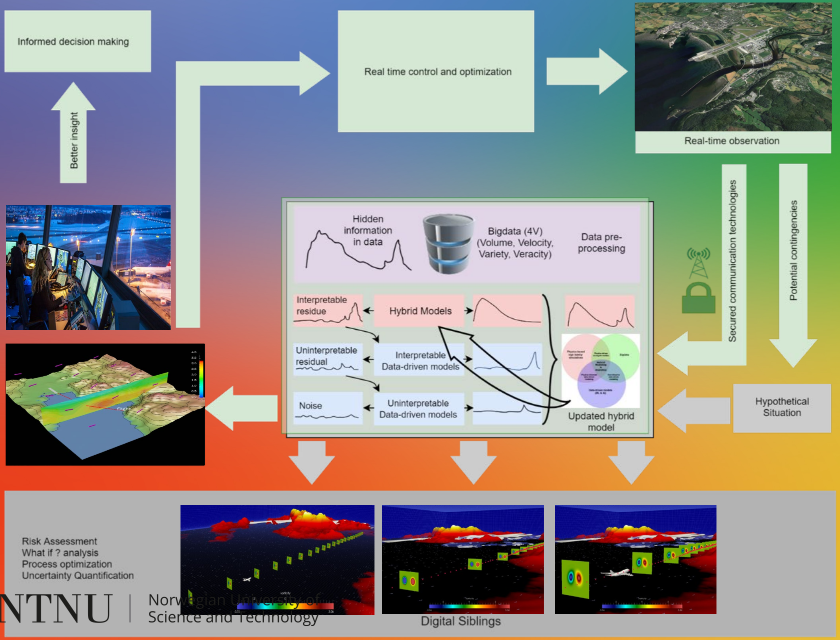
NTNU | Norwegian University of Science and Technology

