Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Federal Department of Justice and Police FDJP

Federal Office of Metrology METAS

Metas.UncLib

Ein Messunsicherheitsrechner für Fortgeschrittene

Metas Seminar 09.09.2009

Michael Wollensack und Marko Zeier

Programming today is a race between software engineers striving to build bigger and better idiot-proof programs, and the Universe trying to produce bigger and better idiots. So far, the Universe is winning.

Rich Cook

Outline

- 1. Introduction
  - 1.1 VNA Tools II Project
  - 1.2 MU according to ISO GUM
  - 1.3 Metas.UncLib
  - 1.4 Linear Uncertainty Propagation
  - 1.5 Large System Modelling
- 2. Features
- 3. MATLAB Demo

1.1 VNA Tools II Project (217.08.FP.060)

- Goal: Development of a GUM based uncertainty calculator for Vector Network Analysis

- Duration: 2 years (Feb. 2008 – Jan 2010)

- METAS RF & MW laboratory:

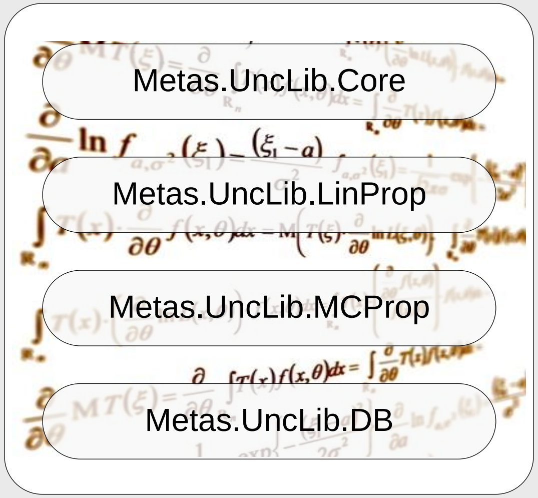
M. Wollensack J. Rüfenacht M. Zeier

- Partner:
  - Agilent Technologies, Santa Rosa, USA
    (Largest manufacturer of VNAs and calibration standards)

