# Physics-Informed Machine Learning for Monitoring Natural Hazards in Digital Twins


**Didier Georges**

*Diagnosis and Control Department – Infinity Team*


PROGRAMME DE RECHERCHE

RISQUES

**RISK Summer School – August 29, 2025, GreEn-ER**

*DITRISC project / TwinNaTech chair – MIAI@UGA*

# Outline

✓What is a **Digital Twin**? The DITRISC project

✓**Scientific challenges in modelling and simulation**, model classification

✓**Some examples of natural hazards**, PDE framework

✓**Synergy between models and data**: data assimilation

✓**Data for natural hazard** monitoring

✓A short introduction to **Deep Learning**, from data-based learning to Physics-Informed learning

✓**PINN approach**

✓**A case study**: A DT for flood monitoring

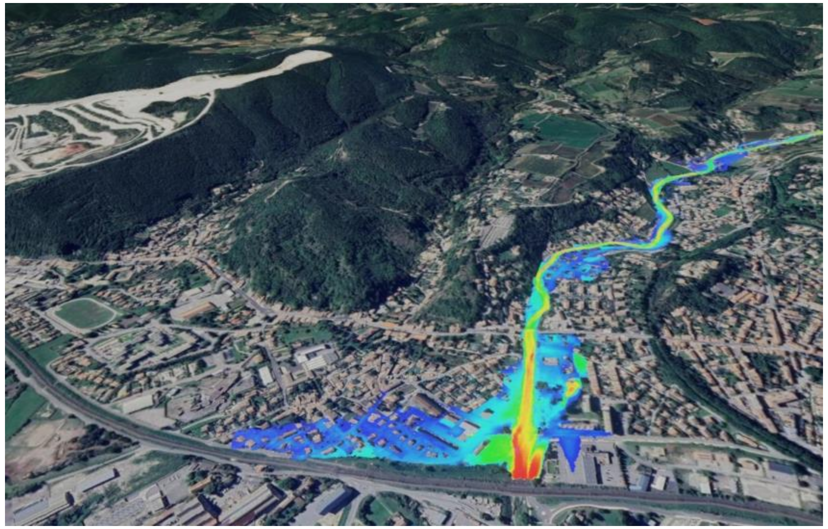
✓**Conclusions**

# Digital Twins for environmental applications

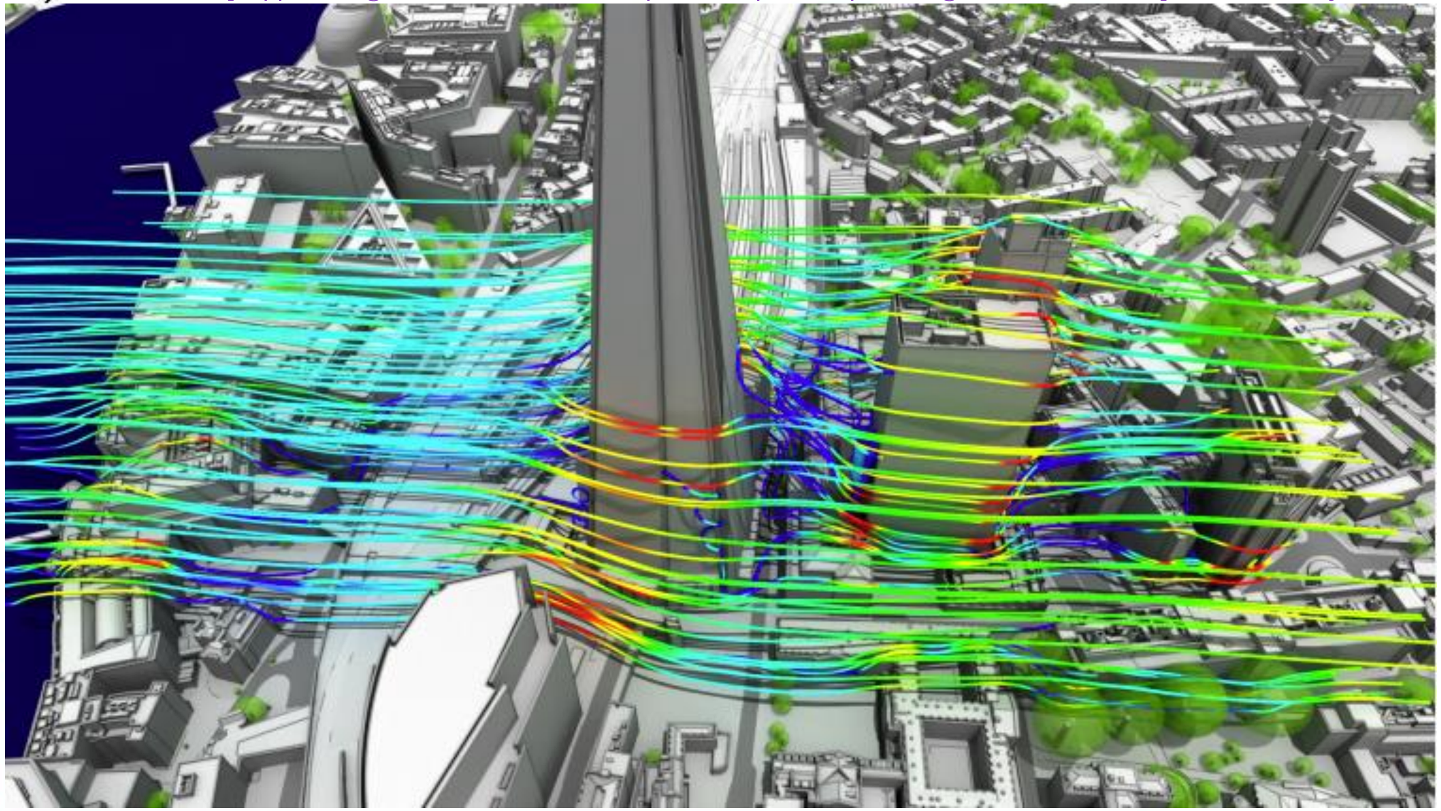
MODELS

- "A digital twin is a virtual representation synchronized with physical things (data), people or processes"(Digital shadows)

https://www.gim-international.com/content/article/can-digital-twin-techniques-serve-city-needs



Scenario of a flood in the Frayol watershed,
Le Teil city, Ardèche, France
with Virtual Reality - P. Bellemain, Gipsa-lab 2024
*DITRISC project*

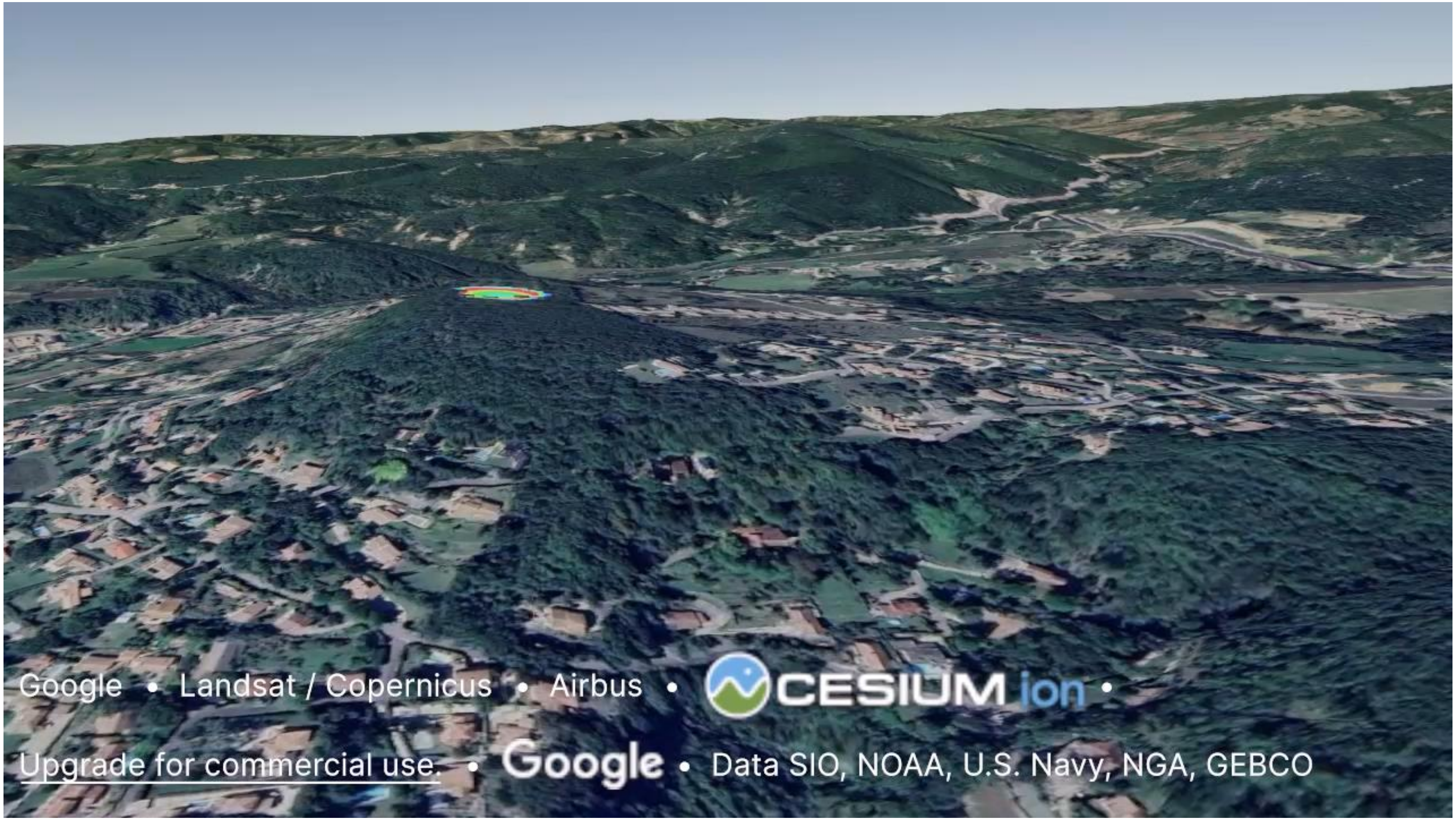


- For monitoring and diagnosis (early detection), prediction : scenarios elaboration, decision-making support