# META MODELLING / HYBRID MODELLING

Adil RASHEED / 14.October.2021

www.hybridmodelling.com

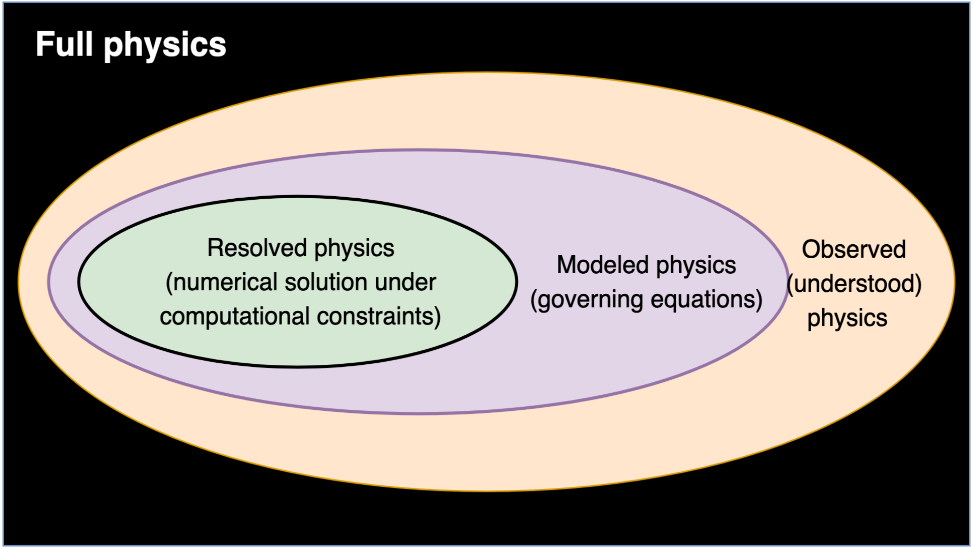
# DIGITAL TWIN

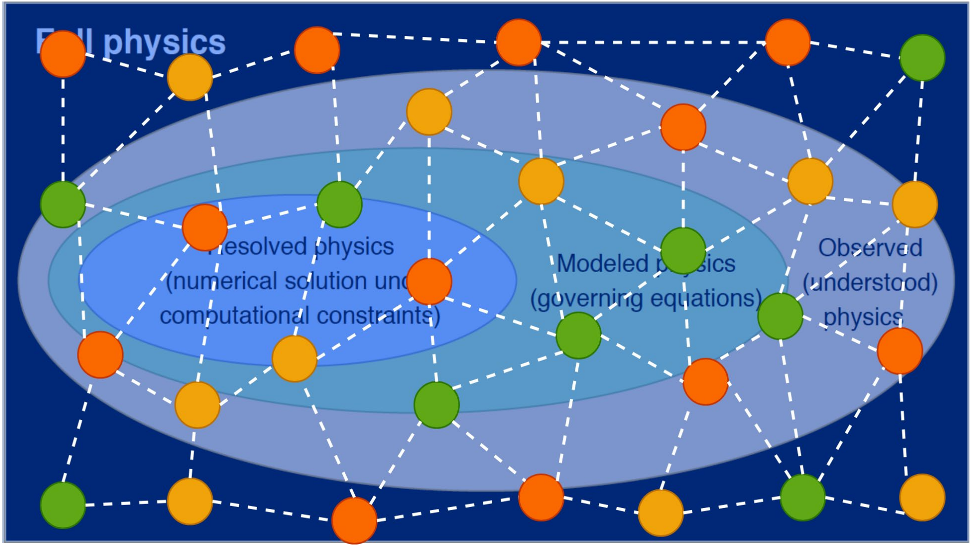



*A **digital twin** is defined as a virtual representation of a physical asset enabled through data and simulators for real-time prediction, monitoring, control and optimization of the asset for improved decision making throughout the life cycle of the asset and beyond.*

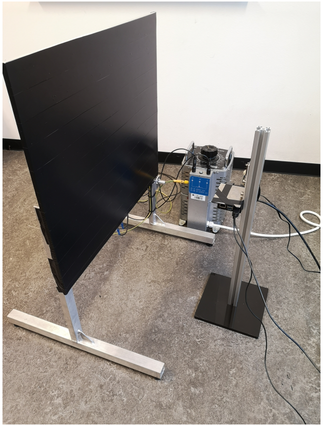
# Content - Models are needed

- Physics based modeling (PBM)
- Data-driven modeling (DDM)
- Hybrid / Meta modeling (MM)


Full physics
Resolved physics
(numerical solution under
computational constraints)
Modeled physics
(governing equations)
Observed
(understood)
physics


ll physics
esolved physics
(numerical solution un
computational constraints)
Modeled
(governing equations)
Observed
(understood)
physics

# Content - Example problem



Challenges: How to model the dynamics ?

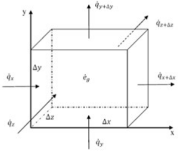
# PBM: - Physics-driven equation discovery

## Modeling

$$\dot{q}_{x+dx} = \dot{q}_x + \frac{\partial \dot{q}_x}{\partial x}\,dx$$

or,

$$\dot{q}_x - \dot{q}_{x+dx} = -\frac{\partial \dot{q}_x}{\partial x}\,dx$$



Fourier's law of heat conduction

$$\dot{q}_x = -k\,dy\,dz\,\frac{\partial T}{\partial x}$$

$$\boxed{\dot{q}_x - \dot{q}_{x+dx} = \frac{\partial}{\partial x}\left(k\,\frac{\partial T}{\partial x}\right) dx\,dy\,dz}$$

$$\dot{q}_y - \dot{q}_{y+dy} = \frac{\partial}{\partial y}\left(k\,\frac{\partial T}{\partial y}\right) dx\,dy\,dz$$

and,

$$\dot{q}_z - \dot{q}_{z+dz} = \frac{\partial}{\partial z}\left(k\,\frac{\partial T}{\partial z}\right) dx\,dy\,dz$$

$$\boxed{\begin{aligned} &\textit{Rate of heat input} + \textit{Rate of heat generation}\\ &\qquad = \textit{Rate of heat output}\\ &\qquad + \textit{Rate of change of heat energy within the body}\end{aligned}}$$

$$\dot{u} = (\rho dx\,dy\,dz)c_p\left(\frac{\partial T}{\partial t}\right)$$

$$(\dot{q}_x + \dot{q}_y + \dot{q}_z) + (\dot{e}_g\,dx\,dy\,dz) = (\dot{q}_{x+dx} + \dot{q}_{y+dy} + \dot{q}_{z+dz}) + (\rho dx\,dy\,dz)c_p\left(\frac{\partial T}{\partial t}\right)$$

$$(\dot{q}_x - \dot{q}_{x+dx}) + (\dot{q}_y - \dot{q}_{y+dy}) + (\dot{q}_z - \dot{q}_{z+dz}) + (\dot{e}_g dx\,dy\,dz) = (\rho dx\,dy\,dz)c_p\left(\frac{\partial T}{\partial t}\right)$$

or,

$$\left\{\frac{\partial}{\partial x}\left(k\,\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k\,\frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(k\,\frac{\partial T}{\partial z}\right)\right\}(dx\,dy\,dz) + (\dot{e}_g\,dx\,dy\,dz) = (\rho dx\,dy\,dz)c_p\left(\frac{\partial T}{\partial t}\right)$$

or,

$$\left\{\frac{\partial}{\partial x}\left(k\,\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k\,\frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(k\,\frac{\partial T}{\partial z}\right)\right\} + \dot{e}_g = \rho c_p\left(\frac{\partial T}{\partial t}\right)$$