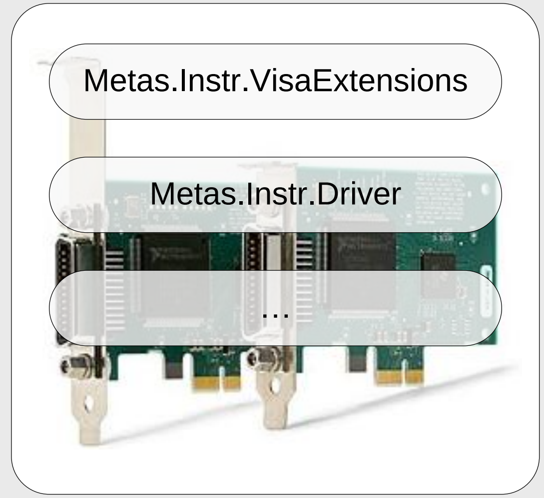
METAS

1.1 VNA Tools II Overview

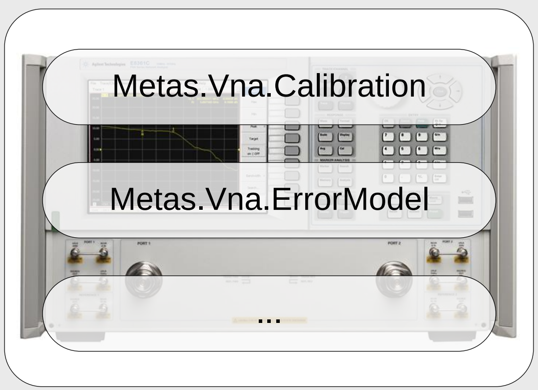
Metas.UncLib



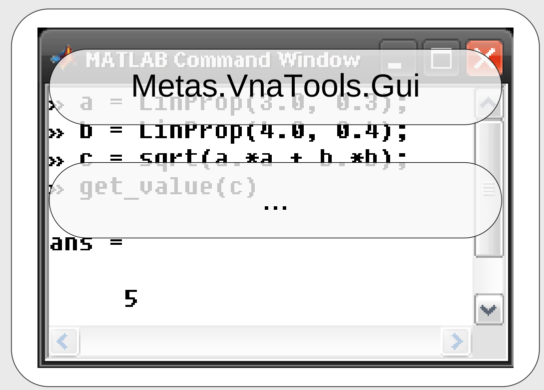
Metas.Instr



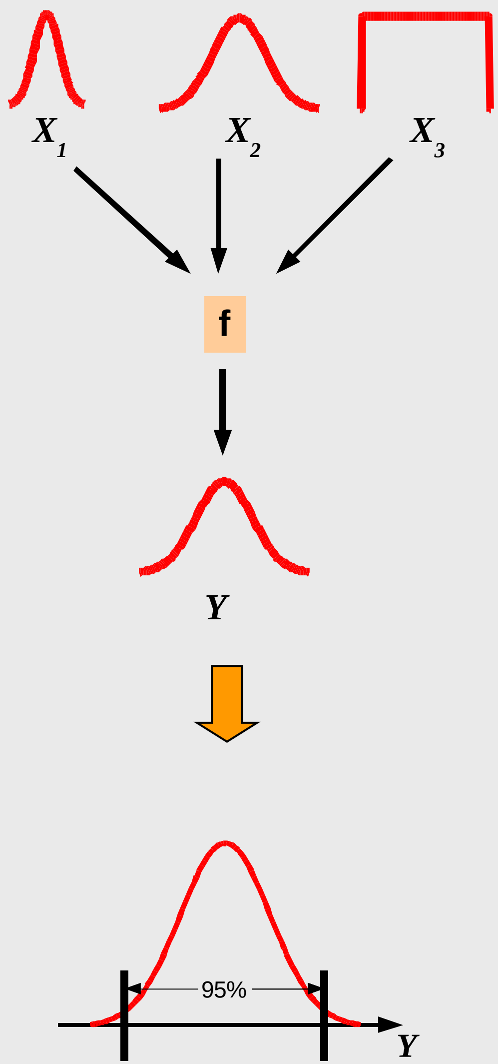
Metas.Vna



Metas.VnaTools



1.2 MU according to ISO-GUM (scalar)



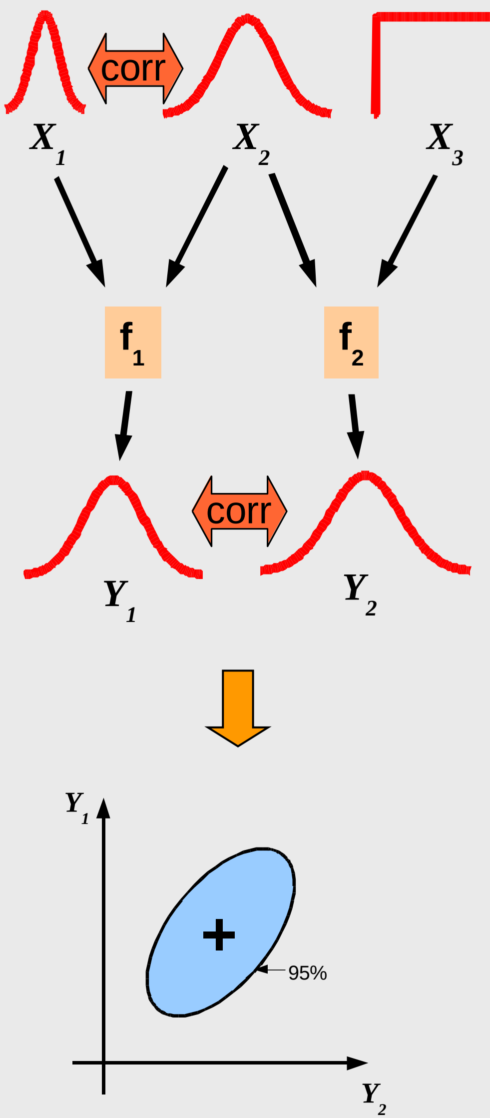
Input quantities $u(X_1), u(X_2), \ldots$

Measurement model

Output quantity $u^2(Y) = \left(\frac{\partial f}{\partial X_1}\right)^2 u^2(X_1) + \left(\frac{\partial f}{\partial X_2}\right)^2 u^2(X_2) + \ldots$

Expanded measurement uncertainty $U^2(Y) = (1.96)^2 \cdot u^2(Y)$

1.2 MU according to ISO-GUM (multivariate)



Correlation between input quantities

$$\mathbf{V_X}$$

Uncertainty Matrix

Several output quantities correlated!

$$\mathbf{V_Y} = \mathbf{J_{f,X}} \mathbf{V_X} \mathbf{J_{f,X}}^{\mathbf{T}}$$

Lin. Uncertainy Propagation

Expanded measurement uncertainty

$$\mathbf{U_Y} = \chi^2_{p,0.95} \cdot \mathbf{V_Y}$$

Calculations are tedious!

1.3 Metas.UncLib

General Purpose Uncertainty Library

It does
- support multidimensional uncertainty calculation
- advanced math (Complex, Vector, Matrix)
- automated linear uncertainty propagation
- Monte Carlo uncertainty propagation (preliminary)
- take care of correlations
- advanced storage / archiving (keeps full information)
- interfacing with other applications

It does NOT
- help to build a measurement model
- have a nice graphical interface
- produce „fancy“ output

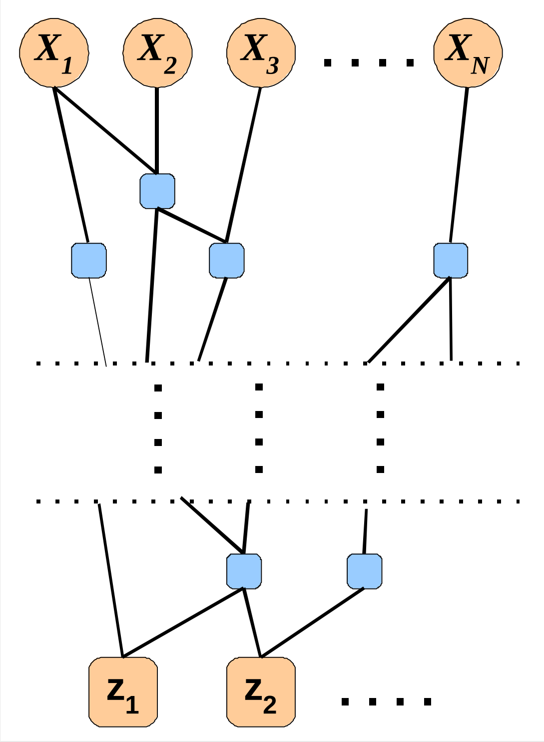
1.4 Linear Uncertainty Propagation (LUP)

Based on *GUM Tree* concept (Blair Hall, IRL/MSL, NZ)

- Automatic Differentiation
  - Calculate Derivatives (sensitivity coefficients) with machine precision
- Object oriented implementation
  - Hide complexity from user

1.4 LUP: Automatic Differentiation

To calculate sensitivity coefficients



Measurement model $\mathbf{z} = \mathbf{f}(\mathbf{x})$

Decomposition in elementary functions

$$\mathbf{z} = \mathbf{f}_1(\mathbf{f}_2 \ldots \mathbf{f}_\mathbf{M}(\mathbf{x}))$$

Derivatives of elementary functions are explicitly programmed

Chain rule $\mathbf{J}_{\mathbf{f}_i,\mathbf{x}} = \mathbf{J}_{\mathbf{f}_i,\mathbf{f}_j} \cdot \mathbf{J}_{\mathbf{f}_j,\mathbf{x}}$

Naturally supported by computers because compiler always decomposes equations into elementary functions

1.4 LUP: Metas.UncLib Object

New Data Type (contains value, dependencies and sensitivities)

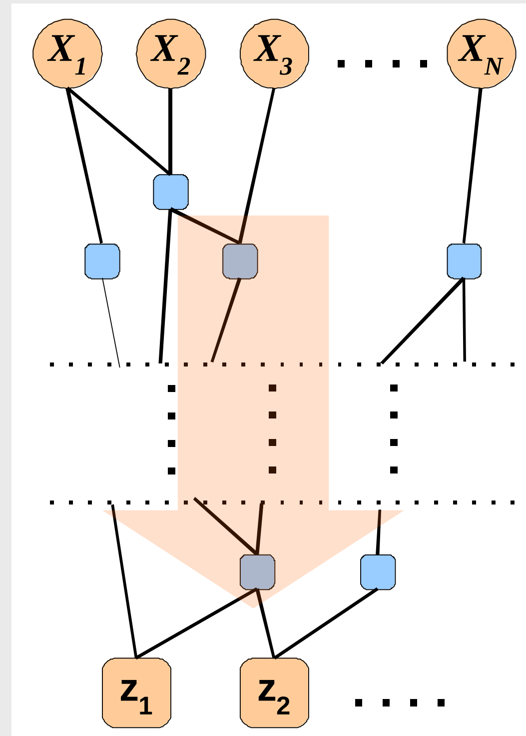
Metas.UncLib Object

value z_i

depends on $(x_1, x_3, x_4, \ldots)$

Jacobian (sensitivities) $\left(\frac{\partial z_i}{\partial x_1}, \frac{\partial z_i}{\partial x_3}, \frac{\partial z_i}{\partial x_4}, \ldots\right)$