FRANCE 2030 PROGRAMME DE RECHERCHE RISQUES

# Digital Twin for environmental applications

*P. Bellemain, GIPSA-lab, 2025*



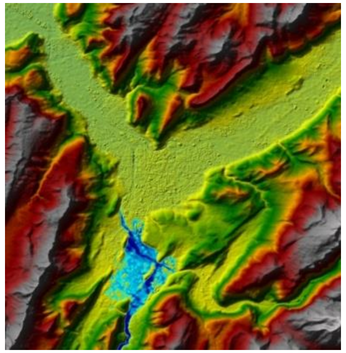

**Simulation of a wildfire near Le Teil City in Virtual Reality**

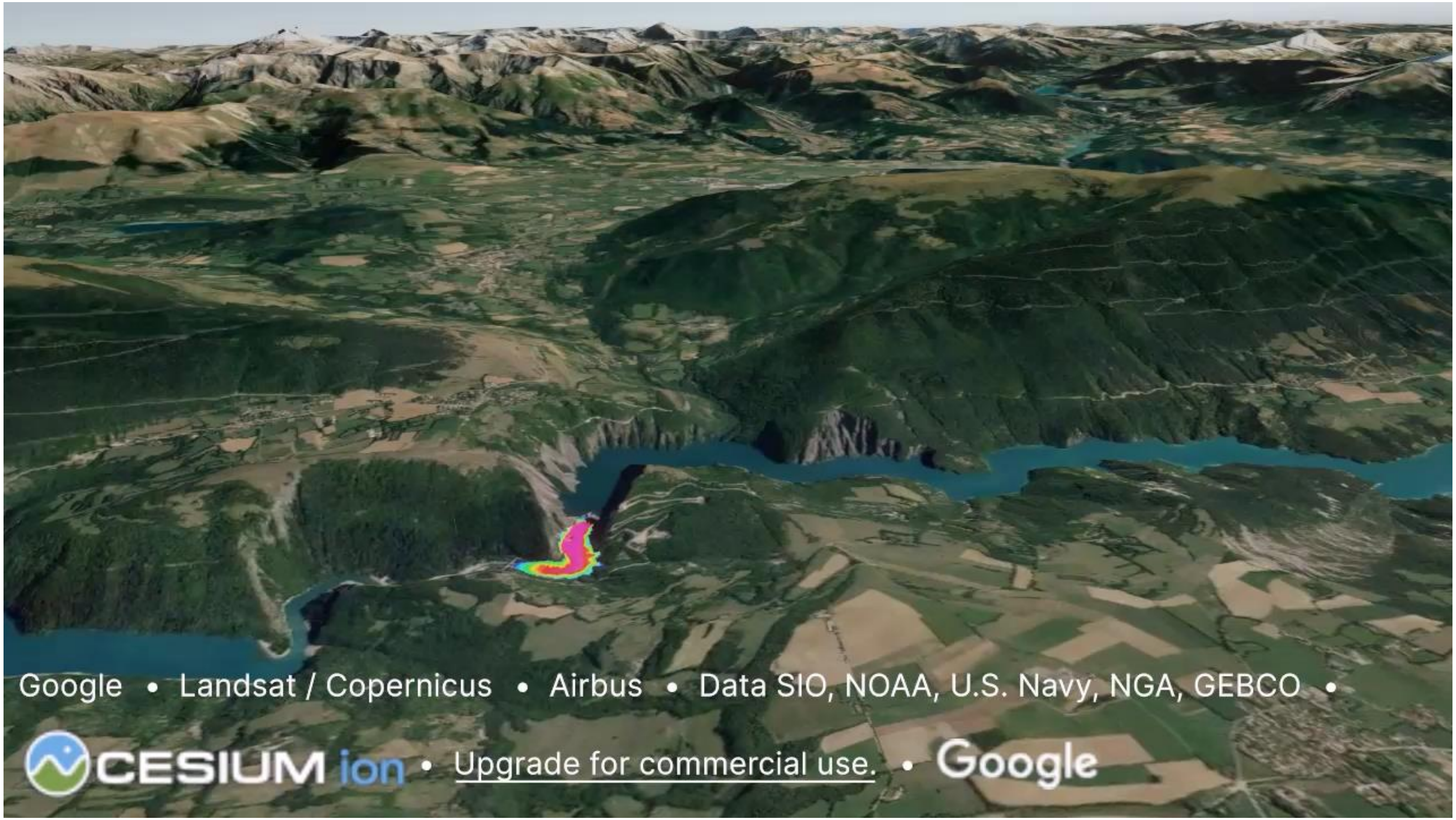
*DITRISC Project funded by MITI-CNRS*

# Digital Twin for environmental applications

P. Bellemain, GIPSA-lab, 2025

Monteynard dam



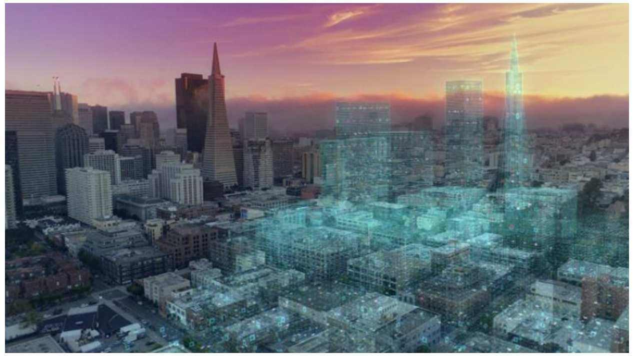



Simulation of Monteynard dam break

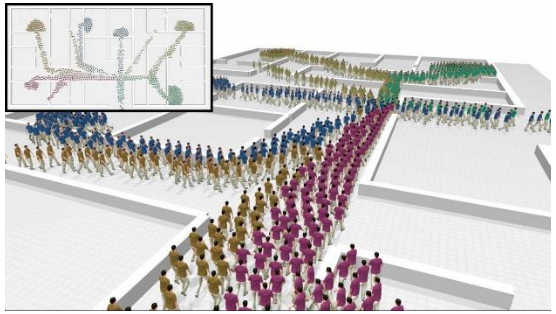
*DITRISC Project funded by MITI-CNRS*

# Scientific challenges of modelling and simulation for DT

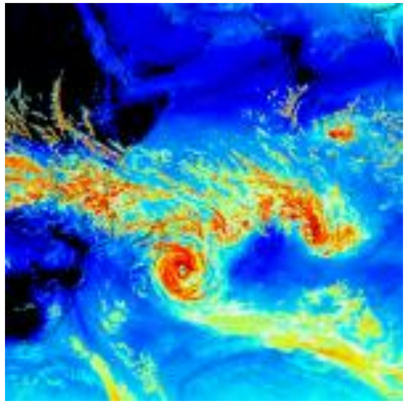
- ✓ Understanding cascading or coupled dynamics
- ✓ Managing different spatial and temporal scales
- ✓ Including human behavior

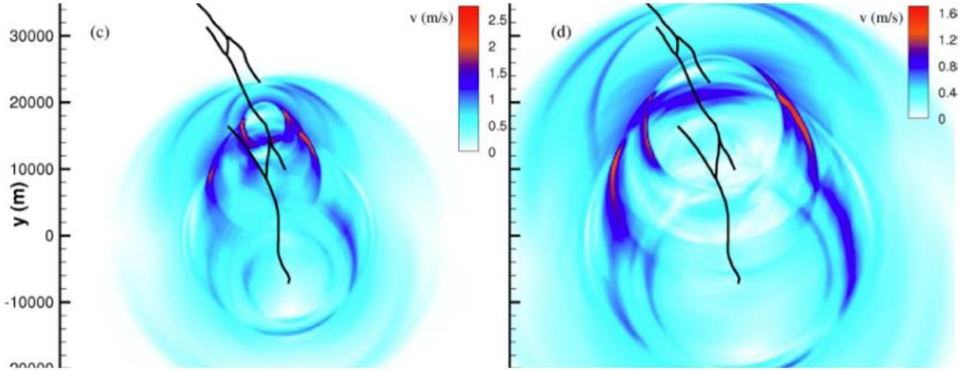


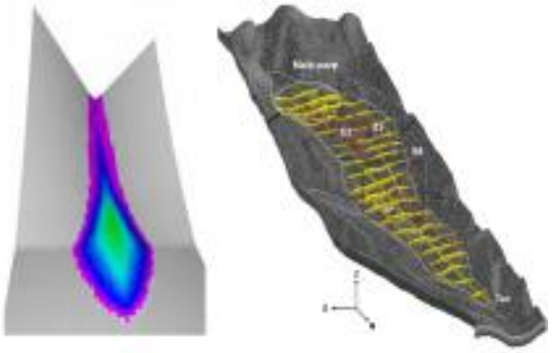
https://aecmag.com/news/digital-twins-for-a-sustainable-built-environment/



**Crowd evacuation simulation**





**Macroscale**
Cyclone simulation

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

**Mesoscale**
Earthquake simulation

**Microscale**
**Landslide simulation**

https://theconversation.com/modeliser-le-climat-grace-au-calcul-scientifique-70461

# **Model classification**: Knowledge vs data-based models

## Von Neumann paradigm

### Physical modelling

- ✓ Compliance with the laws of Physics
- ✓ Interpretability
- ✓ Parametrization

- ✓ Theoretical investment
- ✓ Calibration may be complex
- ✓ Complex phenomena that are sometimes difficult to take into account
- ✓ Simulation cost for large-scale models

