# *Agenda*

**Part 1 (theory, tools):**

1. Coupling microkinetic modeling and CFD
2. Speed-up techniques and coupling kMC and CFD
3. Hierarchical modeling of chemical reactors

**Break & switch link (around 3.30 PM GMT + 2)**

**Part 2 (applications):**

1. catalyticFoam overview (download, installation, use)
2. Set-up of a case-study
3. Use of Machine Learning for tabulation of kinetic schemes (kMC, mean-field)
4. Case-study on coupling kMC and CFD

# Coupling microkinetic modeling and kMC simulations with CFD in heterogeneous catalysis – part 1

Mauro Bracconi & Matteo Maestri

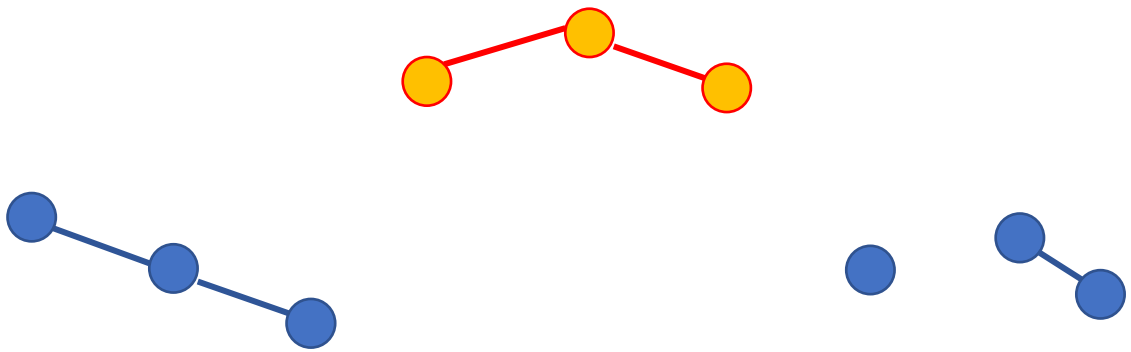
Laboratory of Catalysis and Catalytic Processes LCCP

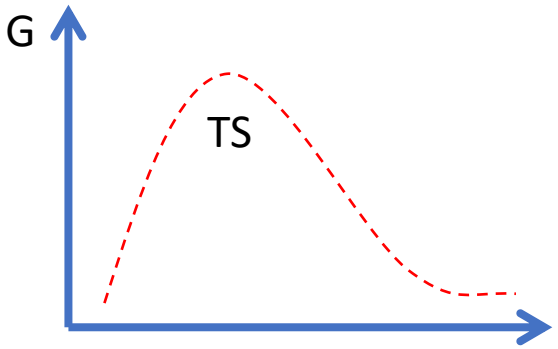
POLITECNICO MILANO 1863

Online workshop on «Theory, applications and tools for multiscale kinetic modeling»
Day 3 – July 15, 2020

***Chemical reactions* reflect the universal tendency of systems to approach equilibrium.**

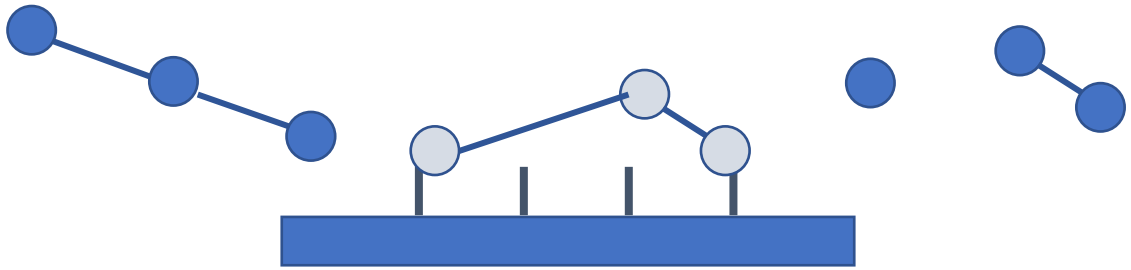
# *The dynamics towards equilibrium are reflected in rates of chemical reactions.*



