# OptiY 4.7

## Data-driven Modeling and Simulation
## based on Physics-informed Machine Learning

October 2023 - OptiY GmbH

# Hilbert Space

## User-friendly Framework for Machine Learning and Modeling (Regression and Classification = Analog and Digital Simulation)

- **Polynomial Vector**

$$\phi_d = \{1, x_d, x_d^2, x_d^3, \ldots, x_d^n\}$$

- **Dirichlet Kernel Vector**

$$\phi_d = \{1, \cos(x_d), \cos(2x_d), \cos(3x_d), \ldots, \cos(nx_d)\}$$

- **User-Defined Vector**

$$\phi_d = \{f_1(x_d), f_2(x_d), f_3(x_d), f_4(x_d), \ldots, f_n(x_d)\}$$

$$\boldsymbol{\phi}(\boldsymbol{x}) = \boldsymbol{\phi_1} \otimes \boldsymbol{\phi_2} \otimes \boldsymbol{\phi_3} \ldots \otimes \boldsymbol{\phi_d}$$

$$\boldsymbol{x} = \{x_1, x_2, x_3, , \ldots x_d\}$$

| Hilbert Space | |
|---|---|
| Uniform Space | ☑ |
| Covariance Function | User Defined |
| Number of Features | 9 |
| 1 | 1 |
| 2 | cos(p1*x1) |
| 3 | cos(p2*x2) |
| 4 | cos(2*p1*x1) |
| 5 | cos(2*p2*x2) |
| 6 | cos(3*p1*x1) |
| 7 | cos(3*p2*x2) |
| 8 | cos(4*p1*x1) |
| 9 | cos(4*p2*x2) |
| Optimization Parameter | ☑ |
| Number of Parameters | 2 |
| Parameter 1 | |
| Name | p1 |
| Value | 1 |
| Lower Boundary | 0 |
| Upper Boundary | 2 |
| Parameter 2 | |
| Name | p2 |
| Value | 1 |
| Lower Boundary | 0 |
| Upper Boundary | 2 |
| Gaussian Noise [%] | 0.01 |
| Approximation Error [%] | 0 |

# Linear and Nonlinear Solvers for Hilbert Space

- **Reproducing Kernel Hilbert Space**

$$k(\mathbf{x}, \mathbf{x}') = \sum \phi(\mathbf{x}).\, \phi(\mathbf{x}')$$

$$L = \log|\mathbf{K}| + \mathbf{y}^T \mathbf{K}^{-1} \mathbf{y}$$

Marginal Likelihood Function

- **Nonlinear Method (Neural Networks)**

$$L = \sum_{i=1}^{m} (\phi(\mathbf{x}_i).\, \boldsymbol{\beta} - y_i)^2 + R$$

Loss Function L + Regularization R
Nonlinear Least-Square Method

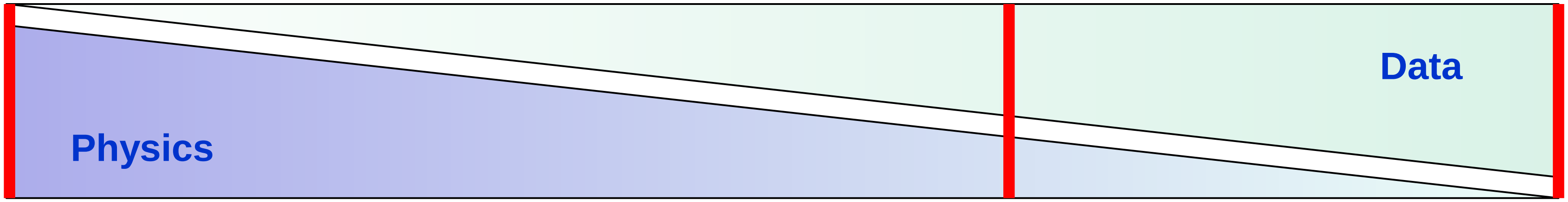
## New Optimization Methods

L-BFGS
Stochastic Gradient Descent
Gauss-Newton

| | |
|---|---|
| **⊟ Kernel Method** | |
| Max. Order | 10 |
| Noise-Optimization | ☐ |
| Optimization Method | Gradient Based |
| Max. Iterations | 30 |
| **⊟ Nonlinear Method** | |
| Include Hilbert Space | ☐ |
| Weight Optimization | ☐ |
| Regularization | None |
| Optimization Method | L-BFGS |
| Max. Iterations | 50 |