updated at each computational step



Overloaded Operators hide complexity
→Objects can be used like ordinary numbers

Uncertainty information on demand

1.4 LUP: Correlations

Caused by common influences

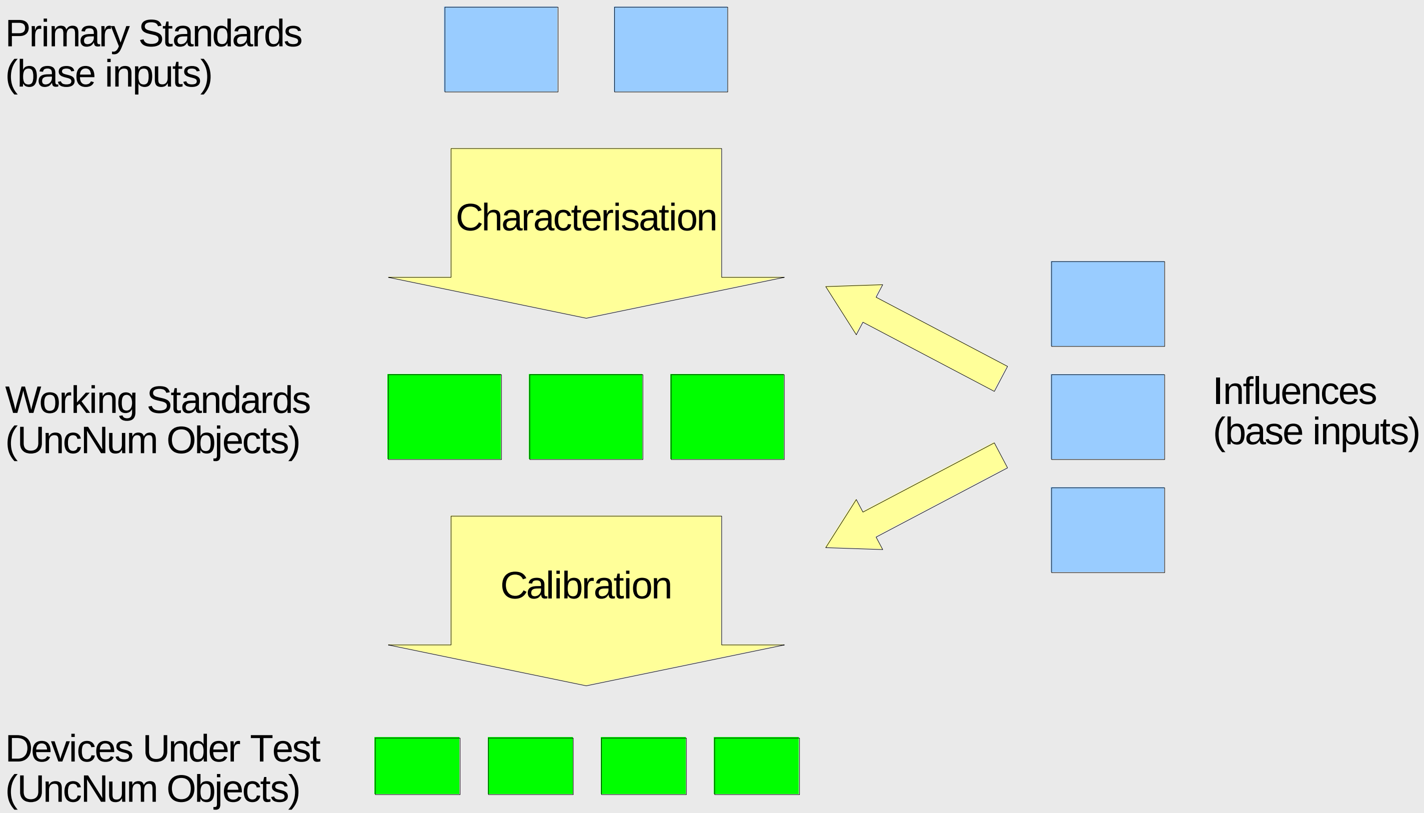
The system takes automatically care of correlations by recognizing common influences

value	z_1	Metas.UncLib Object
depends on	(x_2, x_3)	
Jacobian (sensitivities)	$\left(\frac{\partial z_1}{\partial x_2}, \frac{\partial z_1}{\partial x_3}\right)$	

value	z_2	Metas.UncLib Object
depends on	(x_1, x_2)	
Jacobian (sensitivities)	$\left(\frac{\partial z_2}{\partial x_1}, \frac{\partial z_2}{\partial x_2}\right)$	

1.5 Large System Modelling

The system allows complete modelling of a multi-step process with stored intermediate results (e.g. traceability chain)

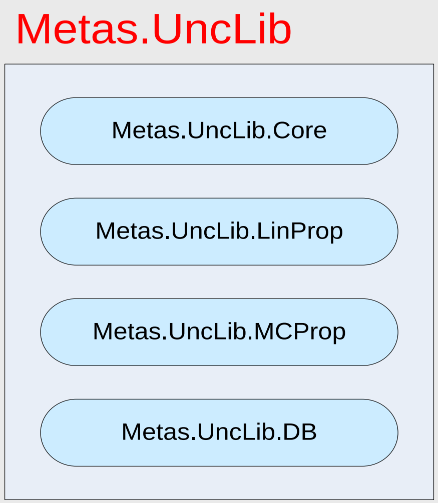


Outline

- 1. Introduction
- 2. Features
  - 2.1 Math functions
  - 2.2 Linear Algebra
  - 2.3 Uncertainty functions
  - 2.4 Storage / Archiving
  - 2.5 Interfacing .NET / COM
- 3. MATLAB Demo

2. Features

Metas.UncLib is a general purpose uncertainty library.