Shallow Water Equations on a sphere

$$\frac{\partial \mathbf{u}_c}{\partial t} = -P \begin{bmatrix} (\mathbf{u}_c \cdot P\nabla_c)u_c + f(\mathbf{x} \times \mathbf{u}_c) \cdot \hat{\mathbf{i}} + g(P\hat{\mathbf{i}} \cdot \nabla_c)h \\ (\mathbf{u}_c \cdot P\nabla_c)v_c + f(\mathbf{x} \times \mathbf{u}_c) \cdot \hat{\mathbf{j}} + g(P\hat{\mathbf{j}} \cdot \nabla_c)h \\ (\mathbf{u}_c \cdot P\nabla_c)w_c + f(\mathbf{x} \times \mathbf{u}_c) \cdot \hat{\mathbf{k}} + g(P\hat{\mathbf{k}} \cdot \nabla_c)h \end{bmatrix} \qquad \frac{\partial h^*}{\partial t} = -(P\nabla_c) \cdot (h^* \mathbf{u}_c)$$

## AI – Machine Learning

### Data-based modelling

- ✓ Easier access to modeling
- ✓ Continuous learning and improvement
- ✓ Low simulation cost once learned

- ✓ Weak interpretability
- ✓ Compliance with the laws of Physics not guaranteed
- ✓ High sensitivity to data quality

$$y(t) = \Phi_{RN}(y(t-1), y(t,-2), \ldots, y(t-n), u(t-1, u(t-2), \ldots, u(t-m)))$$

Auto-regressive surrogate model

# **Modelling of natural hazards** – some examples

Floods, tsunami, landslides, avalanches dynamics

*Shallow water equations*
*(A. Barré de Saint Venant, 1871)*

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(Hu) + \frac{\partial}{\partial y}(Hv) = 0, \;\; H = h - b$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + g\frac{\partial h}{\partial x} = -\tau_x$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + g\frac{\partial h}{\partial x} = -\tau_y$$

$h$ = absolute height, $u$, $v$ = velocity in $x$, $y$ directions

Le glacier du Birch s'est effondré mercredi 28 mai 2025. - AP Photo / Maxar Technologies

*Google images*

# **Modelling of natural hazards** – some examples

## Advection-Diffusion-Reaction - Wildfire dynamics

$$\begin{aligned}
\frac{\partial T}{\partial t} + v_x \frac{\partial T}{\partial x} + v_y \frac{\partial T}{\partial y} &= k\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right) + Sr(T) + \lambda(T - T_a), \\
\frac{\partial S}{\partial t} &= -\beta S r(T), \\
r(T) &= e^{-\alpha/(T-T_a)}, T > T_a, r(T) = 0, T \leq T_a
\end{aligned}$$

*Arrhenius law of combustion*

$T(x, y, t)$ = distributed temperature in the ground layer, $T_a$ = ambient temperature, $S(x, y, t)$ = distributed mass fraction of fuel, $(v_x, v_y)$ = air velocity, $k$ = diffusion coefficient

*Google images*

# **Modelling of natural hazards** – some examples

## Elastic Wave equations - Seismic dynamics

$$\begin{aligned}\rho\partial^2_{t^2}u(t,x) &= \nabla.\sigma(t,x)+f(x)\\ \sigma(t,x) &= \lambda\nabla.u(t,x)I+\mu[\nabla u(t,x)+\nabla u(t,x)^T]\end{aligned}$$

where $u(t,x)$ = displacement vector field and $\sigma(t,x)$ = stress tensor ; $\rho(x)$ = density of the elastic medium, and $\lambda(x)$ and $\mu(x)$ are Lamé elastic constants ; $f(x)$ = seismic source distribution

*BRGM*

Séisme du Teil - 11 novembre 2019

FRANCE 2030 PROGRAMME DE RECHERCHE RISQUES

# Modelling of **crowd dynamics** in evacuation situations – Hughes' model

***PhD thesis – COCHAIR - PEPR IRiMa***

Density of individuals as a function of time t and space coordinates x

$$\partial_t \rho - \nabla \cdot (\rho\, f(\rho)^2\, \nabla \phi) = 0 \quad \text{in } Q_T, \qquad Q_T := (0,T) \times \Omega$$

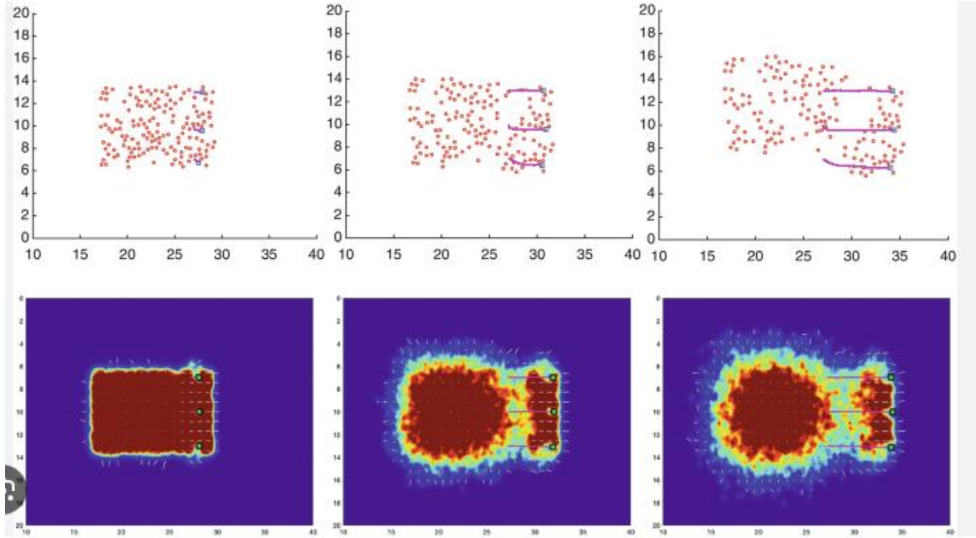
$$|\nabla \phi| = \frac{1}{f(\rho)} \quad \text{in } Q_T.$$

Pedestrians seek to minimize their (accurately) estimated travel time, but modify their velocity to avoid high densities. The potential is thus a solution of the Eikonal equation

$$v = f(\rho)\, u, \qquad |u| = 1,$$

where $f(\rho)$ is a non-increasing function that attains the value one at $\rho = 0$, is positive for $0 < \rho < 1$ and zero at $\rho = 1$.

$$u = -\frac{\nabla \phi}{|\nabla \phi|}.$$



*Multi-agent approach*

*Macroscopic approach*

https://doi.org/10.1007/978-3-030-50450-2_8

# General framework for spatio-temporal modelling: **PDEs**

$$\mathcal{L}(u(x,t),\theta) = g$$

$$\mathcal{B}(u(x,t),\theta)) = f$$

$u(x,t)$ is a physical quantity, function of space $x$ et time $t$.

$\theta$ is a vector of parameters

$\mathcal{L}$ and $\mathcal{B}$ are the differential and boundary/initial condition operators, respectively.

$\mathcal{L}$ is a function of partial derivatives $\frac{\partial}{\partial t}, \frac{\partial}{\partial x}, \ldots, \frac{\partial^k}{\partial x^k}$.

# DT: Increasing synergy between models and data

For better reality representation and forecasting capabilities

## Findings

Complex multi-scale spatio-temporal Phenomena

Unstationarity & uncertainty

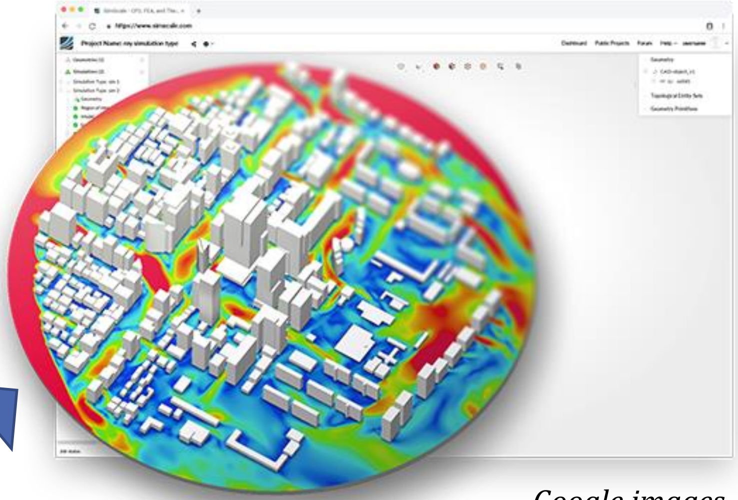
Massive and heterogeneous data, weak signals

## Scientific challenges

Improving the relevance of models for more realistic scenario building

Quickly extract relevant information from data: Detect, predict to alert, anticipate

# Synergy between models and data



*Google images*

***Direct modeling***

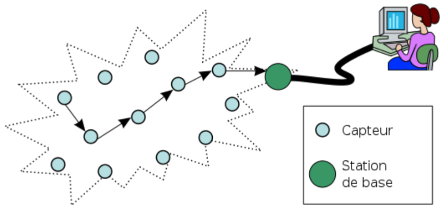
$$\mathbf{x}_k = \mathcal{M}_{k:k-1}(\mathbf{x}_{k-1}, \boldsymbol{\lambda}) + \eta_k$$
$$\mathbf{y}_k = \mathcal{H}_k(\mathbf{x}_k) + \epsilon_k$$