# DDM: Data-driven equation discovery - Prerequisite

- Sequential threshold ridge regression (STRidge)
- Least absolute selection operator (LASSO)
- Computation of derivatives numerically

# DDM: Data-driven equation discovery - $L_1$ Regularization: Ridge

$$J(\beta) = \sum_{i=1}^{n}(y_i - \beta_o - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}\beta_j^2$$

# DDM: Data-driven equation discovery - $L_2$ Regularization: LASSO

$$J(\beta) = \sum_{i=1}^{n}(y_i - \beta_o - \sum_{j=1}^{p} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

# DDM: Data-driven equation discovery - Taylor Series expansion

$$u_{i+1} = u_i + \Delta x \frac{\partial u}{\partial x}|_i + \frac{\Delta x^2}{2}\frac{\partial^2 u}{\partial x^2}|_i + \frac{\Delta x^3}{3}\frac{\partial^3 u}{\partial x^3}|_i + O(\Delta x^4)$$

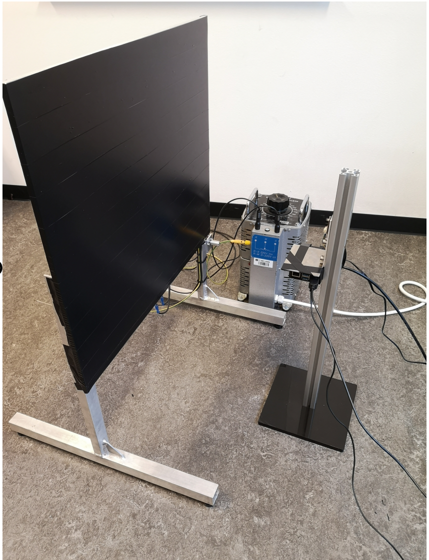
$$u_{i-1} = u_i - \Delta x \frac{\partial u}{\partial x}|_i + \frac{\Delta x^2}{2}\frac{\partial^2 u}{\partial x^2}|_i - \frac{\Delta x^3}{3}\frac{\partial^3 u}{\partial x^3}|_i + O(\Delta x^4)$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} + O(\Delta x^2)$$

$$\frac{\partial u}{\partial x} = \frac{u_i^{n+1} - u_i^n}{\Delta t}$$

# DDM: Data-driven equation discovery - Data sampling

$$\mathbf{u} = \left.\overbrace{\begin{bmatrix} u_1(t_1) & u_2(t_1) & \dots & u_n(t_1) \\ u_1(t_2) & u_2(t_2) & \dots & u_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ u_1(t_m) & u_2(t_m) & \dots & u_n(t_m) \end{bmatrix}}^{\text{spatial locations}}\right\} \text{time snap} \tag{1}$$



# DDM: Data-driven equation discovery - Feature Engineering

$$u_t = \frac{u_j^{n+1} - u_j^{n-1}}{2dt}$$

$$u_{2t} = \frac{u_j^{n+1} - 2u_j^n + u_j^{n-1}}{dt^2}$$

$$u_x = \frac{u_{j+1}^n - u_{j-1}^n}{2dx}$$

$$u_{2x} = \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{dx^2}$$

$$u_{3x} = \frac{u_{j+2}^n - 2u_{j+1}^n + 2u_{j-1}^n - u_{j-2}^n}{2dx^3}$$

$$u_{4x} = \frac{u_{j+2}^n - 4u_{j+1}^n + 6u_j^n + -4u_{j-1}^n - u_{j-2}^n}{dx^4}$$

$$u_{5x} = \frac{u_{j+3}^n - 4u_{j+2}^n + 5u_{j+1}^n - 5u_{j-1}^n + 4u_{j-2}^n - u_{j-3}^n}{2dx^5}$$

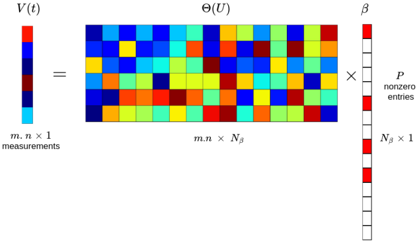
# DDM: Data-driven equation discovery - LASSO regression

$$\mathbf{V(t)} = [\mathbf{U}_t]$$

$$\tilde{\mathbf{\Theta}}(\mathbf{U}) = [\mathbf{U} \quad \mathbf{U}_x \quad \mathbf{U}_{2x} \quad \mathbf{U}_{3x} \quad \mathbf{U}_{4x} \quad \mathbf{U}_{5x}]$$