- Linear Uncertainty propagation
- Monte Carlo propagation
- Complex Math
- Vector and Matrix Math
- Linear Algebra
- Storage / Archiving
- .NET / COM Interface

2.1 Math functions

A list with the available functions and constants:

- Sqrt(x)
- Exp(x)
- Log(x)
- Log10(x)
- Log(x, y)
- Pow(x, y)
- Sign(x)

- Sin(x)
- Cos(x)
- Tan(x)
- Asin(x)
- Acos(x)
- Atan(x)
- Atan2(x, y)

- Sinh(x)
- Cosh(x)
- Tanh(x)
- Asinh(x)
- Acosh(x)
- Atanh(x)

- Real(x)
- Imag(x)
- Abs(x)
- Angle(x)
- Conj(x)

- PI
- E
- J
- One
- Zero

METAS

2.2 Linear Algebra

A list with the available functions:

▪ Dot(M1, M2)	Matrix multiplication of matrix M1 and M2
▪ Lu(M)	LU decomposition of matrix M
▪ Det(M)	Determinate of matrix M
▪ Inv(M)	Matrix inverse of M
▪ Solve(A, Y)	Solve linear equation system: A * X = Y
▪ Cholesky(M)	Cholesky decomposition of matrix M
▪ LstSqrSolve(A, Y)	Least square solve of overdetermined linear equation system.
▪ WeightedLstSqrSolve(A, Y, W)	Weighted least square solve of over-determined linear equation system.
▪ Fft(M)	Fast Fourier transformation
▪ Ifft(M)	Inverse Fast Fourier transformation

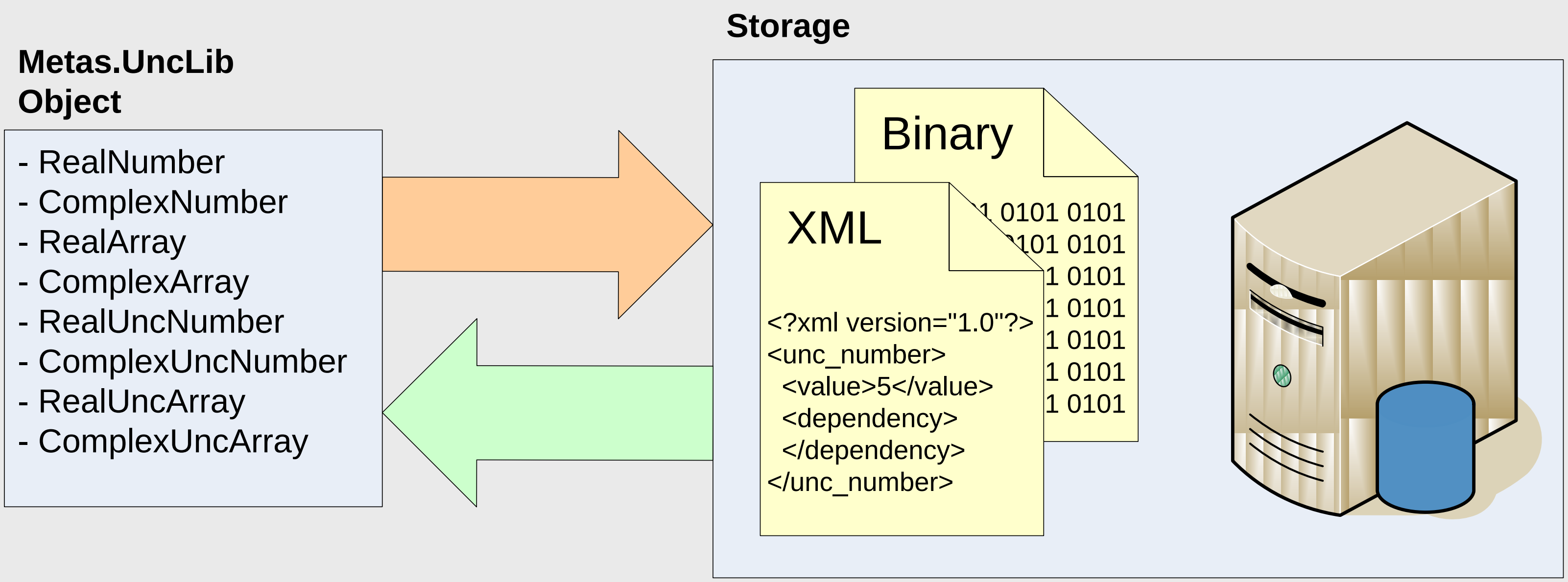
2.3 Uncertainty functions?

The available functions to obtain information from a ‘Metas.UncLib’ object:

	C# **‘Metas.UncLib’**	**MATLAB** **‘LinProp’ or ‘MCProp’**
Returns the **value**.	y.GetValue()	get_value(y)
Returns the **standard uncertainty**.	y.GetStdUnc()	get_stdunc(y)
Returns the inverse **degrees of freedom**.	y.GetIDof()	get_idof(y)
Returns the sensitivities to the virtual base inputs (with value 0 and uncertainty 1). This is equal to the uncertainties components	y.GetJacobi()	get_jacobi(y)
Returns the sensitivities to the intermediate results.	y.GetJacobi2(x)	get_jacobi2(y,x)
Returns the **uncertainty component**.	y.GetUncComponent(x)	get_unc_component(y, x)
Returns the **correlation matrix**.	[y1 y2 ...]. GetCorrelation()	get_correlation([y1 y2 ...])
Returns the **covariance matrix**.	[y1 y2 ...]. GetCovariance()	get_covariance([y1 y2 ...])

2.4 Storage / Archiving

- Save computed ‘Metas.UncLib’ objects.
- Reload stored ‘Metas.UncLib’ objects.
  After loading all information is restored.



2.5 Interfacing .NET / COM

Metas.UncLibCE

- Metas.UncLibCE.Core
- Metas.UncLibCE.LinProp
- Metas.UncLibCE.MCProp

- Pocket PC
- Smart Phone
- Windows CE Device

Metas.UncLib

- Metas.UncLib.Core
- Metas.UncLib.LinProp
- Metas.UncLib.MCProp
- Metas.UncLib.DB

- Visual Basic
- Visual C#
- Visual C++
- LabVIEW
- Mathematica
- Office 2007
- …

Metas.UncLibCOM

- Metas.UncLibCOM.Core
- Metas.UncLibCOM.LinProp
- Metas.UncLibCOM.MCProp
- Metas.UncLibCOM.DB

- C++
- Matlab
- Excel 97 - 2003
- ...

Metas.UncLib.Matlab

- LinProp
- MCProp
- LinPropDB

- Use Metas.UncLib in Matlab scripts and functions.