Parameters
Knowledge/Representation
Models

Synthetic Data

On-Field Data

**Sensors**




- GPS
- Camera
- Accelerometers
- Microphone

***Inverse modelling*** , **Data Assimilation**, **Inverse Problems**

*Google images*

# What is **data assimilation**?

**Estimate states and/or parameters of a physical system using sensor measurements and a model of the system dynamics**

Sensor measurements

An optimization problem

Minimize

$$\mathcal{J}(\mathbf{x}_{K:0}) = \frac{1}{2}\sum_{k=0}^{K}\|\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\|^2_{\mathbf{R}_k^{-1}} + \frac{1}{2}\sum_{k=1}^{K}\|\mathbf{x}_k - \mathcal{M}_{k:k-1}(\mathbf{x}_{k-1})\|^2_{\mathbf{Q}_k^{-1}} + \frac{1}{2}\|\mathbf{x}_0 - \mathbf{x}^{\mathrm{b}}\|^2_{\mathbf{B}^{-1}}$$

State at time K

$$\mathbf{x}_k = \mathcal{M}_{k:k-1}(\mathbf{x}_{k-1}, \boldsymbol{\lambda}) + \boldsymbol{\eta}_k$$

Dynamical model

$$\mathbf{y}_k = \mathcal{H}_k(\mathbf{x}_k) + \boldsymbol{\epsilon}_k$$

Sensor model

Regularization term

✓ Accurate method (with adjoint system)

✓ Significant computation time

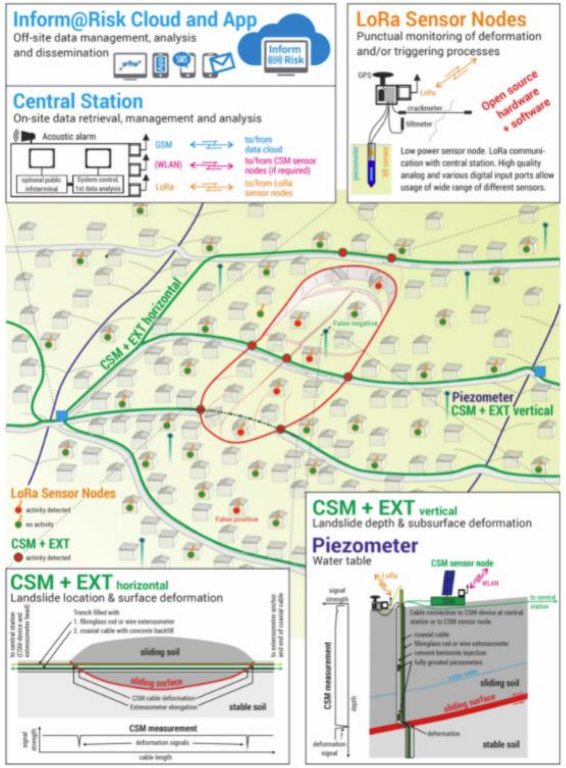
Carrassi et al, 2018, data assimilation in geosciences
https://arxiv.org/abs/1709.02798

# What is **data for DT and natural hazard monitoring?**

- ✓ Geophysical sensors deployment using Wireless Sensor Networks by satellites
- ✓ New sources of Data: Citizen data