# DDM: Data-driven equation discovery - Methodology for more complex equations

$$\frac{\partial T}{\partial t} = \nu \left\{ \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right\} + S$$



# DDM: Data-driven equation discovery - Numerical Discretization 2D

$$\frac{\partial T}{\partial t} = \nu \left\{ \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right\}$$

$$\frac{T_{i,j}^{n+1} - T_{i,j}^{n}}{\Delta t} = \nu \frac{T_{i+1,j}^{n} - 2T_{i,j}^{n} + T_{i-1,j}^{n}}{\Delta x^2} + \nu \frac{T_{i,j+1}^{n} - 2T_{i,j}^{n} + T_{i,j-1}^{n}}{\Delta y^2}$$

$$T_{i,j}^{n+1} = T_{i,j}^{n} + \frac{\nu \Delta t}{\Delta x^2}(T_{i+1,j}^{n} - 2T_{i,j}^{n} + T_{i-1,j}^{n}) + \frac{\nu \Delta t}{\Delta y^2}(T_{i,j+1}^{n} - 2T_{i,j}^{n} + T_{i,j-1}^{n})$$

# MM / HAM - PBM vs DDM

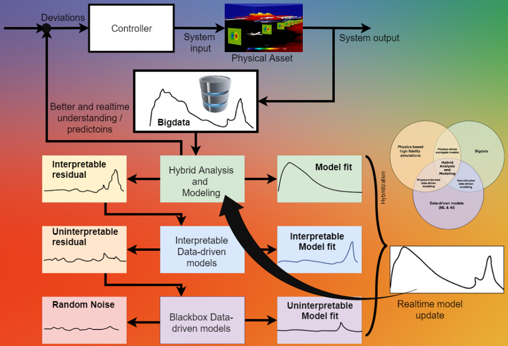
| PBM | DDM |
|---|---|
| +Solid foundation | − Mostly black-boxes |
| − Difficult to assimilate very long term historical data | + Takes into account long term historical data and experiences |
| − Computationally expensive, sensitive and susceptible to numerical instability | + Once the model is trained, it is very stable and efficient for making predictions / inferences |
| + Errors / uncertainties can be bounded and estimated | − Not possible to bound errors / uncertainties |
| + Less susceptible to bias | − Bias in data is reflected in the model prediction |
| + Generalizes well to new problems with similar physics | − Poor generalization on unseen problems |

# MM / HAM - Desired characteristics in a modeling approach

- Generalizability / Robustness
- Trustworthiness / Transparent / Explainable
- Computational efficiency yet accurate
- Dynamically adapting and evolving
- Automated physics discovery

# MM / HAM - Big Data Cybernetics philosophy

## Big Data Cybernetics

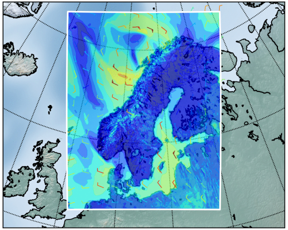


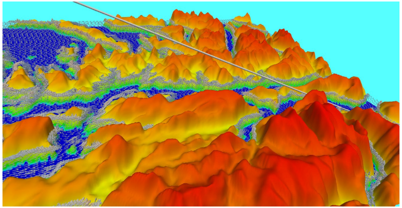

*A new paradigm in steering the world with big data characterized by high volume, high velocity, high vareity and high veracity*

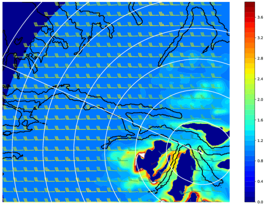
# MM / HAM - What and why ?

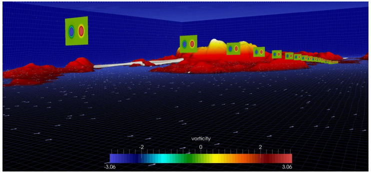
- Metamodeling is a simplified model created from a complex model
- The aim of metamodeling is to invest computational resources to develop fast mathematical approximations of sophisticated, computationally demanding models