Outline

- 1. Introduction
- 2. Features
- 3. MATLAB Demo
  - 3.1 Right triangle example
  - 3.2 GUM H2 example
  - 3.3 Resistor cube example
  - 3.4 Circle fit example
  - 3.5 Air line characterization example

3. MATLAB Demo

MATLAB Command Window

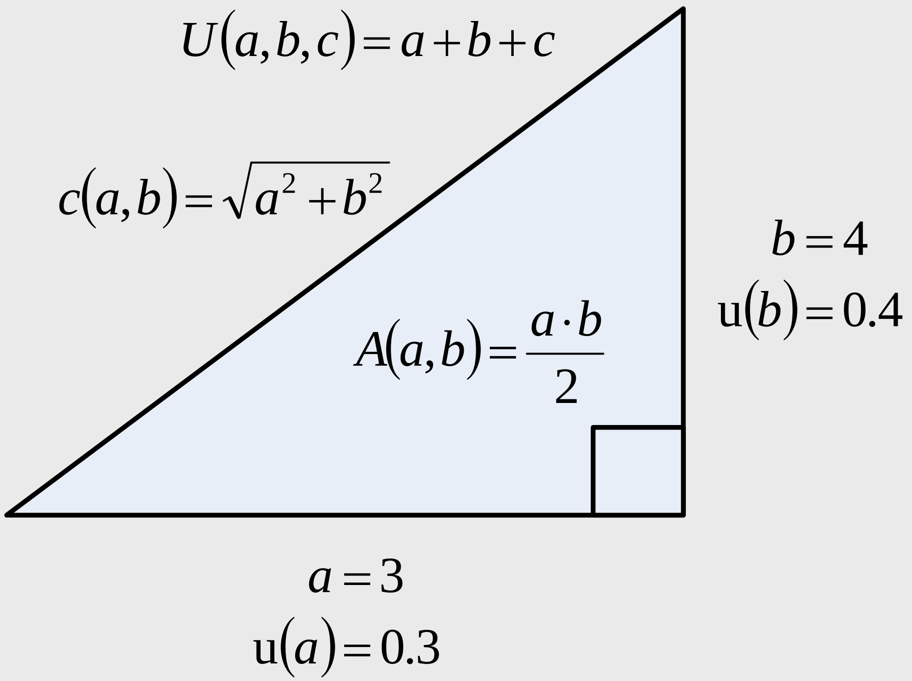
```
  To get started, type one of these: helpwin, helpdesk, or demo.
  For product information, visit www.mathworks.com.

>> a = LinProp(3.0, 0.3);
>> b = LinProp(4.0, 0.4);
>> c = sqrt(a.*a + b.*b);
>> get_value(c)

ans =

     5

>> get_stdunc(c)

ans =

      0.36715

>>
```

3.1 Right triangle

- Cathetus a = 3, u(a) = 0.3
- Cathetus b = 4, u(b) = 0.4

- What's the value and uncertainty of the hypotenuse c?

- What's the value and uncertainty of the perimeter U?



- What's the value and uncertainty of the area A?

- What's the correlation between A and U?

3.2 GUM H2 example

H.2 Simultaneous resistance and reactance measurement

This example demonstrates the treatment of multiple measurands or output quantities determined simultaneously in the same measurement and the correlation of their estimates. It considers only the random variations of the observations; in actual practice, the uncertainties of corrections for systematic effects would also contribute to the uncertainty of the measurement results. The data are analysed in two different ways with each yielding essentially the same numerical values.

H.2.1 The measurement problem

The resistance R and the reactance X of a circuit element are determined by measuring the amplitude V of a sinusoidally-alternating potential difference across its terminals, the amplitude I of the alternating current passing through it, and the phase-shift angle ϕ of the alternating potential difference relative to the alternating current. Thus the three input quantities are V, I, and ϕ and the three output quantities — the measurands — are the three impedance components R, X, and Z. Since $Z^2 = R^2 + X^2$, there are only two independent output quantities.

H.2.2 Mathematical model and data

The measurands are related to the input quantities by Ohm's law:

$$R = \frac{V}{I}\cos\phi; \quad X = \frac{V}{I}\sin\phi; \quad Z = \frac{V}{I} \qquad \text{(H.7)}$$

Consider that five independent sets of simultaneous observations of the three input quantities V, I, and ϕ are obtained under similar conditions (see B.2.15), resulting in the data given in Table H.2. The arithmetic means of the observations and the experimental standard deviations of those means calculated from Equations (3) and (5) in 4.2 are also given. The means are taken as the best estimates of the expected values of the input quantities, and the experimental standard deviations are the standard uncertainties of those means.

http://www.bipm.org/utils/common/documents/jcgm/JCGM_100_2008_E.pdf

3.2.1 GUM H2 example – Input quantities

Table H.2 — Values of the input quantities V, I, and ϕ obtained from five sets of simultaneous observations

Set number k	Input quantities V (V)	I (mA)	ϕ (rad)
1	5,007	19,663	1,045 6
2	4,994	19,639	1,043 8
3	5,005	19,640	1,046 8
4	4,990	19,685	1,042 8
5	4,999	19,678	1,043 3
Arithmetic mean	$\bar{V} = 4{,}999\,0$	$\bar{I} = 19{,}661\,0$	$\bar{\phi} = 1{,}044\,46$
Experimental standard deviation of mean	$s(\bar{V}) = 0{,}003\,2$	$s(\bar{I}) = 0{,}009\,5$	$s(\bar{\phi}) = 0{,}000\,75$
Correlation coefficients			
$r(\bar{V},\bar{I}) = -0{,}36$ $r(\bar{V},\bar{\phi}) = 0{,}86$ $r(\bar{I},\bar{\phi}) = -0{,}65$			

3.2.2 Matlab – Definition of the inputs

```
unc = @LinProp;

meas = [5.007 19.663e-3 1.0456; ...
        4.994 19.639e-3 1.0438; ...
        5.005 19.640e-3 1.0468; ...
        4.990 19.685e-3 1.0428; ...
        4.999 19.678e-3 1.0433];

input_values = mean(meas, 1);
input_covar = cov(meas)./size(meas, 1);

inputs = unc(input_values, input_covar);
v = inputs(1);
i = inputs(2);
phi = inputs(3);
```