**Crowdsensing**

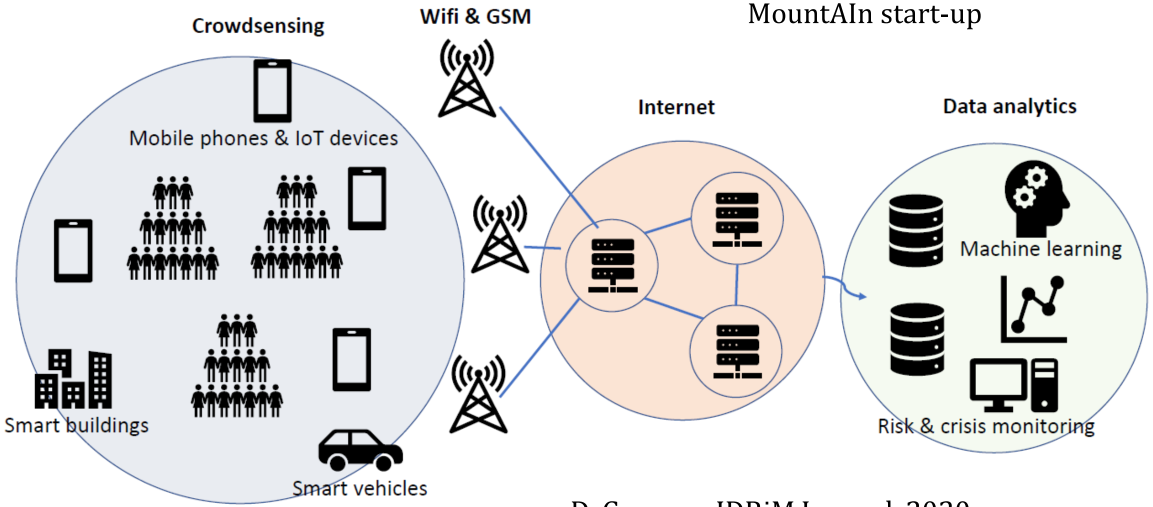

https://www.latimes.com/




https://doi.org/10.3390/s21082609

***Internet of Things (IoT)***


MountAIn start-up

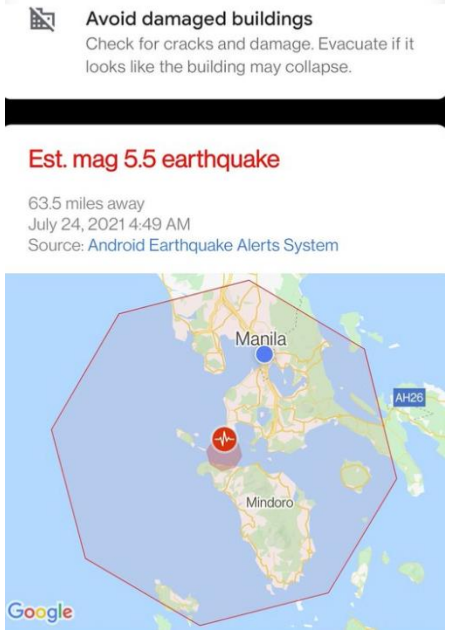



Sensors : GPS, camera, accelerometer, temperature

D. Georges, IDRiM Journal, 2020
https://doi.org/10.5595/001c.17963

Android's earthquake alert system


https://www.youtube.com/watch?v=1_9PeG6Icpg



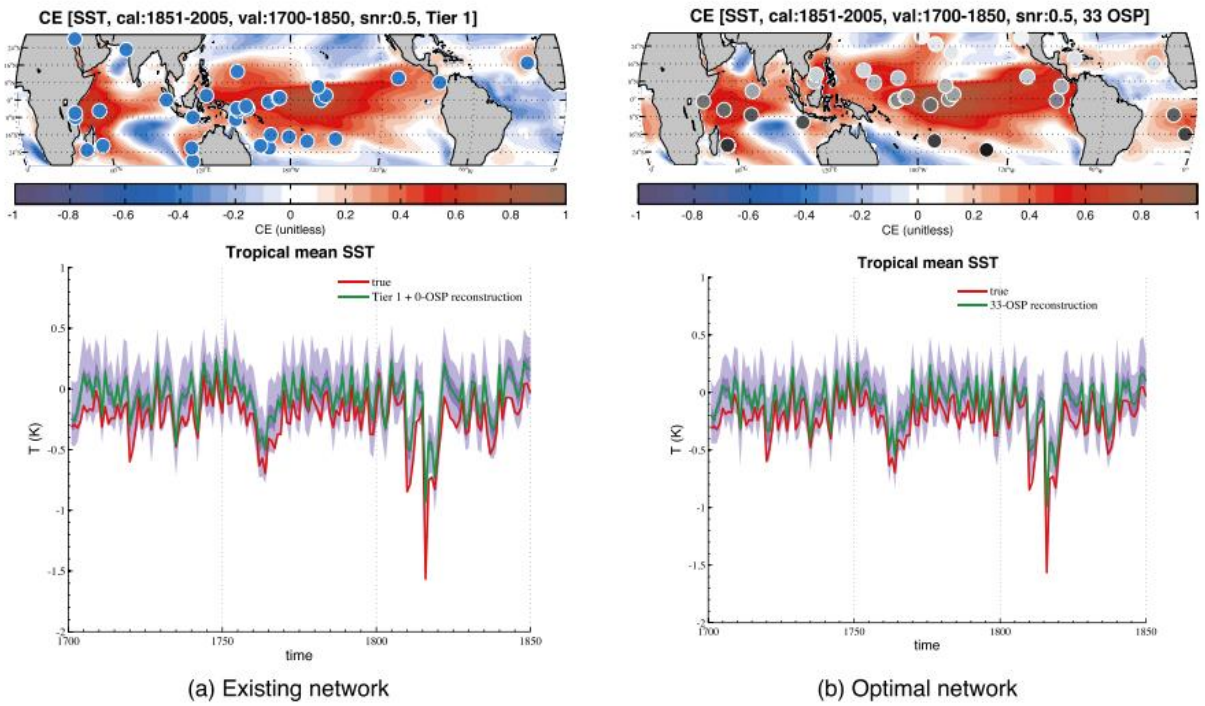

https://twitter.com

# A challenge: **The optimal deployment of sensors**

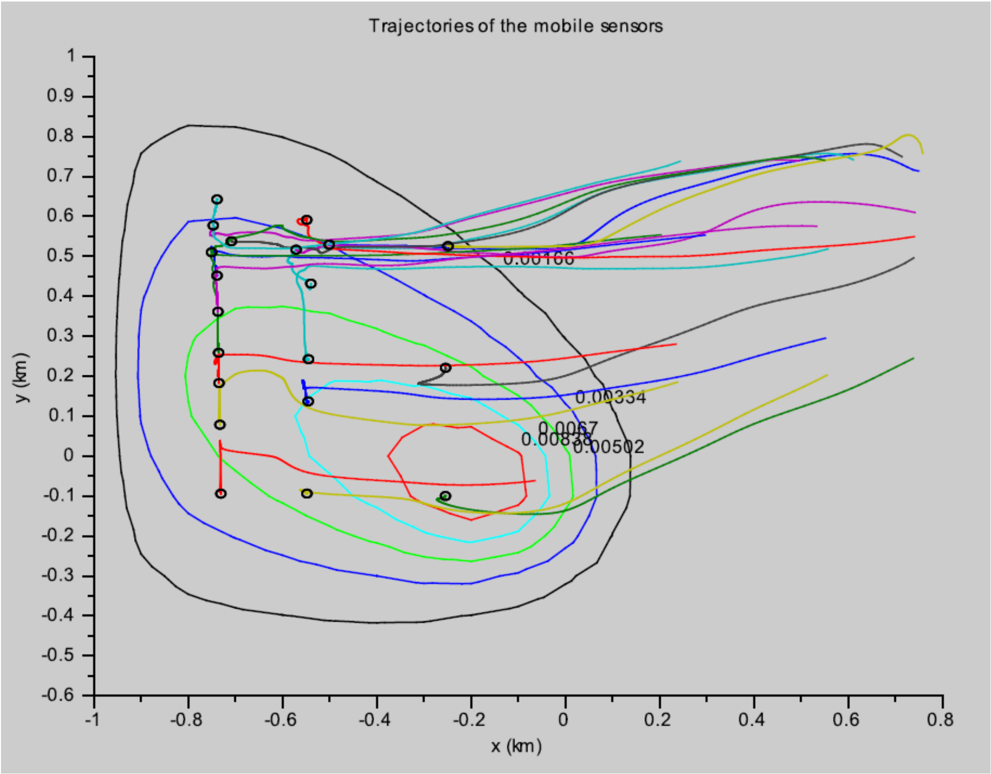
**Optimal sensor location**

Maximizing an observability indicator: e.g., sensitivity of measurements to parameters/states to be estimated

Optimal network of coral proxies for paleoclimate monitoring




*M. Comboul et al, J. of Climate, 2015*

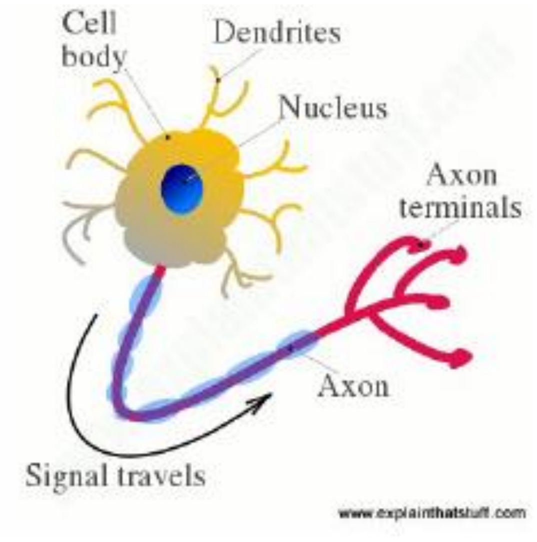


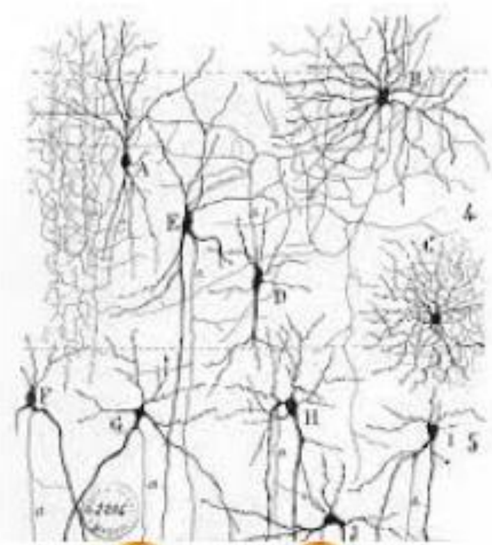

Optimal navigation of drones for pollution detection

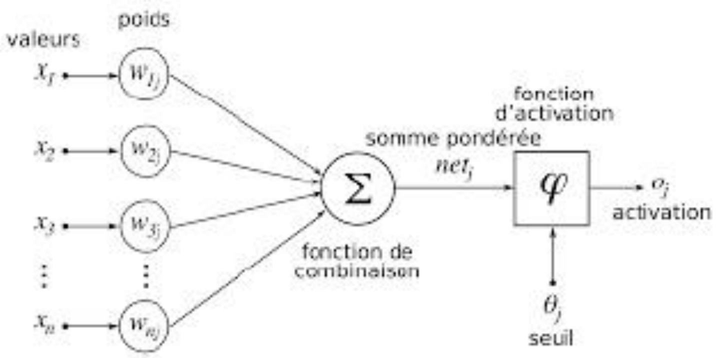
*D. Georges, IDRiM Journal, 2020*

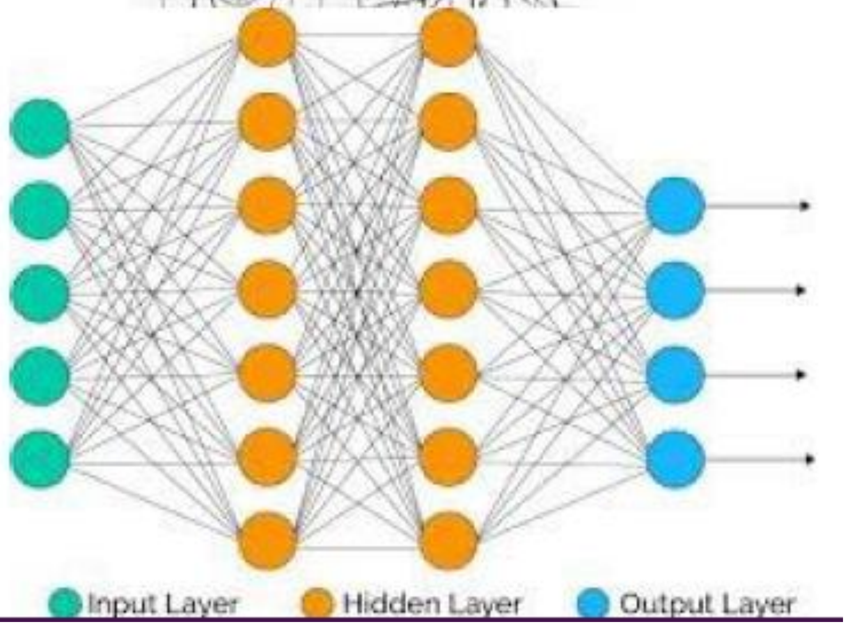
# Deep Learning: What is an **artificial neural network** (ANN)?

An artificial neuron = a mathematical approximation of a biological neuron





# Deep Learning: What is **supervised learning** of a ANN?

**Finding a relationship** between a set of data X and a set of data Y,

by solving a **nonlinear regression problem**:

Minimization of a loss function

$$\min_{\theta} \frac{1}{2} \sum_{i=1}^{L} \|\Phi_\theta(X_i) - Y_i\|^2$$

**Update of the parameters of the neural network**,
by using a stochastic gradient approach:

$$\theta^{k+1} = \theta^k - \nu \frac{\partial \Phi_\theta}{\partial \theta}^T (\Phi_\theta(X_k) - Y_k)$$

***Back propagation computation (Le Cun et al., 1989)***

# An example in natural hazard monitoring : Data-based wildfire ignition location

Direct learning of the inverse function based on snapshots of sensor outputs

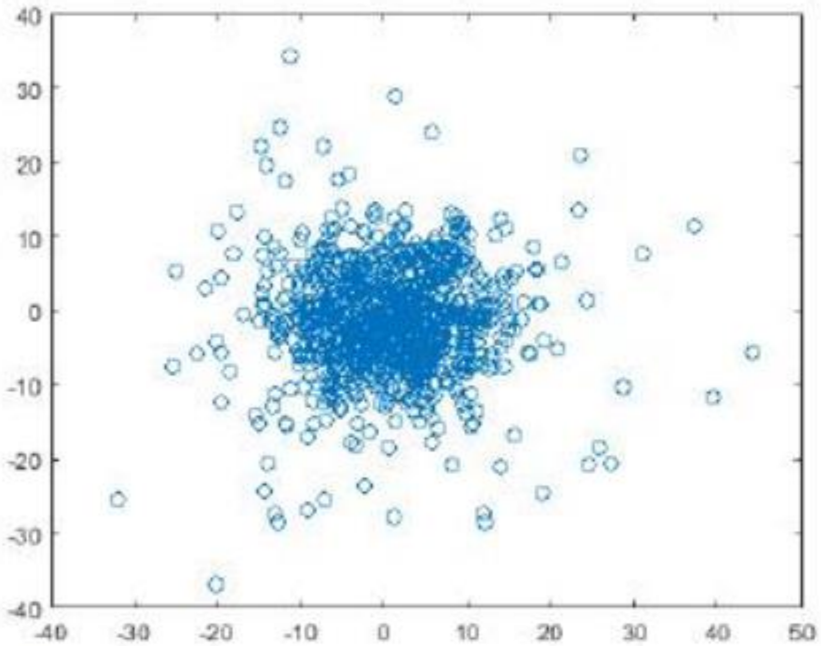
*L. Barbance, S. Benkaddour, S. Drissi, D. Georges, 2019*