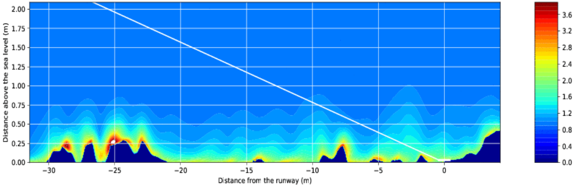
# Aircraft wake modeling



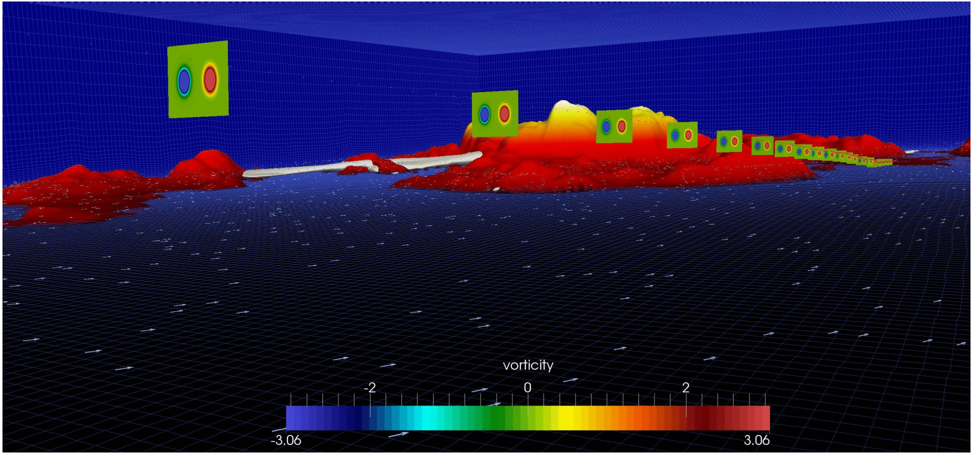




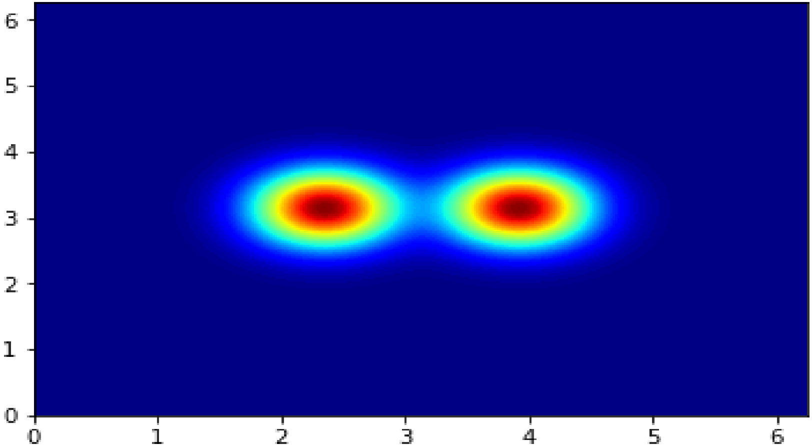


# MM / HAM - Motivation



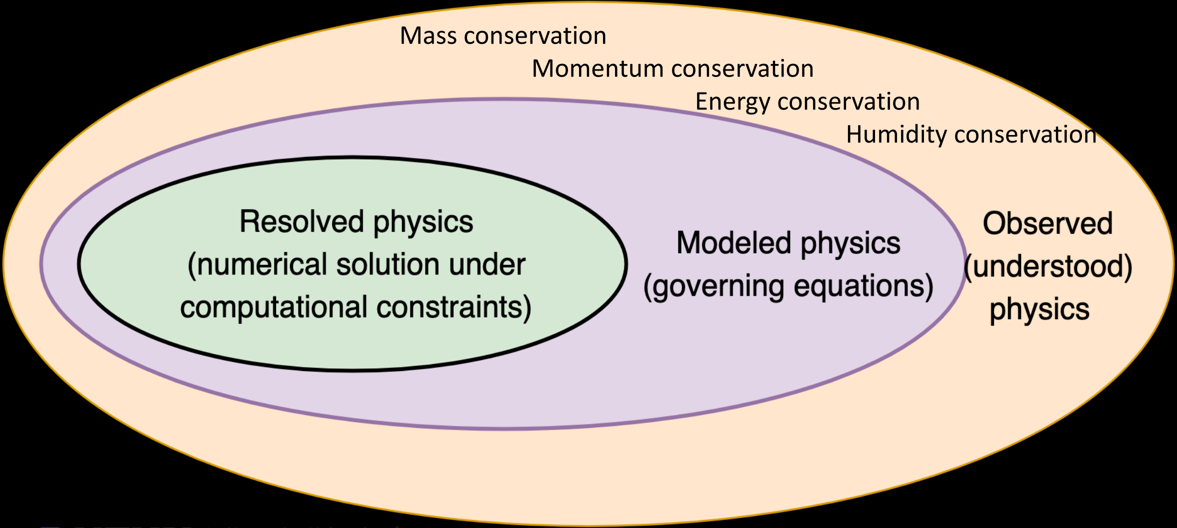
# MM / HAM - Aircraft wakes



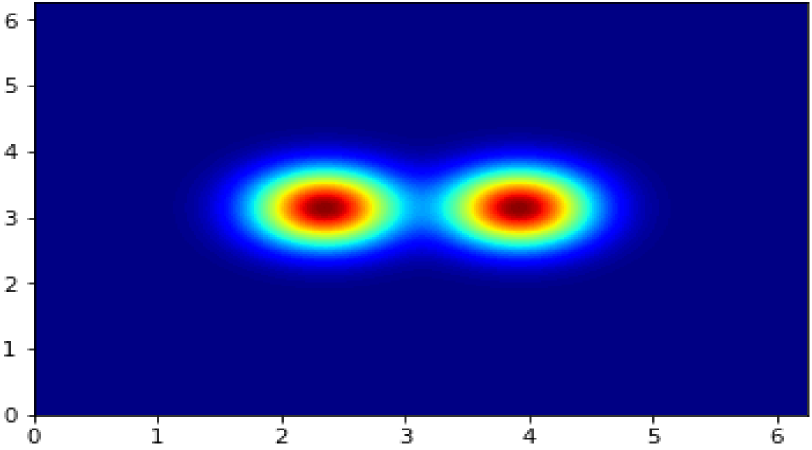
# MM / HAM - Step 1a: Physics based equation discovery

- Mass conservation equation
- Momentum conservation equation
- Energy conservation equation
- Humidity conservation equation
- Unknown physics

# Full physics



# MM / HAM - Step 1b: Data-driven equation discovery



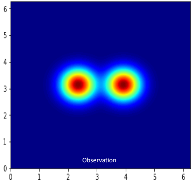
# MM / HAM - Discovered equation

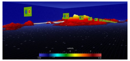
$$\frac{\partial \omega}{\partial t} + \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x} - \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y} = \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right) + \Pi$$

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = -\omega$$

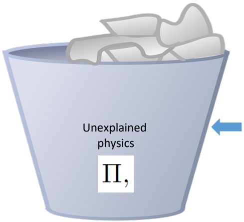
where $\omega$ is the vorticity defined as $\omega = \nabla \times u$, $u = [u, v]^T$ is the velocity vector, and $\psi$ is the stream function

# MM / HAM - Step 1: Physics based modeling



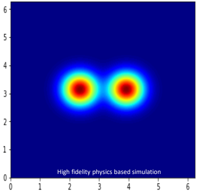
Snapshots using LIDARS and RADARS




$$\frac{\partial \omega}{\partial t} + J(\omega, \psi) = \frac{1}{Re}\nabla^2 \omega$$

$$J(\omega, \psi) = \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x} - \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y} \qquad \nabla^2 \psi = -\omega$$