3.2.3 GUM H2 example – Output quantities

Table H.3 — Calculated values of the output quantities R, X, and Z: approach 1

Measurand index l	Relationship between estimate of measurand y_l and input estimates x_i	Value of estimate y_l, which is the result of measurement	Combined standard uncertainty $u_c(y_l)$ of result of measurement
1	$y_1 = R = (\overline{V}/\overline{I})\cos\overline{\phi}$	$y_1 = R = 127{,}732\ \Omega$	$u_c(R) = 0{,}071\ \Omega$ $u_c(R)/R = 0{,}06 \times 10^{-2}$
2	$y_2 = X = (\overline{V}/\overline{I})\sin\overline{\phi}$	$y_2 = X = 219{,}847\ \Omega$	$u_c(X) = 0{,}295\ \Omega$ $u_c(X)/X = 0{,}13 \times 10^{-2}$
3	$y_3 = Z = \overline{V}/\overline{I}$	$y_3 = Z = 254{,}260\ \Omega$	$u_c(Z) = 0{,}236\ \Omega$ $u_c(Z)/Z = 0{,}09 \times 10^{-2}$
Correlation coefficients $r(y_l, y_m)$			
$r(y_1, y_2) = r(R, X) = -0{,}588$ $r(y_1, y_3) = r(R, Z) = -0{,}485$ $r(y_2, y_3) = r(X, Z) = 0{,}993$			

3.2.4 Matlab – Compute the outputs

```
r = v./i.*cos(phi);
x = v./i.*sin(phi);
z = v./i;

outputs = [r x z];

output_values = get_value(outputs)
output_stdunc = get_stdunc(outputs)
output_corr = get_correlation(outputs)
```

3.2.5 Matlab – Output quantities

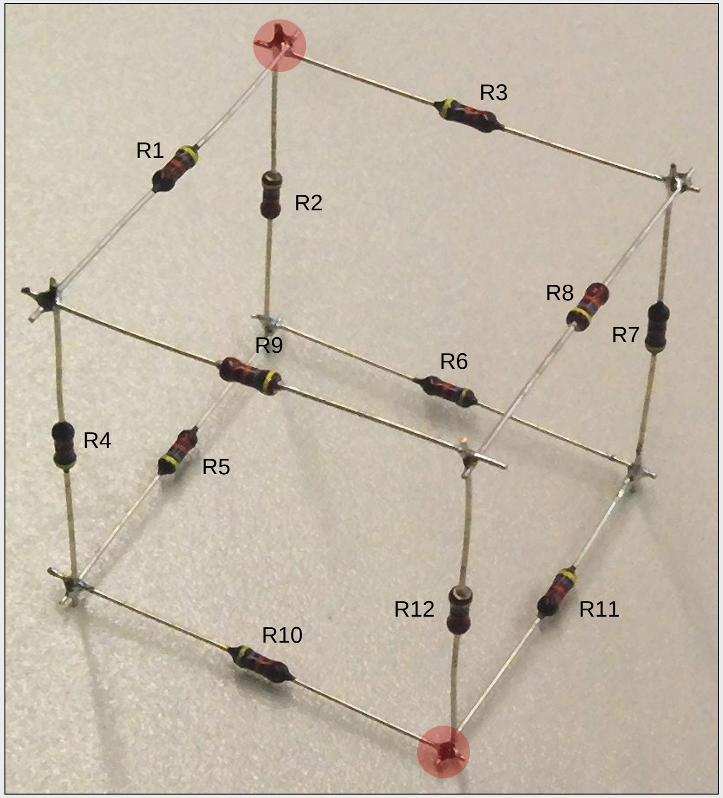
Table H.3 — Calculated values of the output quantities R, X, and Z: approach 1

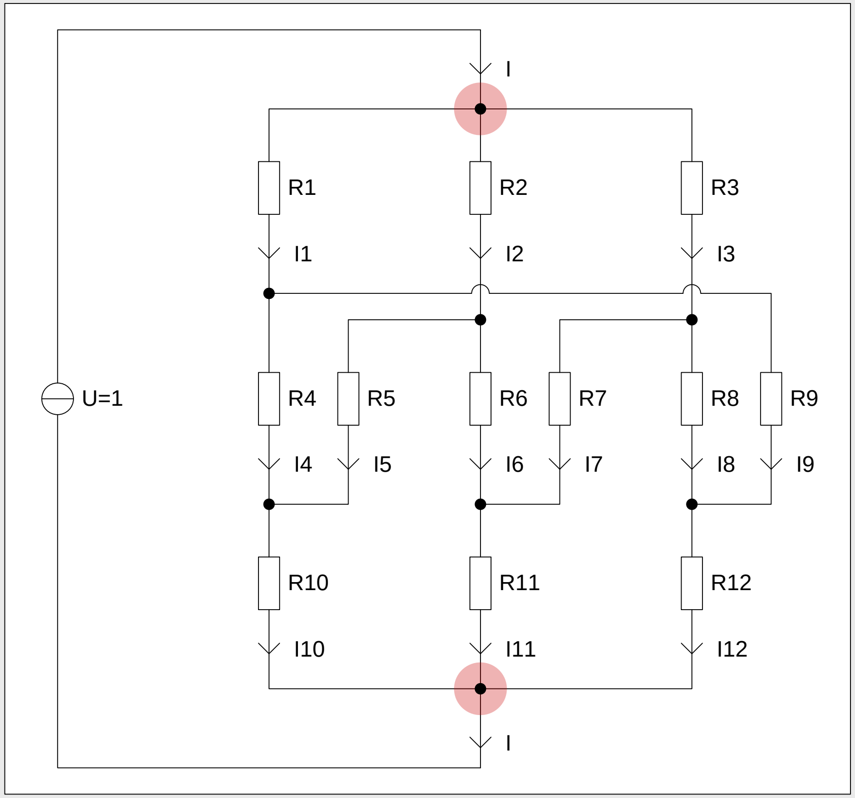
Measurand index l	Relationship between estimate of measurand y_l and input estimates x_i	Value of estimate y_l, which is the result of measurement	Combined standard uncertainty $u_c(y_l)$ of result of measurement
1	$y_1 = R = (\overline{V}/\overline{I})\cos\overline{\phi}$	$y_1 = R = 127{,}732\ \Omega$	$u_c(R) = 0{,}071\ \Omega$ $u_c(R)/R = 0{,}06 \times 10^{-2}$
2	$y_2 = X = (\overline{V}/\overline{I})\sin\overline{\phi}$	$y_2 = X = 219{,}847\ \Omega$	$u_c(X) = 0{,}295\ \Omega$ $u_c(X)/X = 0{,}13 \times 10^{-2}$
3	$y_3 = Z = \overline{V}/\overline{I}$	$y_3 = Z = 254{,}260\ \Omega$	$u_c(Z) = 0{,}236\ \Omega$ $u_c(Z)/Z = 0{,}09 \times 10^{-2}$
Correlation coefficients $r(y_l, y_m)$			
$r(y_1, y_2) = r(R, X) = -0{,}588$ $r(y_1, y_3) = r(R, Z) = -0{,}485$ $r(y_2, y_3) = r(X, Z) = 0{,}993$			

```
output_values =
      127.73      219.85      254.26
output_stdunc =
    0.071071     0.29558     0.23634
output_corr =
           1    -0.58843    -0.48526
    -0.58843           1     0.99251
    -0.48526     0.99251           1
```

3.3 Resistor cube example

- What‘s the equivalent resistor of the cube between the two red junctions?
- And the uncertainty of it?