# Physics-informed Machine Learning (PIML)

| Physics-Model | Data-Model |
| --- | --- |
| • Physical laws by partial differential equations, boundary/initial conditions and constraints.<br>• Model validation through some measurement data possible.<br>• Serial implementation and long computing time. | • Measurement data from prototype. (huge data required)<br>• Physical laws not necessary.<br>• Machine learning (regression and classification) automatically.<br>• Parallel implementation and real-time computing. |



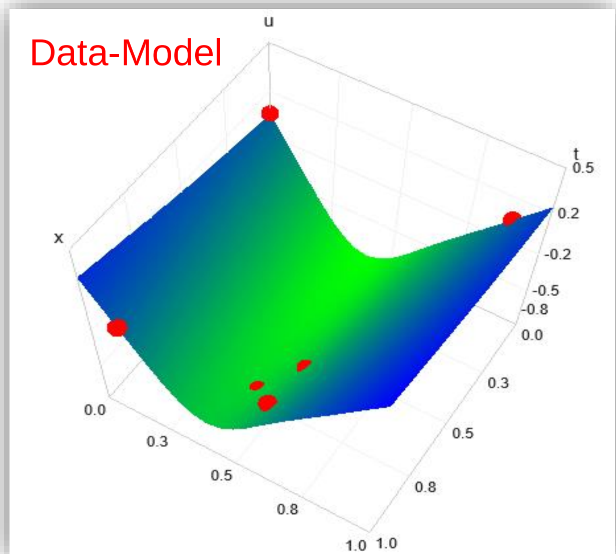

**Data-driven Modeling and Simulation based on PIML**

- Any mix from some data and some physical components
- Parallel Implementation and real-time computing

OptiY

Multidisciplinary Analysis and Optimization

# Case Comparison: Data vs. PIML

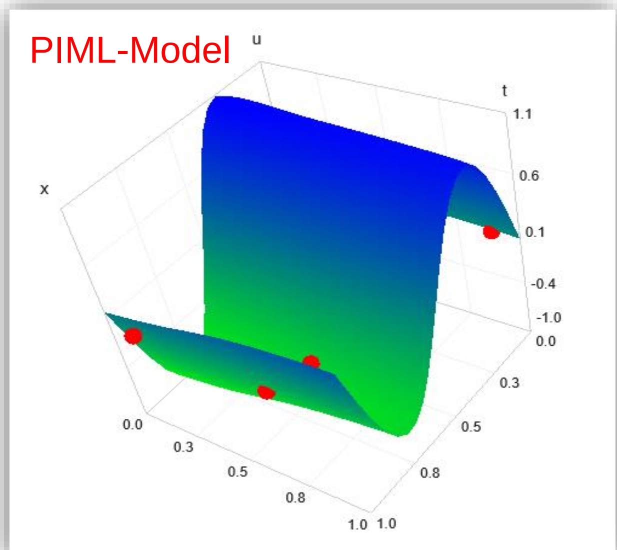



| Hilbert Space | |
|---|---|
| Uniform Space | ☑ |
| Covariance Function | Matérn Class 3/2 |
| Number of Features | 12 |
| Gaussian Noise [%] | 0.01 |
| Approximation Error [%] | 0 |
| State Variables | ☐ |
| Partial Differential Equation | ☐ |
| Boundary Conditions | ☐ |
| Constraints | ☐ |
| Parameters | ☐ |

Data

No Physics

Data-model from only some data points is inaccurate and not useable.




| Hilbert Space | |
|---|---|
| Uniform Space | ☑ |
| Covariance Function | Matérn Class 3/2 |
| Number of Features | 12 |
| Gaussian Noise [%] | 0.01 |
| Approximation Error [%] | 0 |
| State Variables | ☐ |
| Partial Differential Equation | ☑ |
| PDE | derivate(u,t)-derivate(u,x,x)=exp(-t)*(4*π*π-1)*sin(2*π*x) |
| Linear | ☑ |
| Sampling Level | 10 |
| User Defined | ☐ |
| Boundary Conditions | ☑ |
| Number of Boundaries | 1 |
| Boundary 1 | |
| Boundary Function | u=0 |
| Number of fixed Values | 1 |
| Fixed Parameter | x |
| Value | 0 |
| Sampling Level | 10 |
| User Defined | ☐ |
| Constraints | ☐ |
| Parameters | ☐ |