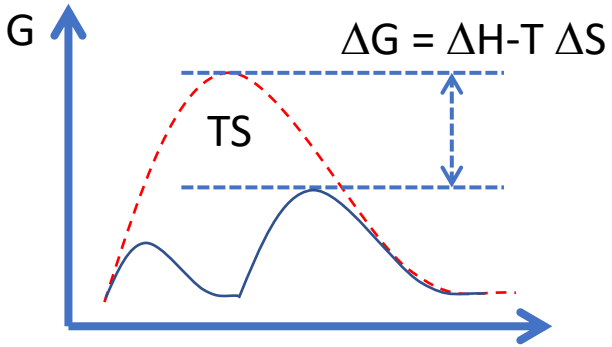



$$r \propto \exp\left(-\frac{G^{TS} - G_R}{RT}\right)$$

# *Catalysis: the role of the active sites…*

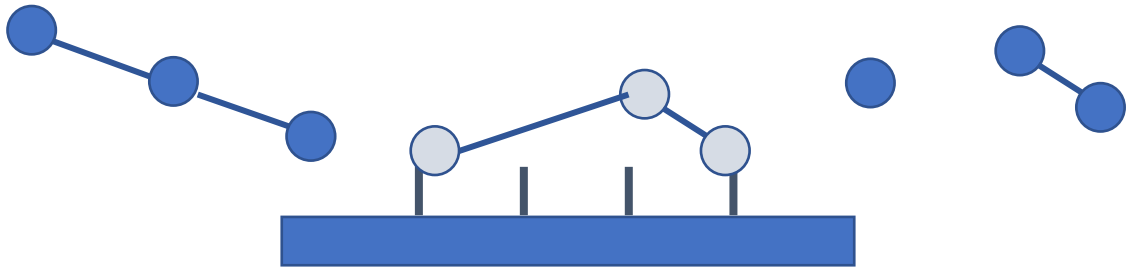


Catalyst: functional material with specific “active sites”, which provide a stabilization of the free energy of the TS

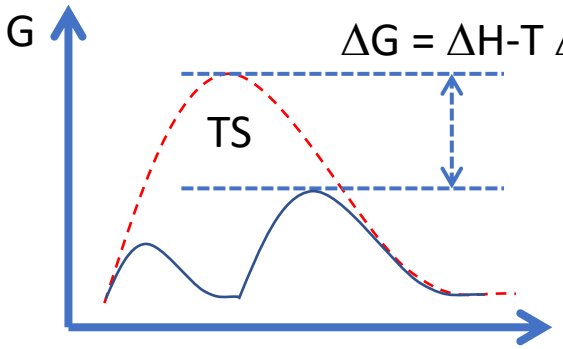



$$r \propto \exp\left(-\frac{G^{TS} - G_R}{RT}\right)$$

# *Catalysis: the role of the active sites…*



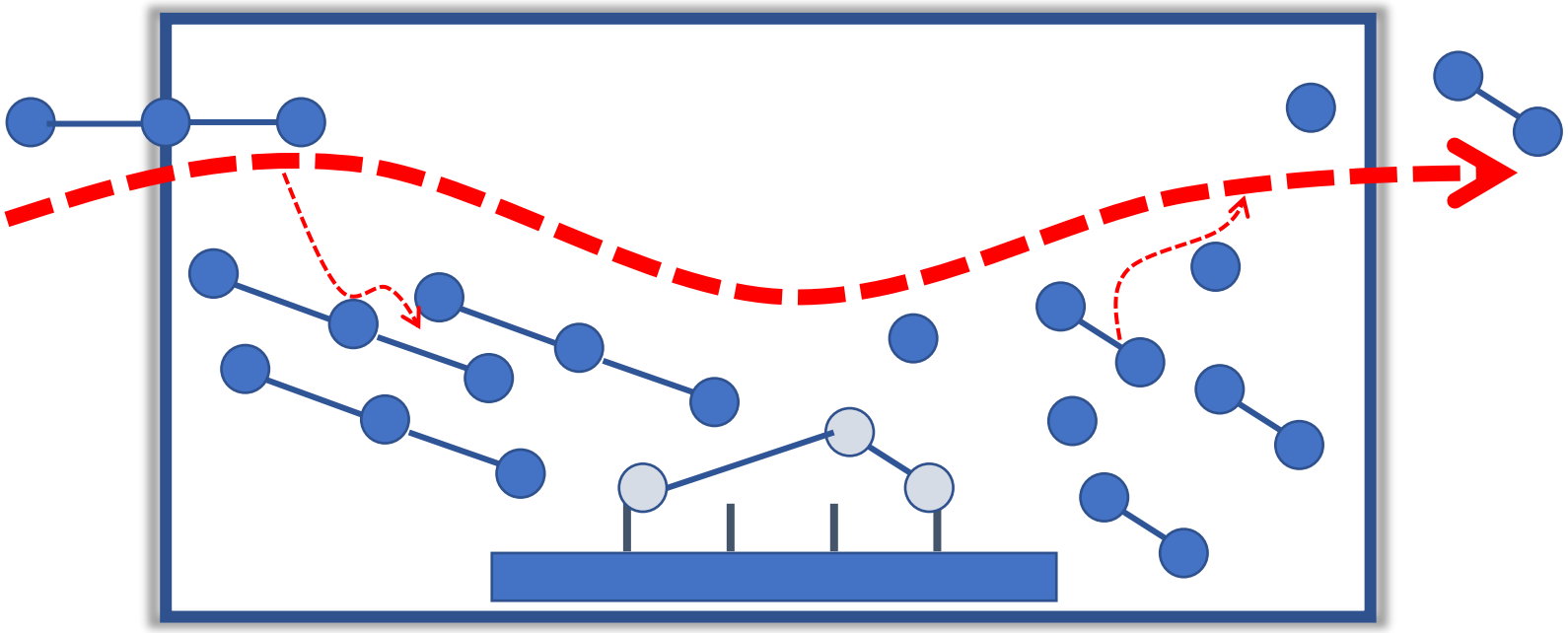
Catalyst: functional material with specific “active sites”, which provide a stabilization of the free energy of the TS