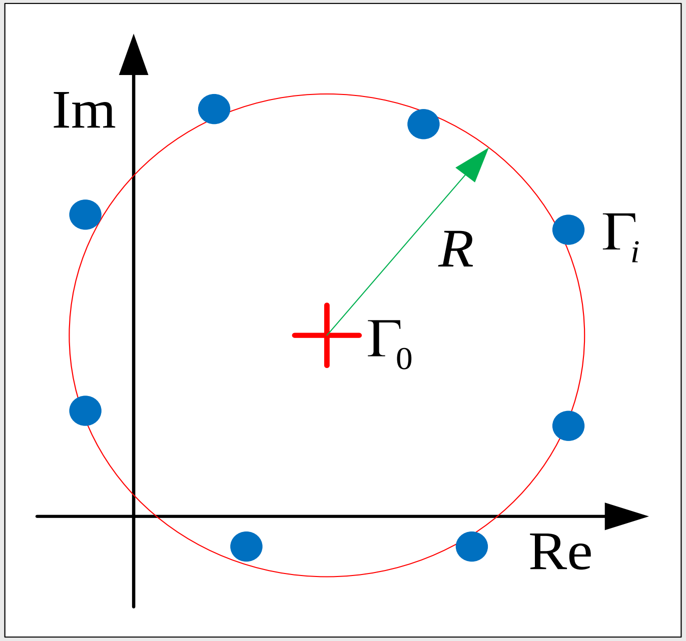
$$\underset{(12\times 1)}{U} = \underset{(12\times 12)}{R} \times \underset{(12\times 1)}{I} \rightarrow \underset{(12\times 1)}{I} = \underset{(12\times 12)}{R}^{-1} \times \underset{(12\times 1)}{U}$$

3.4 Circle fit example

- Where is the center of the circle? What is the uncertainty?

Equation for a circle:

$$\left|\Gamma_i - \Gamma_0\right|^2 = R^2 \quad i = 1, \ldots, n$$



Linearization:

$$2 \cdot \mathrm{Re}(\Gamma_i) \cdot \underbrace{\mathrm{Re}(\Gamma_0)}_{a} + 2 \cdot \mathrm{Im}(\Gamma_i) \cdot \underbrace{\mathrm{Im}(\Gamma_0)}_{b} + \underbrace{R^2 - \left|\Gamma_0\right|^2}_{c} = \left|\Gamma_i\right|^2$$

Circle fit equation system:

$$\mathrm{A} \times \mathrm{p} = \mathrm{d}$$

over determined

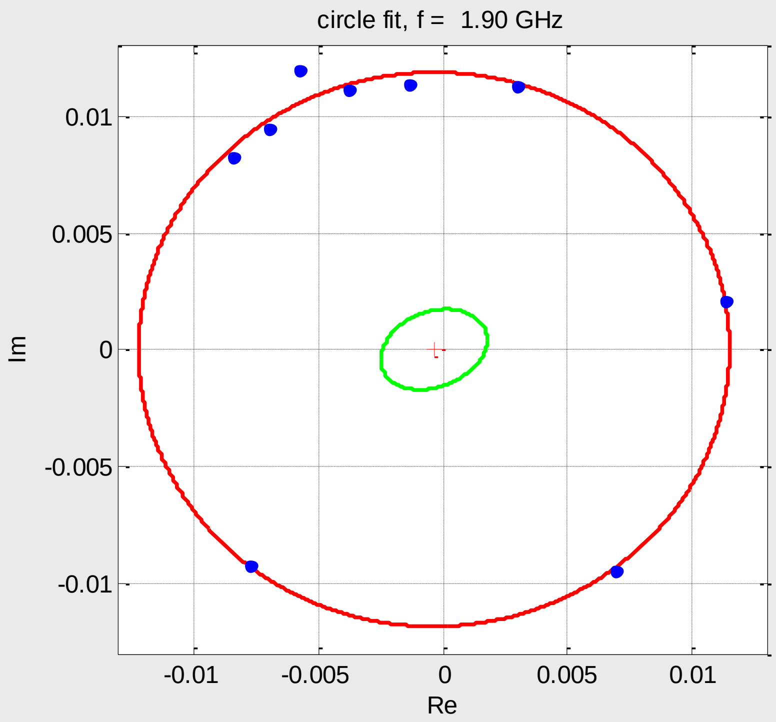
$$\mathrm{A} = \begin{bmatrix} 2 \cdot \mathrm{Re}(\Gamma_1) & 2 \cdot \mathrm{Im}(\Gamma_1) & 1 \\ 2 \cdot \mathrm{Re}(\Gamma_2) & 2 \cdot \mathrm{Im}(\Gamma_2) & 1 \\ \vdots & \vdots & \vdots \\ 2 \cdot \mathrm{Re}(\Gamma_n) & 2 \cdot \mathrm{Im}(\Gamma_n) & 1 \end{bmatrix} \quad \mathrm{p} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \quad \mathrm{d} = \begin{bmatrix} \left|\Gamma_1\right|^2 \\ \left|\Gamma_2\right|^2 \\ \\ \left|\Gamma_n\right|^2 \end{bmatrix}$$

$$\Gamma_0 = a + b \cdot \mathrm{i} \qquad R = \sqrt{c + a^2 + b^2}$$

- And how to compute the uncertainty? The analytic solution is too complicated. Too much data for Monte Carlo.
  → Metas.UncLib is the ideal solution!