Same Data

Partial Differential Equation

Boundary Condition

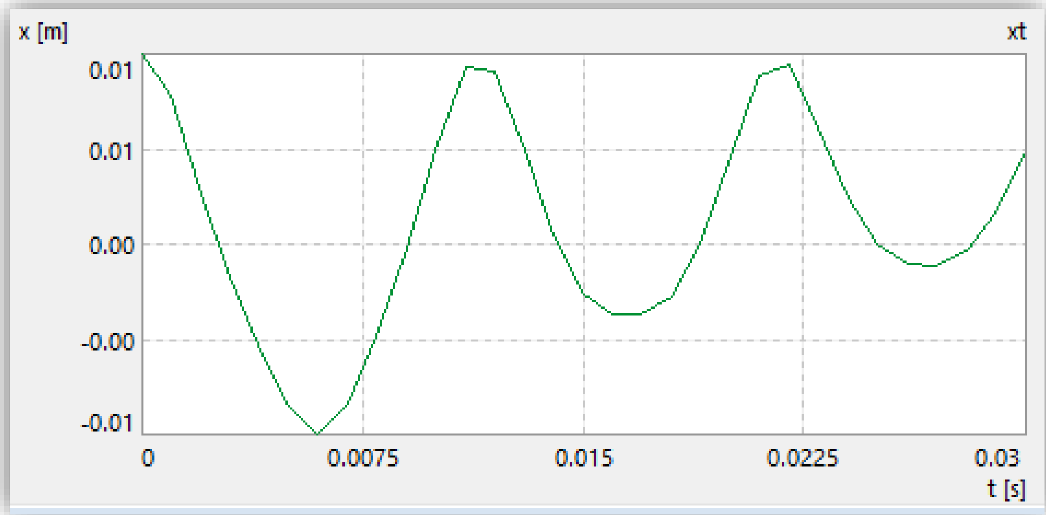
PIML-model from same data and some physical components is accurate and useable

www.optiy.eu

# Dynamical Systems / 1D-Systems

- Combination from machine learning and numerical integration (Euler, Heun, Runga-Kutta)
- Strong nonlinear systems
- Multiphysics with different disciplinary fields (Heat, Fluid, Static, Current, Energy etc.)
- Interactions between different partial systems

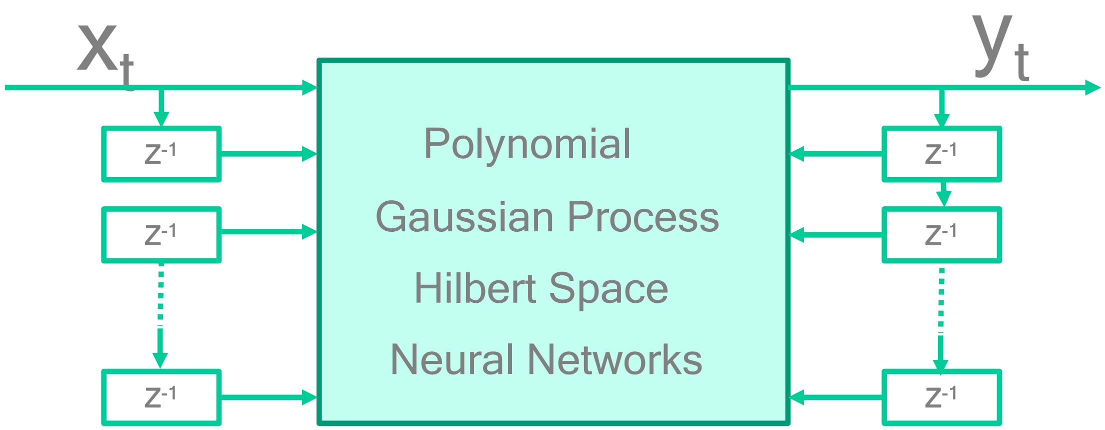
| 1D-Variables | |
|---|---|
| X-Max | 0.03 |
| X-Step | 0.0001 |
| X-Integration | Euler |