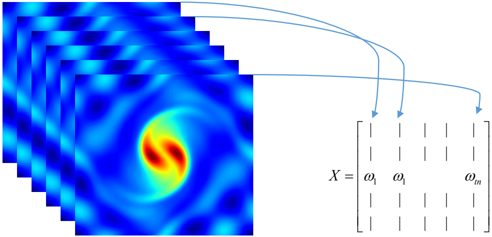
Generalization

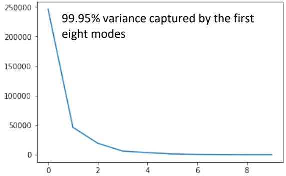


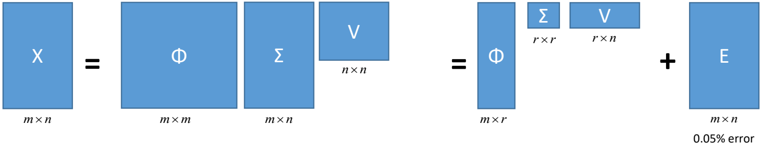

☺ Better Generalization
☹ Computationally expensive
☹ Inaccurate

Python notebook

# MM / HAM - Step 2: PCA: Computing the principal components

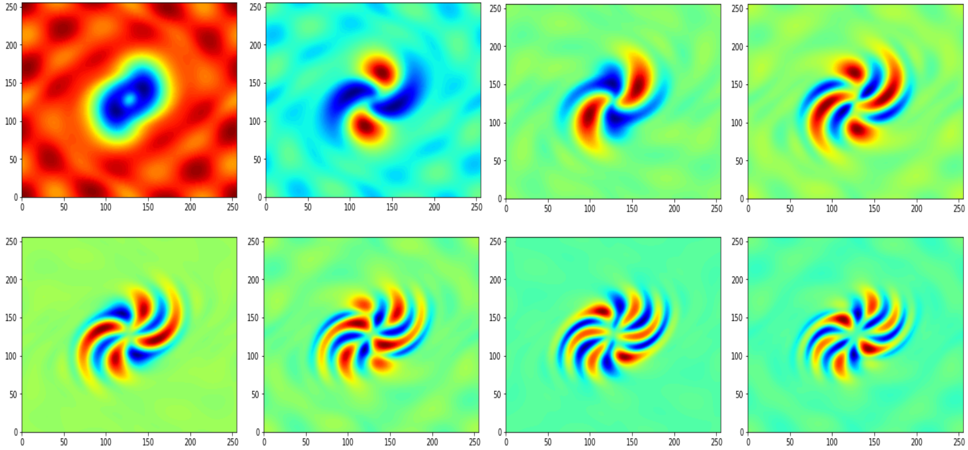



Python notebook

# MM / HAM - Orthonormal basis for a particular Reynolds Number



# MM / HAM - Different basis for different Reynolds Number

# MM / HAM - Step 3: Galerkin projection / dimensionality reduction

$$\frac{\partial \omega}{\partial t} + \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x} - \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y} = \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right) + \Pi$$

# MM / HAM - Step 3: Galerkin projection / dimensionality reduction

$$\frac{\partial \omega}{\partial t} + \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x} - \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y} = \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right) + \Pi$$

$$\omega\left(x, t_n\right) = \sum_{k=1}^{R} a_k(t_n)\phi_k(x)$$

# MM / HAM - Step 3: Galerkin projection / dimensionality reduction

$$\frac{\partial \omega}{\partial t} + \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x} - \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y} = \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right) + \Pi$$

$$\omega\left(x, t_n\right) = \sum_{k=1}^{R} a_k(t_n)\phi_k(x)$$

$$\frac{\mathrm{d}a_k}{\mathrm{d}t} = \underbrace{\sum_{i=1}^{R} \mathfrak{L}_{ik} a_i + \sum_{i=1}^{R}\sum_{j=1}^{R} \mathfrak{N}_{ijk} a_i a_j}_{\text{Physics-based model } G(a_k)} + \underbrace{\tilde{C}_k}_{\text{Hidden physics}}$$

# MM / HAM - Step 3: Galerkin projection / dimensionality reduction

$$\frac{\mathrm{d}a_k}{\mathrm{d}t} = \underbrace{\sum_{i=1}^{R} \mathfrak{L}_{ik} a_i + \sum_{i=1}^{R}\sum_{j=1}^{R} \mathfrak{N}_{ijk} a_i a_j}_{\text{Physics-based model } G(a_k)} + \underbrace{\tilde{C}_k}_{\text{Hidden physics}}$$

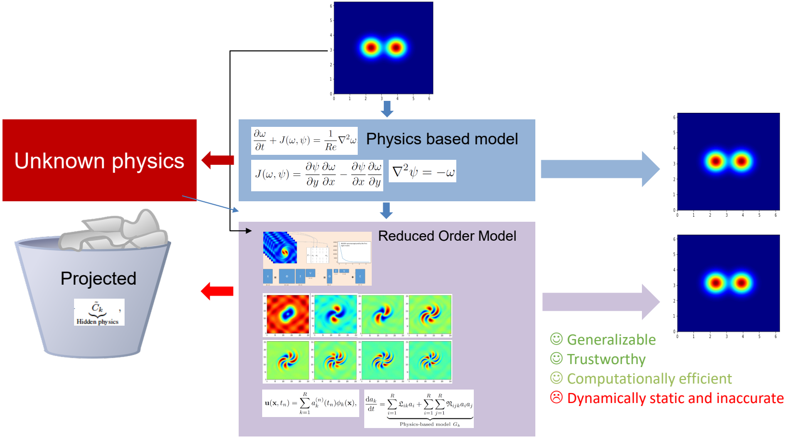
where

$$\mathfrak{L}_{ik} = \left\langle \frac{1}{\mathrm{Re}} \left( \frac{\partial^2 \phi_i^\omega}{\partial x^2} + \frac{\partial^2 \phi_i^\omega}{\partial y^2} \right), \phi_k^\omega \right\rangle$$

$$\mathfrak{N}_{ijk} = \left\langle -\left( \frac{\partial \phi_i^\omega}{\partial x}\frac{\partial \phi_j^\psi}{\partial y} - \frac{\partial \phi_i^\omega}{\partial y}\frac{\partial \phi_j^\psi}{\partial x} \right), \phi_k^\omega \right\rangle$$