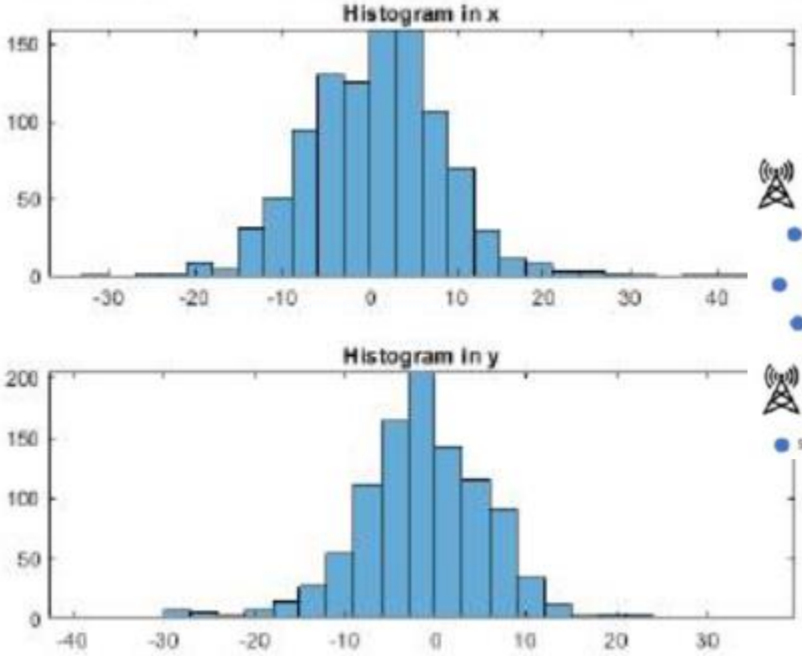
Google images

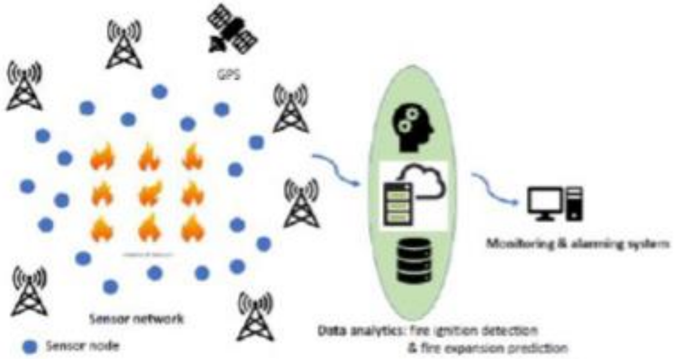
y=temperature measurements

Inverse function $p = \mathcal{F}(y)$ → Convolutional Neural Network → p = fire start location

Distribution of location errors




GRENOBLE INP Ense3 UGA

85 % of location errors < 10 m
With 30 sensors scattered on a cell [500m, 500m]

Learning with synthetic data obtained from model simulations

# From **data-based** machine learning to **physics-informed** machine learning

- ***Bringing together the "best of the two worlds" of modelling:***

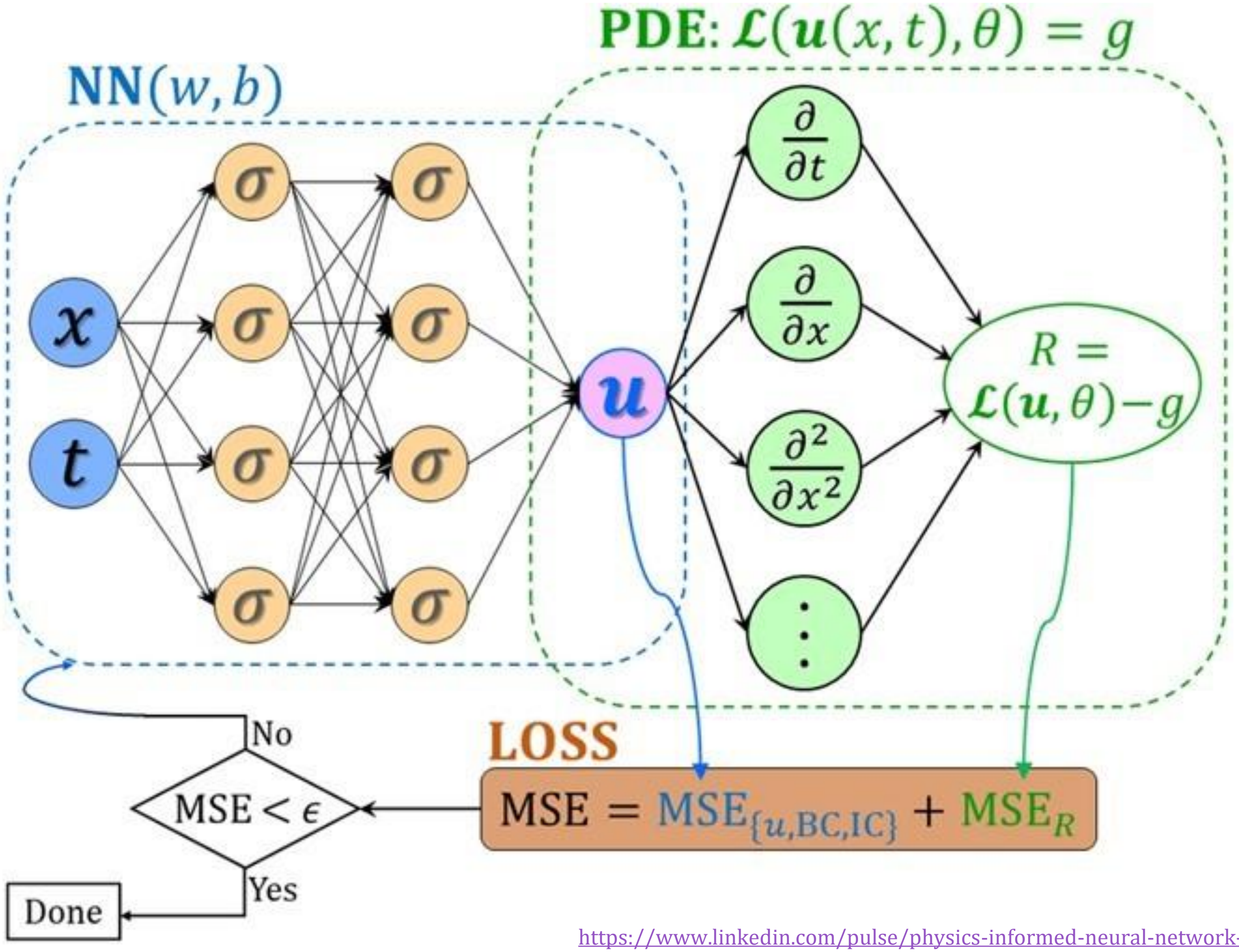
- The data is used to "enrich" the solution derived from a deep neural network

- But **the solution complies with the laws of physics**

# What is a **Physics-Informed Neural Network** (PINN)?

(*Raissi et al., 2019*)

The physical model is now included in the loss function




https://www.linkedin.com/pulse/physics-informed-neural-network-pinn-une-r%C3%A9volution-la-mitchozounou-0eaje/

# Mathematical formulation of PINNs

$$\min_{(w,b)} MSE_{\{u,BC,IC\}} + MSE_R$$

$$MSE_R = \frac{K_m}{N} \sum_{k=1}^{N} (\mathcal{L}(u_{NN}(x_k, t_k)), \theta) - g)^2$$

If alone, a pure data-based NN learning

$$MSE_{\{u,BC,IC\}} = \frac{K_d}{M} \sum_{k=1}^{M} (u_{NN}(x_k, t_k) - y(x_k, t_k))^2 + \frac{K_b}{N} \sum_{k=1}^{N} (\mathcal{B}(u_{NN}(x_k, t_k)), \theta) - f)^2$$

$y(x_k, t_k)$ is an observable (solution of the PDE obtained from real data and/or simulations).

$u_{NN}$ is the approximation produced by the neural network.

$\boxed{\text{Data assimilation}}$ (estimation of the parameters $\theta$ and $u(x,t)$ without knowing the initial state $u(x,0)$) can be solved by considering

$$\min_{(w,b),\theta} MSE_{\{u,BC\}} + MSE_R$$

# **A direct link** with the classical **data assimilation approach**

Similar to the first term of MSE_{u,BC}

Similar to MSE_R

Minimize

$$\mathcal{J}(\mathbf{x}_{K:0}) = \frac{1}{2}\sum_{k=0}^{K} \|\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\|^2_{\mathbf{R}_k^{-1}} + \frac{1}{2}\sum_{k=1}^{K} \|\mathbf{x}_k - \mathcal{M}_{k:k-1}(\mathbf{x}_{k-1})\|^2_{\mathbf{Q}_k^{-1}} + \frac{1}{2}\|\mathbf{x}_0 - \mathbf{x}^{\mathrm{b}}\|^2_{\mathbf{B}^{-1}}$$

State at time K

$$\mathbf{x}_k = \mathcal{M}_{k:k-1}(\mathbf{x}_{k-1}, \boldsymbol{\lambda}) + \boldsymbol{\eta}_k$$

Dynamical model

$$\mathbf{y}_k = \mathcal{H}_k(\mathbf{x}_k) + \boldsymbol{\epsilon}_k$$

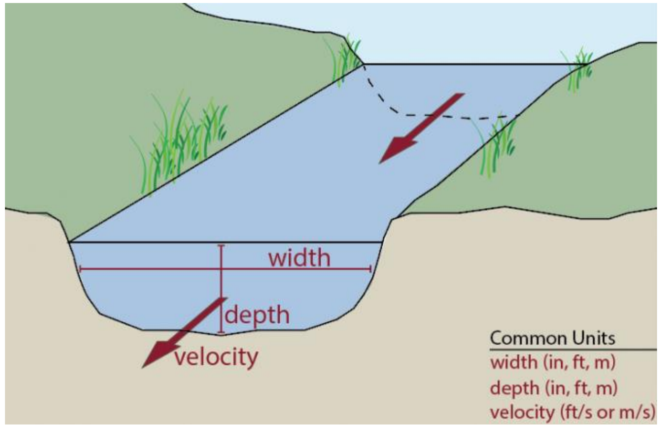
Sensor model

# **A case study – DT for flood monitoring** based on a 1D runoff model

**A simplification of shallow water equations:**

***The kinematic wave equation***


Guadalupe flash flood – Texas, July 2025 – abc news




*Google images*

Physical model

$$\alpha\beta Q(x,t)^{\beta-1}\frac{\partial Q}{\partial t}(x,t)+\frac{\partial Q}{\partial x}(x,t)=0, x\in[0,L], t\in(0,T),$$
$$Q(x=0,t)=f(t)\ (BC),$$
$$Q(x=0,t)=Q_0(x)\ (IC)$$

$Q(x,t)$ is the water flow rate, as a function of space coordinate $x$ and time $t$.