| Hilbert Space | |
|---|---|
| Include X-Axis | ☑ |
| Include 1D-Variables | ☐ |
| X-Nonlinearity | None |
| Uniform Space | ☐ |
| k | |
| Covariance Function | Polynomial |
| Order of Feature | 2 |
| t | |
| Covariance Function | Exponential |
| Number of Features | 12 |
| Gaussian Noise [%] | 0.01 |
| Approximation Error [%] | 0 |
| State Variables | ☐ |
| Partial Differential Equation | ☑ |
| PDE | m/k*derivate(xt,t,t)+c/k*derivate(xt,t)+xt=1/k |
| Linear | ☑ |
| Sampling Level | 10 |
| User Defined | ☐ |
| Boundary Conditions | ☑ |
| Number of Boundaries | 2 |
| Boundary 1 | |
| Initial Value | xt=0.02 |
| Number of fixed Values | 1 |
| Fixed Parameter | t |
| Boundary 2 | |
| Initial Value | derivate(xt,t)=0 |
| Number of fixed Values | 1 |
| Fixed Parameter | t |
| Constraints | ☐ |
| Parameters | ☐ |

# Nonlinear Autoregressive Exogenous Model (NARX) for 1D-Systems

- Autoregressive components with exogenous variables
- Specific structure and architecture depending on the problem.
- 1D-modeling for strong nonlinear systems
- Data-driven discovery of partial differential equations




| Hilbert Space | |
|---|---|
| Include X-Axis | ☐ |
| Include 1D-Variables | ☑ |
| xt | ☐ |
| vt | ☐ |
| ut | ☑ |
| X-Nonlinearity | Autoregressive Exogenous Model |
| X-Input Order | 2 |
| All X-Input Orders | ☑ |
| X-Output Order | 2 |
| All X-Output Orders | ☑ |
| Uniform Space | ☑ |
| Covariance Function | Polynomial |
| Order of Feature | 2 |

$$y_t = f(y_{t-1}, , y_{t-2}, y_{t-3}, , \ldots, y_{t-m}, x_t, x_{t-1} x_{t-2} x_{t-3} \ldots x_{t-n})$$