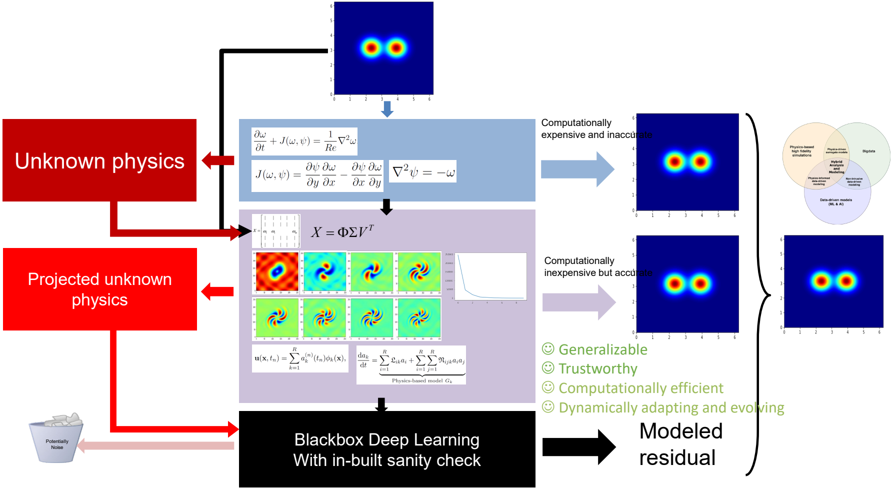
Python notebook

# MM / HAM – Results using GP-ROM

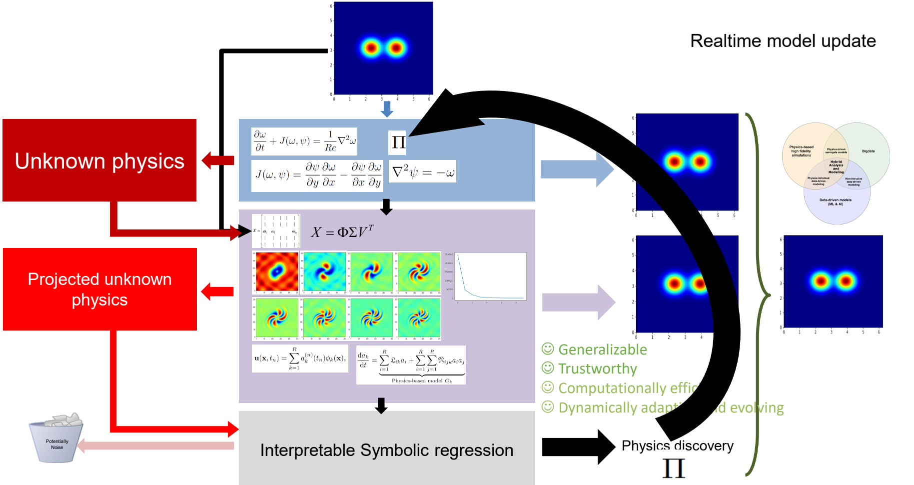



- ☺ Generalizable
- ☺ Trustworthy
- ☺ Computationally efficient
- ☹ Dynamically static and inaccurate

# MM / HAM – DL-based correction

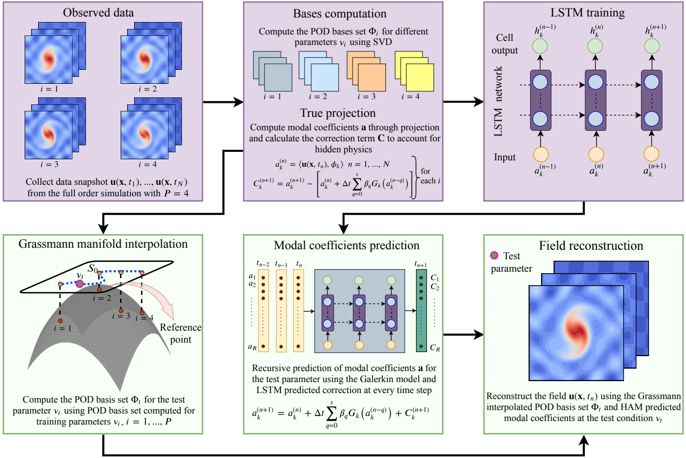


# MM / HAM - Model update




$$= 0.0099\ \sin(x)\ \cos(y)\ \exp(-0.078\ t) - 1.47 \times 10^{-6}$$

# MM / HAM – Full algorithm



Thank you for your attention

NTNU | Norwegian University of Science and Technology