$\beta = 0.6$ and $\alpha = \left(\frac{nb^{2/3}}{\sqrt{S_0}}\right)^{3/5}$

$n$ is a roughness coefficient, $b$ is the width of the channel, $S_0$ is th slope of the channel,

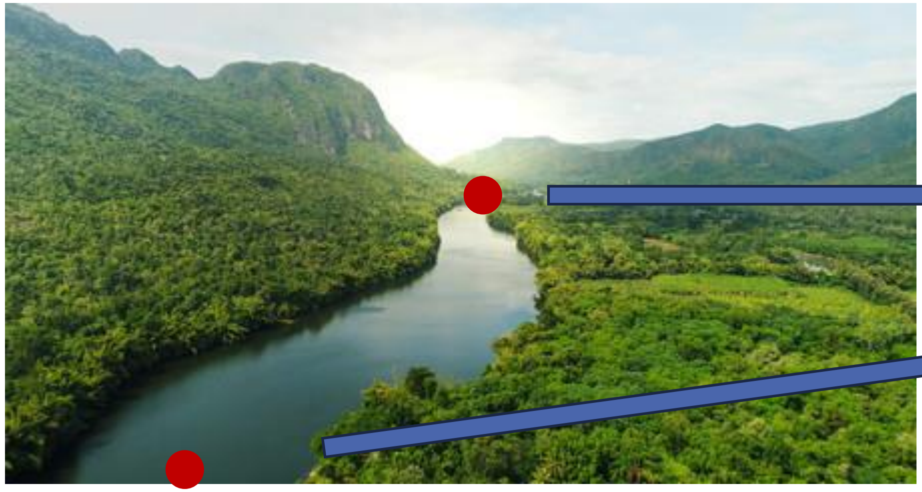
$f(t)$ is the upstream flow rate defining the boundary condition, and $Q_0(x)$ is the initial flow rate distribution.

Data : Flow rate

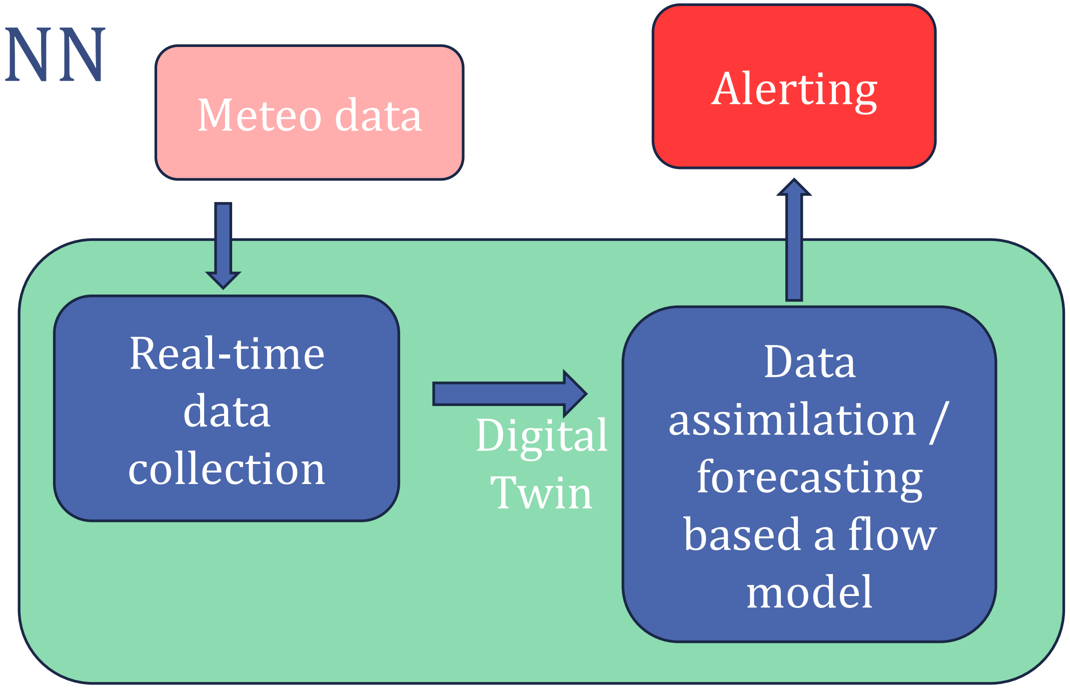
$$y_i^Q(t) = Q(x = x_i^s, t)$$

$x_i^s$ is the location of the sensor $i$

# DT: **Flood monitoring** using a PINN

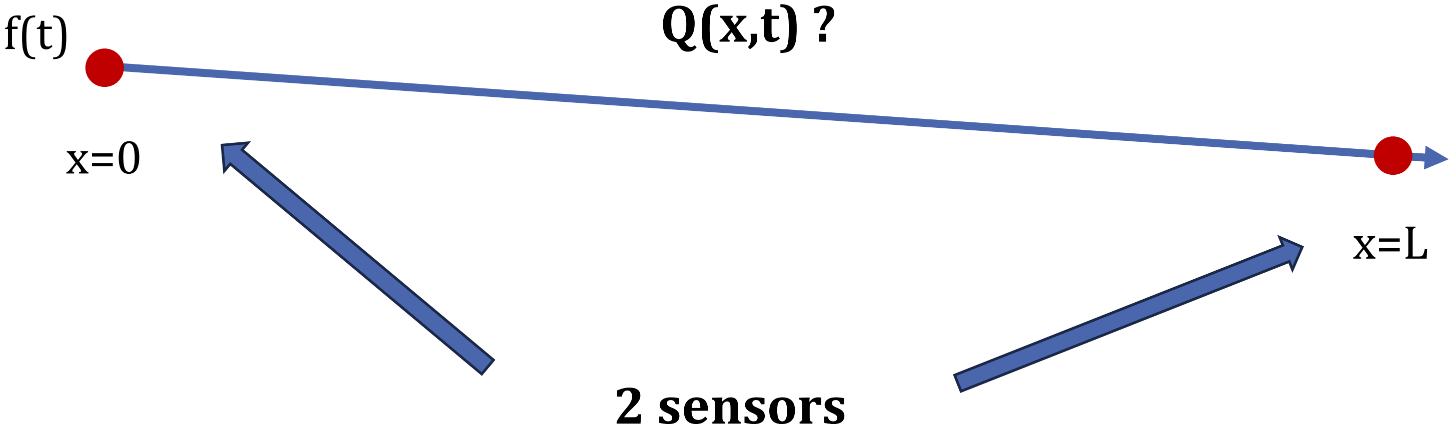


Google images



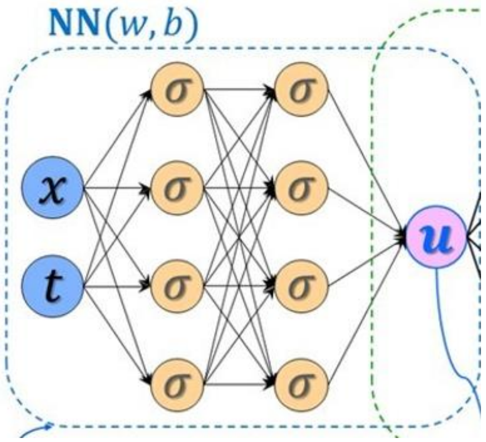

***Receding horizon approach***

**From only 2 sensors at the extremities,**
**can we estimate the flow rate Q(x,t) for all x and t?**



# PINN formulation for flood monitoring

https://www.linkedin.com/pulse/physics-informed-neural-network-pinn-une-r%C3%A9volution-la-mitchozounou-0eaje/




$$\min_{(w,b)} MSE_{\{u,BC\}} + K_r \times MSE_R$$

$$MSE_R = \frac{1}{N}\sum_{k=1}^{N}\underbrace{(\alpha\beta Q_{NN}(x_k,t_k)^{\beta-1}\frac{\partial Q_{NN}}{\partial t}(x_k,t_k) + \frac{\partial Q_{NN}}{\partial x}(x_k,t_k))^2}_{\text{PDE residual}}$$

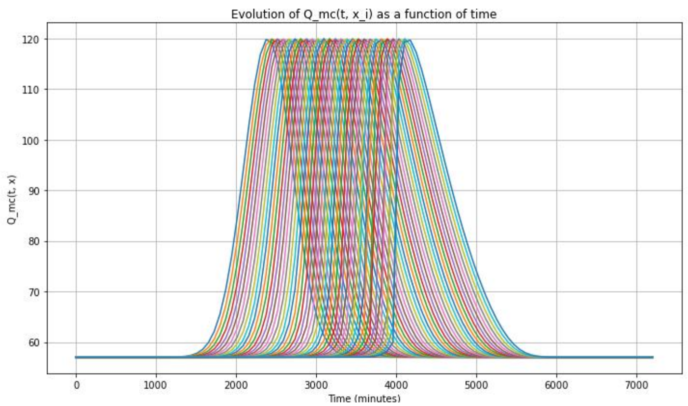
$$MSE_{\{u,BC\}} = \frac{1}{M}\sum_{i=1}^{N_s}\sum_{k=1}^{M}\underbrace{(Q_{NN}(x_i^s,t_k) - y_i^Q(t_k))^2}_{\text{Data sensor fitting term}} + \frac{1}{M}\sum_{k=1}^{M}\underbrace{(u_{NN}(0,t_k) - f(t_k))^2}_{\text{BC residual}}$$