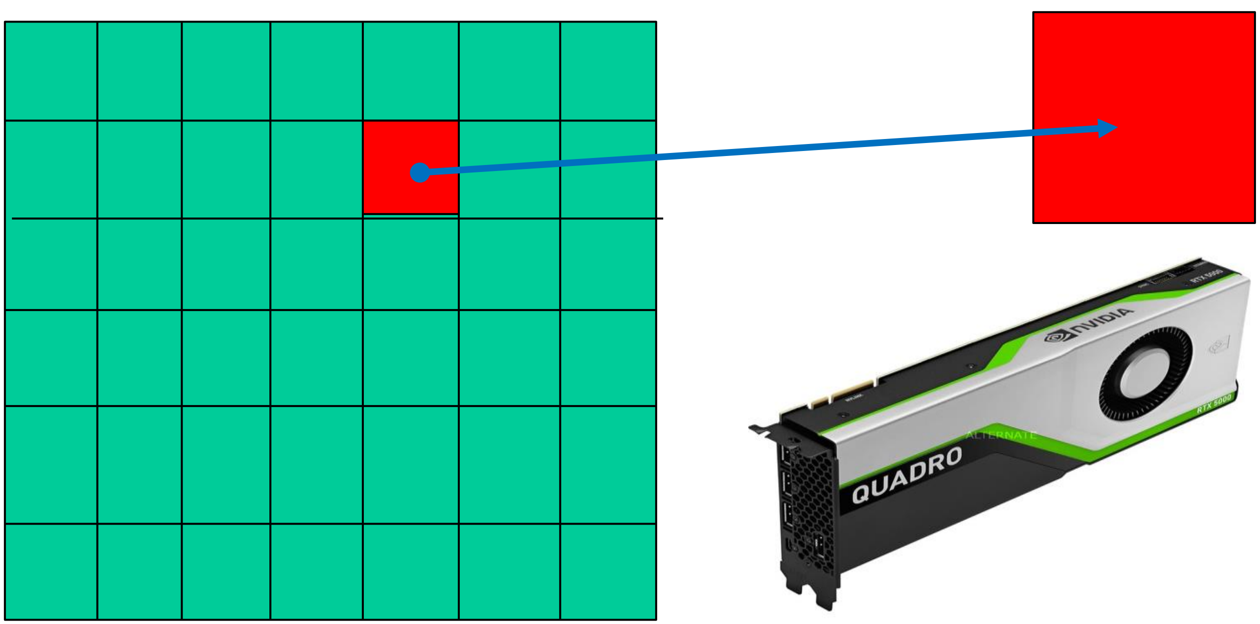
# Hierarchical Matrix for Big Data

- Big matrix on CPU divided in small hierarchical matrices loadable on GPU with small memory
- Matrix computing on small GPU-memory possible
- Fast machine learning for big data

| Hierachical Matrix | |
|---|---|
| Big Data | Automatic |
| Max. Matrix Size | 32 |

Big Matrix divided in Hierarchical Matrices on CPU

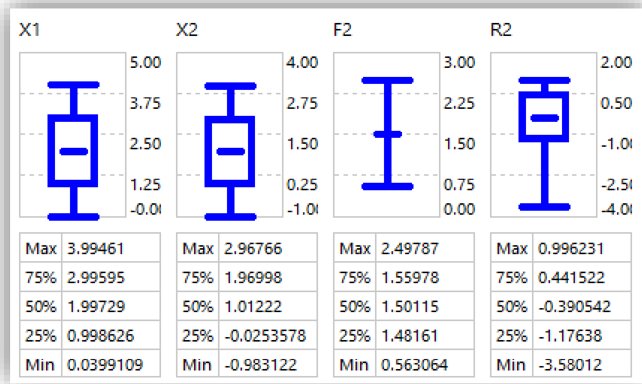
Small GPU Memory



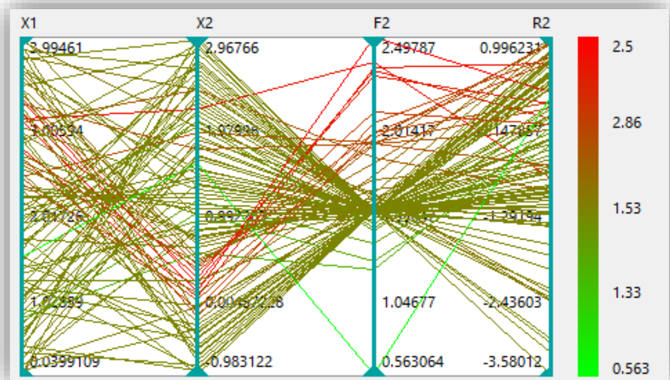
OptiY

Multidisciplinary Analysis and Optimization

# New Graphical Presentations (DirectX 12)



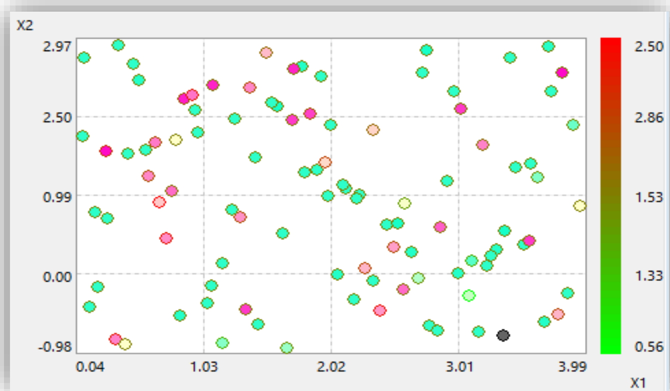

Box-Plot




Parallel Chart

| | X-Actuator-Position | | Y-Actuator-Position | | Max. Stress von Mises | |
|---|---|---|---|---|---|---|
| Plunger Length | 3.88 | 3.88 | 10.20 | 10.20 | 16.27 | 16.29 |
| Bracket High 1 | 0.00 | 0.00 | 0.07 | 0.07 | 0.31 | 0.32 |
| Bracket High 2 | 6.36 | 6.36 | 13.75 | 13.75 | 1.73 | 1.74 |
| Link 2 Length | 24.46 | 24.46 | 48.05 | 48.05 | 47.85 | 47.92 |
| Piston Length | 0.00 | 0.00 | 0.05 | 0.05 | 0.41 | 0.41 |
| Link 1 Length | 0.00 | 0.00 | 0.07 | 0.07 | 0.34 | 0.34 |
| Casing Length | 56.17 | 56.17 | 0.00 | 0.00 | 0.20 | 0.20 |