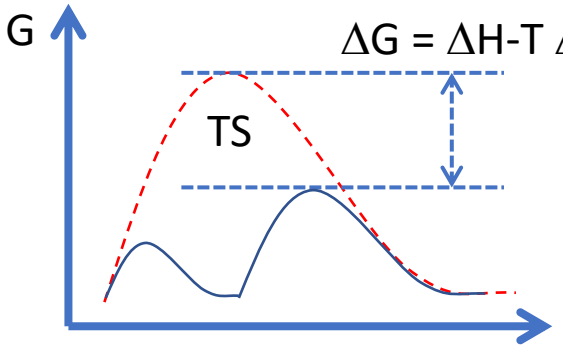

$$r \propto \exp\left(-\frac{G^{TS} - G_R}{RT}\right)$$

G depends not only on H and S, but also on the local chemical potential

# *Catalysis: the role of the active sites…*
# *…and their environment.*



Catalyst: functional material with specific “active sites”, which provide a stabilization of the free energy of the TS



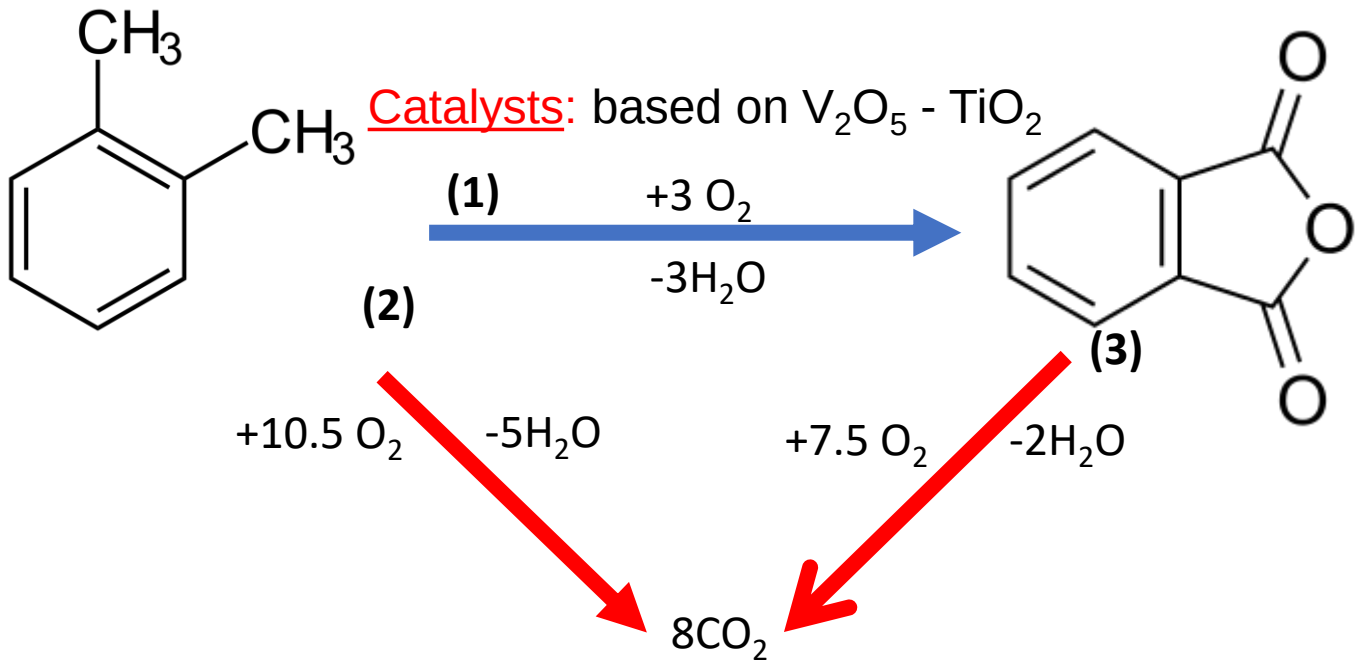

$$\Delta G = \Delta H - T\,\Delta S$$

$$r \propto \exp\left(-\frac{G^{TS} - G_R}{RT}\right)$$