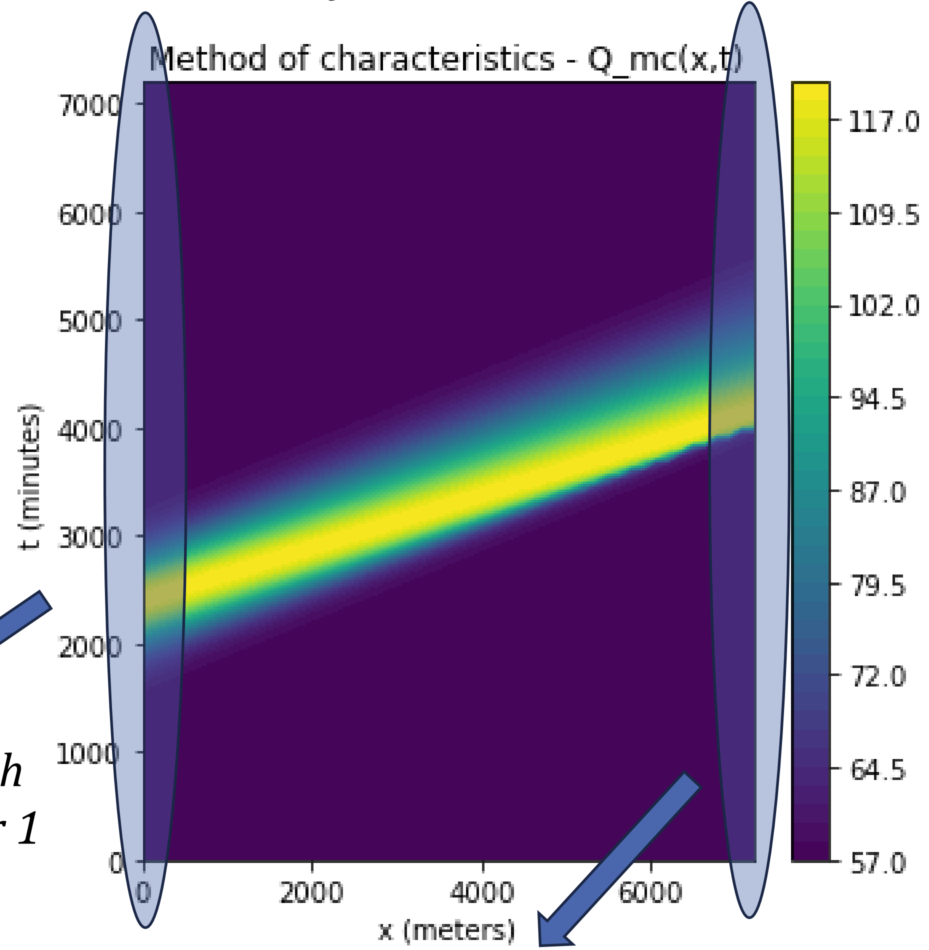
**NN: 1 hidden layer with 100 neurons**

# Results

## Channel of 7200m with width of 60m,
## Gaussian distribution of the inflow hydrograph

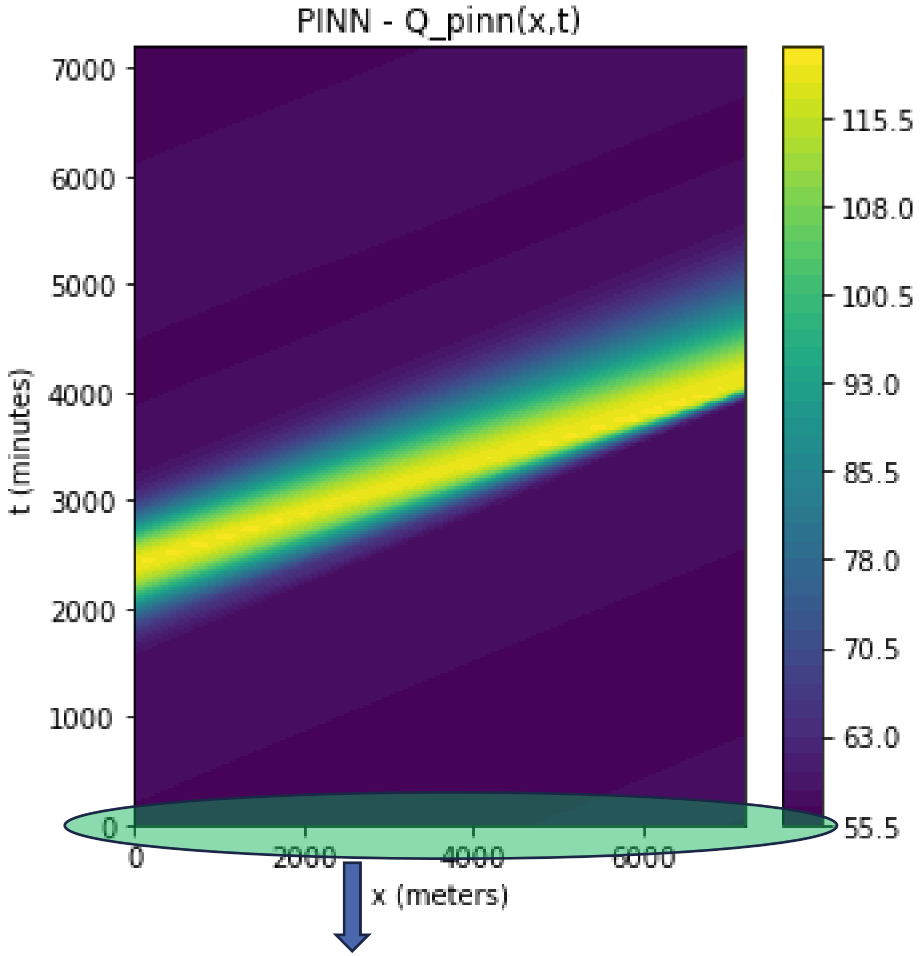
- The « real » data is obtained from a simulation based on the *method of characteristics*




*Inflow hydrograph upstream– sensor 1*

*Downstream flow - sensor 2*




**Good estimation of the unknown initial state!**

# Conclusions

- **Machine learning is set to play an important role in the development of digital twins:**

  **The synergy between models and data is facilitated by PINNs:**
  *Multi resolution, no need for adjoint system calculation, meshless approach, complexity independent of dimensions…*

- **Many challenges remain**, amongst them:

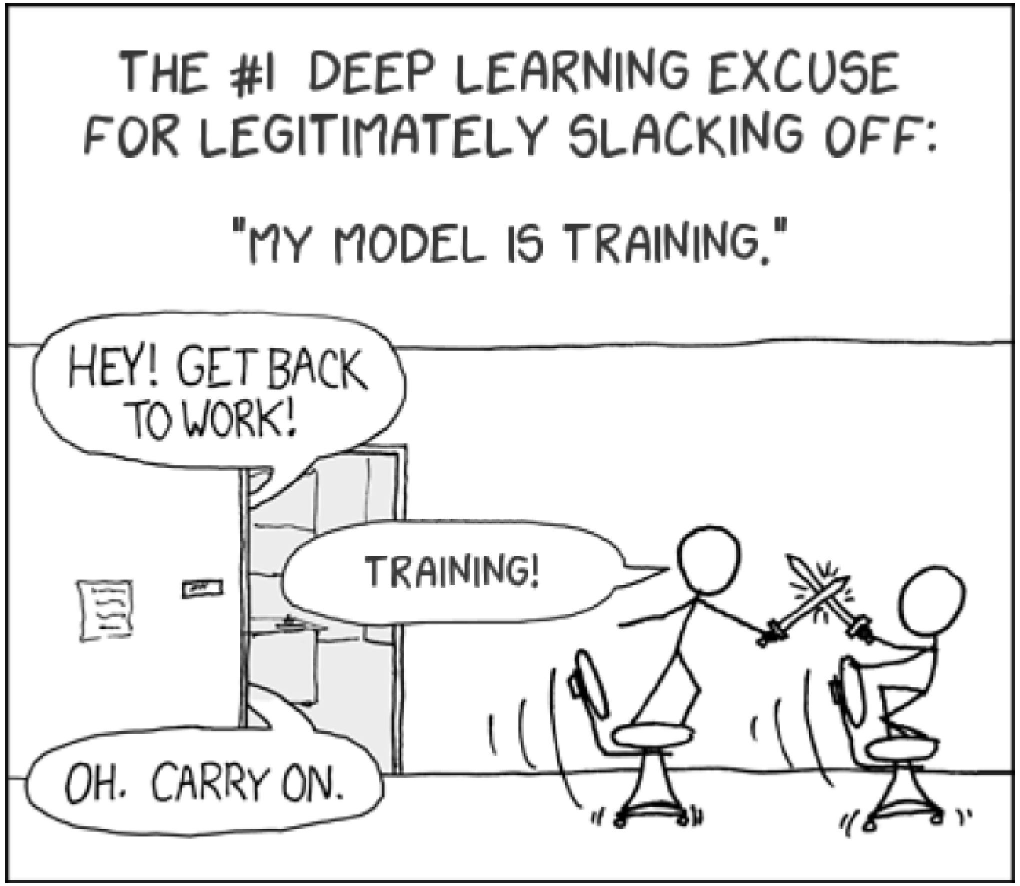
  **Efficient learning  of parametrized solutions for complex physical models:**

  *high-dimensional simulation/estimation problem for a variety of data and parameters,*

  *very heterogeneous domains (many internal BCs),*

  *improvement through optimal sensor location …*

# *Thank you for your attention!*




Source iFunny – Google images