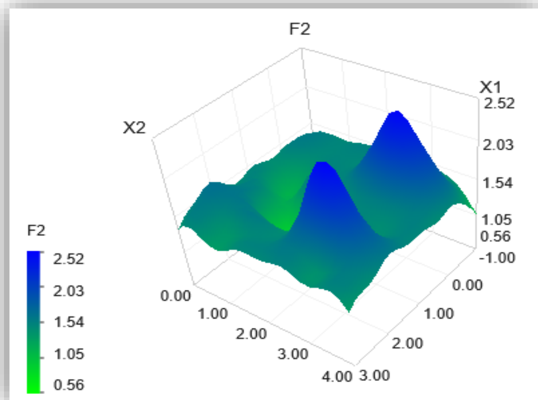
Sensitivity Matrix



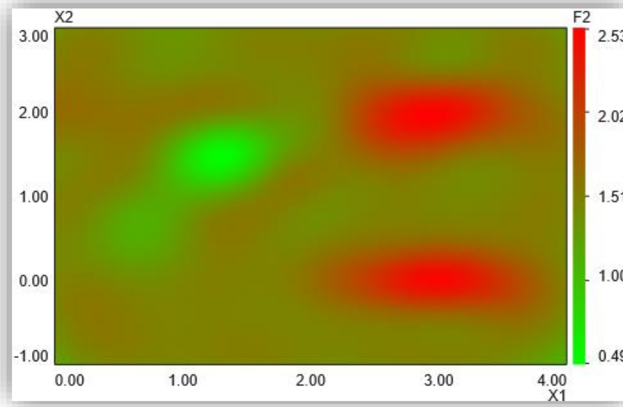

2D Scatter-Plot

www.optiy.eu

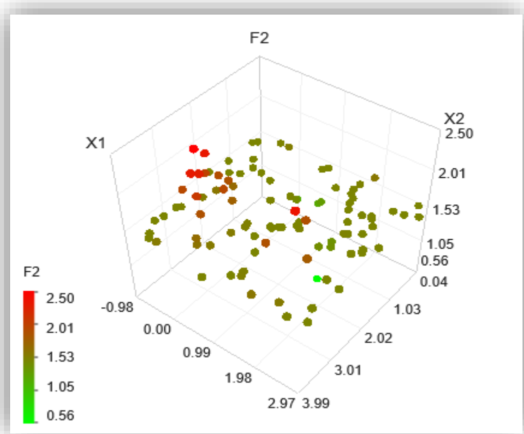
# New Graphical Presentations (DirectX 12)