G depends not only on H and S, but also on the local chemical potential

# *The importance of the reactor: tailoring the environment.*

**REACTION AND CATALYST**



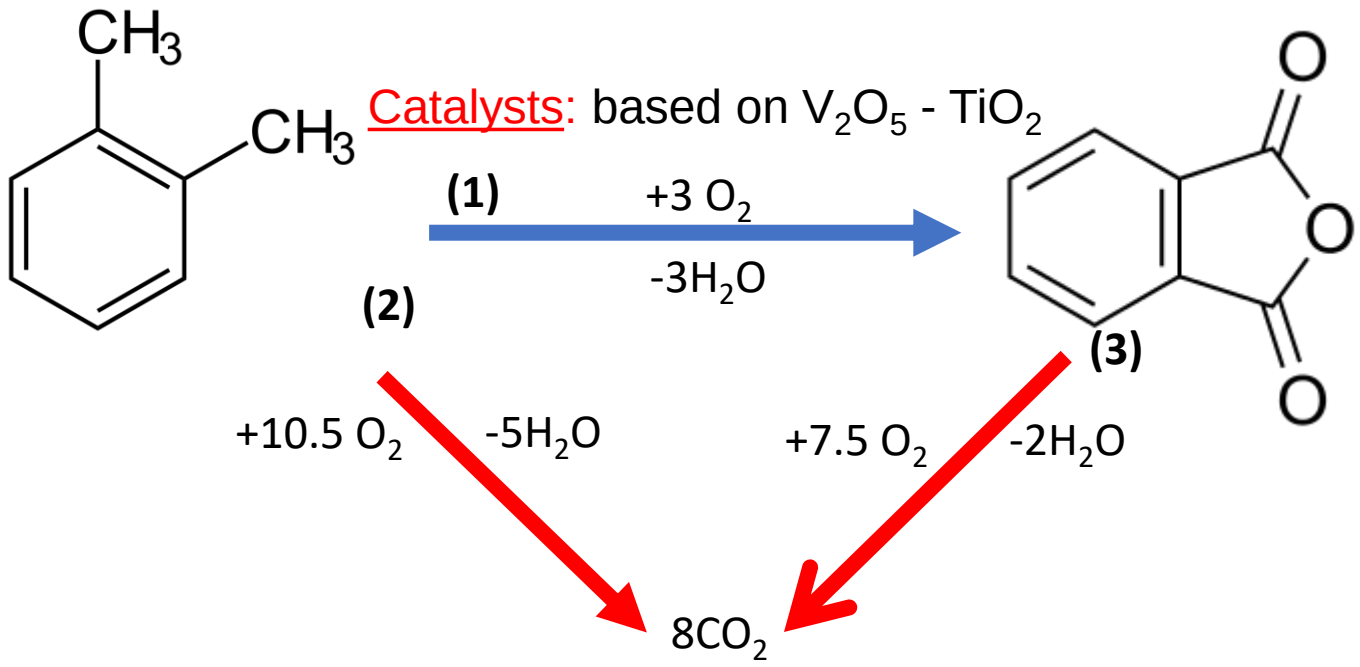

(1) **$C_6H_4(CH_3)_2 + 3O_2 \rightarrow C_6H_4(CO)_2O + 3H_2O$ $\Delta H$=-1285409 kJ/kmol**

(2) $C_6H_4(CH_3)_2 + 10.5O_2 \rightarrow 8CO_2 + 5H_2O$ $\Delta H$=-4564000 kJ/kmol

(3) $C_6H_4(CO)_2O + +7.5\ O_2 \rightarrow 8CO_2 + 2H_2O$ $\Delta H$=-3278591 kJ/kmol