3.4.1 Circle fit example – Matlab

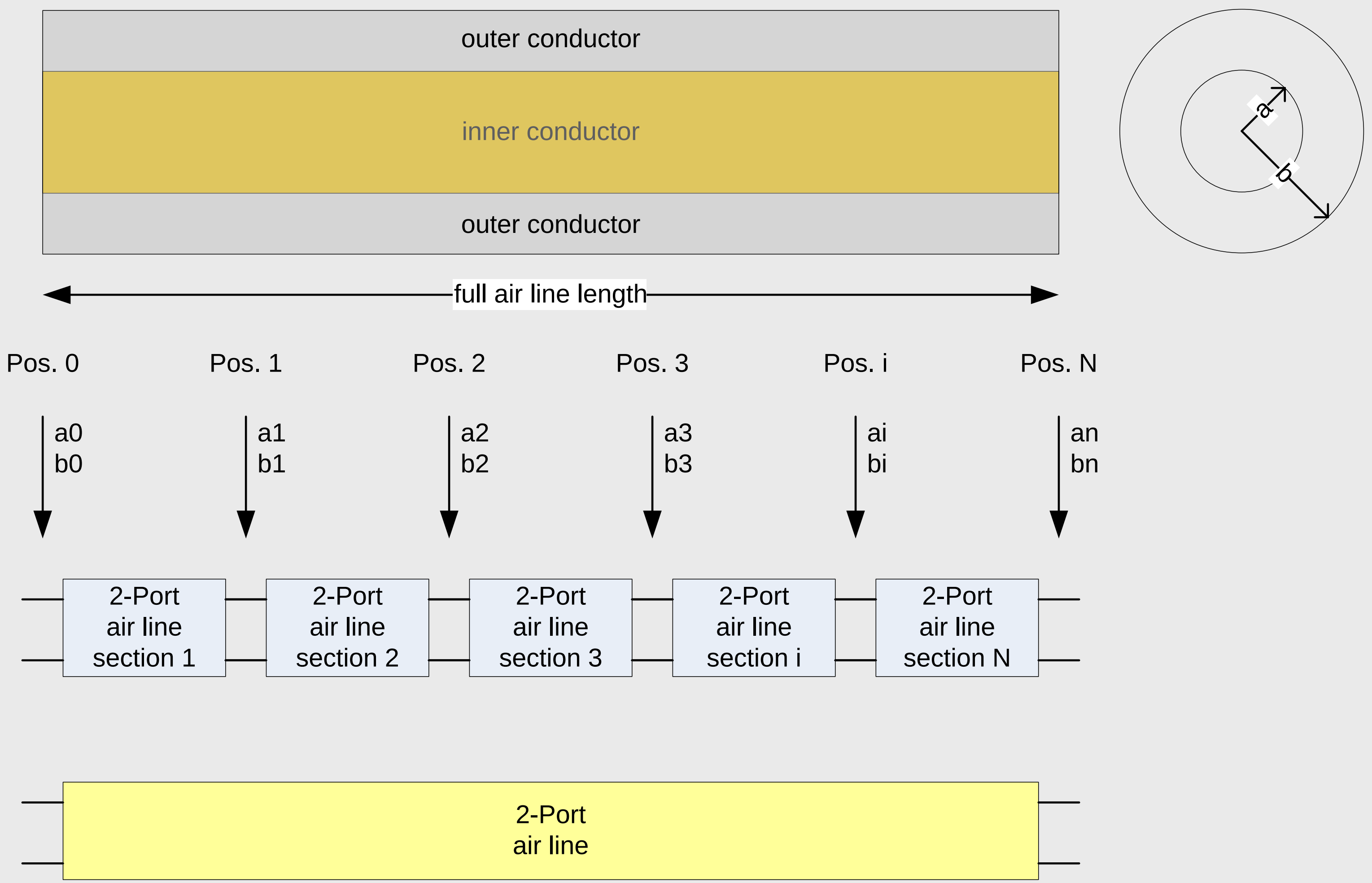
```
unc = @LinProp;
%% Circle points
f = 1.9; % (GHz)

V = diag([6.9e-4^2, 6.9e-4^2]);

s1 = unc(-8.43E-3 +8.28E-3*i, V);
s2 = unc(-6.99E-3 +9.43E-3*i, V);
s3 = unc(-3.73E-3 +1.11E-2*i, V);
s4 = unc( 3.00E-3 +1.13E-2*i, V);
s5 = unc( 1.14E-2 +2.11E-3*i, V);
s6 = unc( 6.92E-3 -9.43E-3*i, V);
s7 = unc(-7.75E-3 -9.27E-3*i, V);
s8 = unc(-5.76E-3 +1.20E-2*i, V);
s9 = unc(-1.31E-3 +1.14E-2*i, V);

s = [s1 s2 s3 s4 s5 s6 s7 s8 s9].';

%% Circle fit
A = [2.*real(s) 2.*imag(s) ones(size(s))];
d = abs(s).^2;
p = A\d;
S0 = p(1) + i*p(2);
R = sqrt(p(3) + abs(S0).^2);
```




```
S0.value =
  -0.00035669 +3.0153e-005i


S0.covariance =
  1.8235e-007   3.6297e-008
  3.6297e-008   1.1865e-007
```

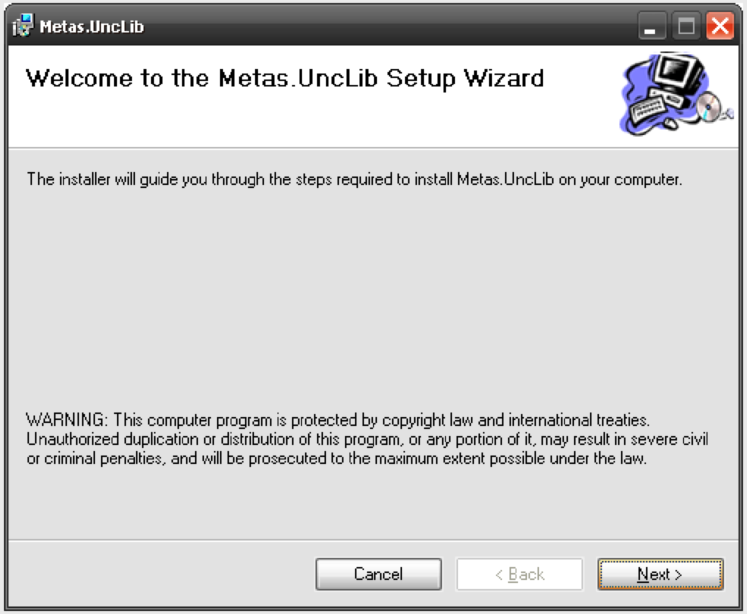
3.5 Air line characterization example

- S-Parameter of an air line in function of mechanical data.



Get Metas.UncLib now!

- To get Metas.UncLib please contact Michael Wollensack michael.wollensack@metas.ch
- It’s distributed in two setup files (*.msi). One for Metas.UncLib the other one for Metas.UncLib.Matlab



- To use it from Matlab you need at least Version 2006b.

Thank you for your attention …

The VNA Tools II Team in action …

METAS

Questions