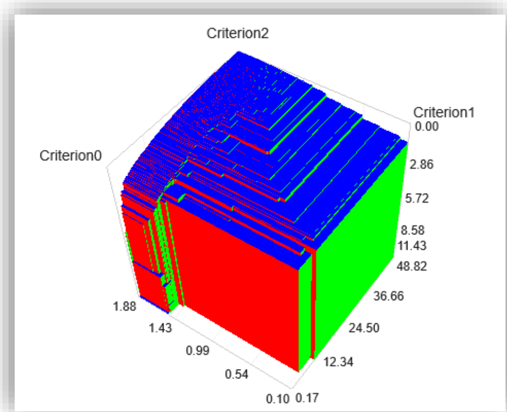
3D Surface



2D Surface

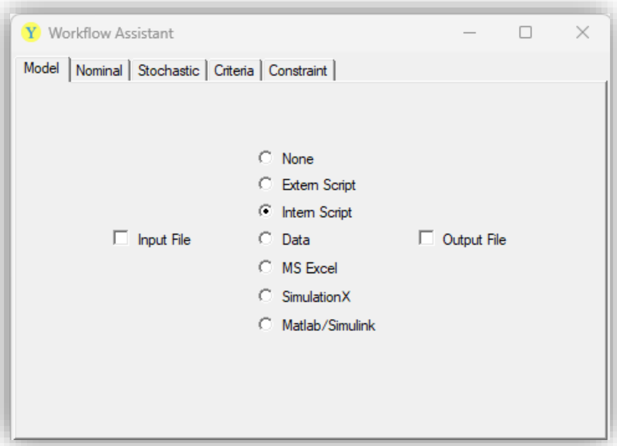


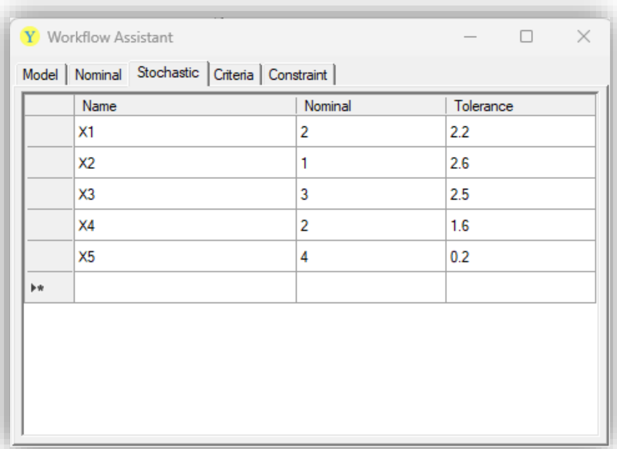
3D Scatter-Plot

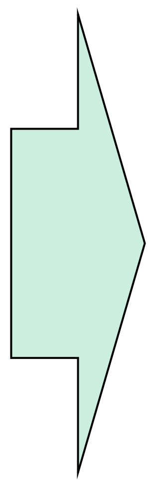


3D Hypervolume

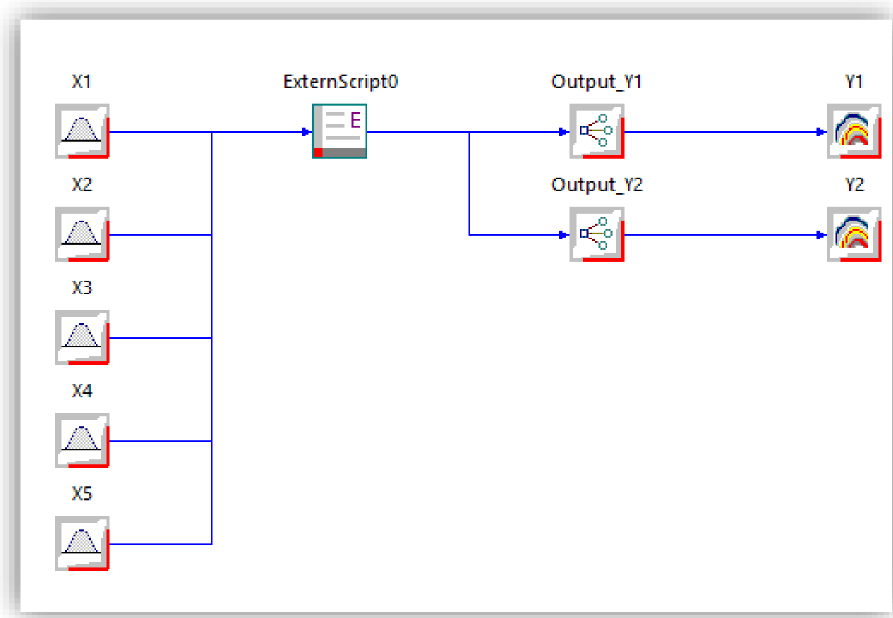
# Workflow Assistant



## Automatic Workflow Creation



# Many Small Enhancements

- User-defined sampling on the meta-model and save in data-table
- Selection of input parameters for metamodel
- String as model parameter for “Input File”
- PowerShell is the new scripting for “Extern Script”
- Debug-mode for “Output File” with the option “Show Failure”
- Differential evolution method is implemented
- Improved Hooke-Jeeves method
- User-defined start values for evolution strategies and differential evolution
- New design of experiment method “2n+1” is implemented
- Data import in “Nominal- and Stochastic-Editor” for parameter values
- Data export for correlation matrix
- Scatter-Plot can show data from different experiments as cluster