TD port data is recorded → backups on HDD → all simulations finished → CST is closed
Simulation Framework: MATLAB execution

CST_planewave implements various automated or interactive features:

- load ResultReader library
- create CST COM Application Object
- find all ports in project, allocate variables
- find planewave and get handle to planewave
- set TD source to Plane Wave
- get simulation parameters
- transform desired array coordinate system w.r.t. CST global coordinate system
- control Plane Wave pol./ plane normal
- run simulation
- read results
- …
Simulation Framework: Result Structure

CST_Planewave_RawData

- .ports
  - .x
  - .y
- .planewave
  - .x
  - .y
- .v_eval_phi
- .v_eval_theta
- .params
- .s_filename

TD data for all ports defined in project
x-data, $N_\theta \times N_\phi \times N_{\text{ports}}$ cell array, $N_{\text{RX}}$-length vector
y-data, $N_\theta \times N_\phi \times N_{\text{ports}}$ cell array, $N_{\text{RX}}$-length vector
TD data for impinging plane wave
x-data, $N_{\text{PW}}$-length vector
y-data, $N_{\text{PW}}$-length vector
phi evaluation angles in [rad]
theta evaluation angles in [rad]
units, $f_{\text{max}}, f_{\text{min}}$
string containing path and filename to CST project

Why use cell array?
Simulation time (result length) dependent on $\theta_i, \phi_i$
Simulation Framework: MATLAB Postprocessing

Raw TD simulation data has to be postprocessed
- resampling
- interpolation
- alignment across $\theta_i, \phi_i$ (unequal result length)
- FFT
- normalization
- $H^{ISO}, H^{MC}, C_n$ computation
- propagation delay (phase) compensation
- ...

CST_Planewave_RawData

Calc_RecMode
Params

$\begin{bmatrix}
H_1^{MC} \\
\vdots \\
H_N^{MC}
\end{bmatrix}$ or $H^{ISO}$ (+ prop. phase)

calc_Coupling
EffectiveHeight

$H^{ISO}$

Compensate_CouplingEffectiveHeight

$C_n^{CO}$ (+ prop. phase)
Simulation Framework: MATLAB Postprocessing

Raw TD simulation data has to be postprocessed:

- resampling
- interpolation
- alignment across $\theta_i, \phi_i$ (unequal result length)
- FFT
- normalization
- $H_{ISO}$, $H_{MC}$, $C_n$ computation
- propagation delay (phase) compensation
- ...

If reference data from E-Field Probe simulation is used, correct phase information can already be obtained during receive mode parameter computation.
Simulation Results & Examples
Simulation: Biconical Antenna Array

- First step: cross-verification → compare realized gain
- Excellent agreement our algorithm ↔ CST internal NF-FF transform
- Plane Wave excitation: phase reference unknown → phase correction
  - either: estimate linear phase part + compensate
  - or (better): calibrate using phase data from E-Field probe setup
Simulation: Biconical Antenna Array, $\theta = 90^\circ, \phi = 0^\circ$

- Broadside plane wave -> symmetry
- Note gain and phase distortion due to mutual coupling
- $G_{\text{eff}}^C(\omega, \theta, \phi)$ or $|C_n(\omega, \theta, \phi)|$ quantify deviation from isolated behaviour
Simulation: Vivaldi Stick Antenna Array

- First step: cross-verification -> compare realized gain -> agreement
- Example here: beneficial effect of mutual coupling
  - Realized Gain at low frequencies increases in array environment
Simulation: Vivaldi Stick Antenna Array, $\theta = 90^\circ, \phi = 0^\circ$

- Broadside plane wave $\rightarrow$ symmetry
- Obvious increase of realized Gain at $f \geq 5$ GHz $\Rightarrow$ Why?
  - $G_{\text{eff}}^C(\omega, \theta, \phi)$ has significant influence
  - $\angle C_n(\omega, \theta, \phi) - \angle H^{\text{ISO}}(\omega, \theta, \phi)$ is small
Summary
Summary

- We proposed a formalism for RX UWB antenna array characterization
- We developed and introduced a CST MWS/MATLAB co-sim. framework
- Results of two array structures were shown
- Basically, RX simulation is not necessary here, but is an advantageous option and ensures proper excitation according to theory

Outlook

- RX simulation is very time consuming (one simulation per angle)
- Framework is extended to obtain results from TX simulation
- Motivation: “close the loop” across TX and RX array theory and experiments

Note

- MATLAB functions provided
Thank you for your attention.

“Receiving Mode CST MWS – Matlab Co-Simulations for Time and Frequency Domain UWB Small Antenna Array Characterization”

Markus Gardill
markus.gardill@fau.de

CST European User Conference 2013