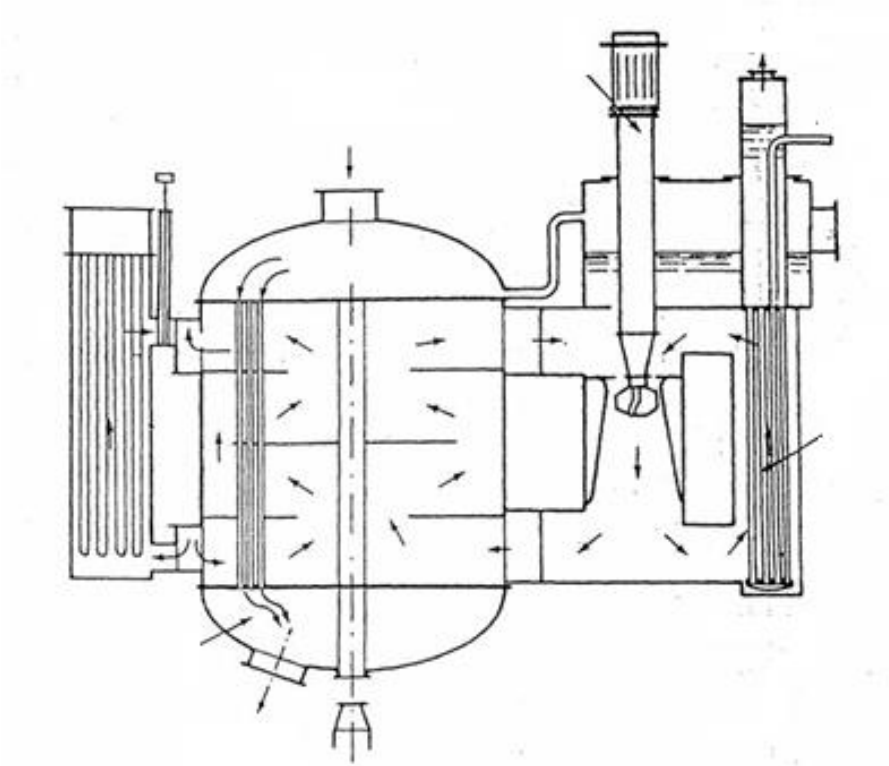
# *The importance of the reactor: tailoring the environment.*

**REACTION AND CATALYST**




**REACTOR**

multi-tubular fixed bed, cooled by circulation of **molten salts**



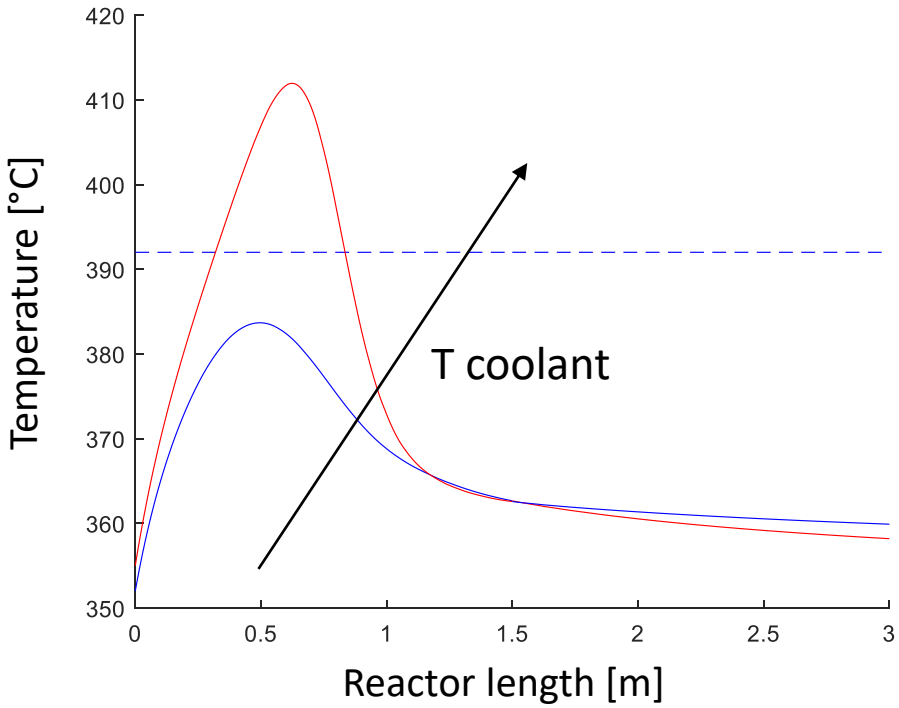
(1) $\mathbf{C_6H_4(CH_3)_2 + 3O_2 \rightarrow C_6H_4(CO)_2O + 3H_2O}$ **$\Delta H$=-1285409 kJ/kmol**

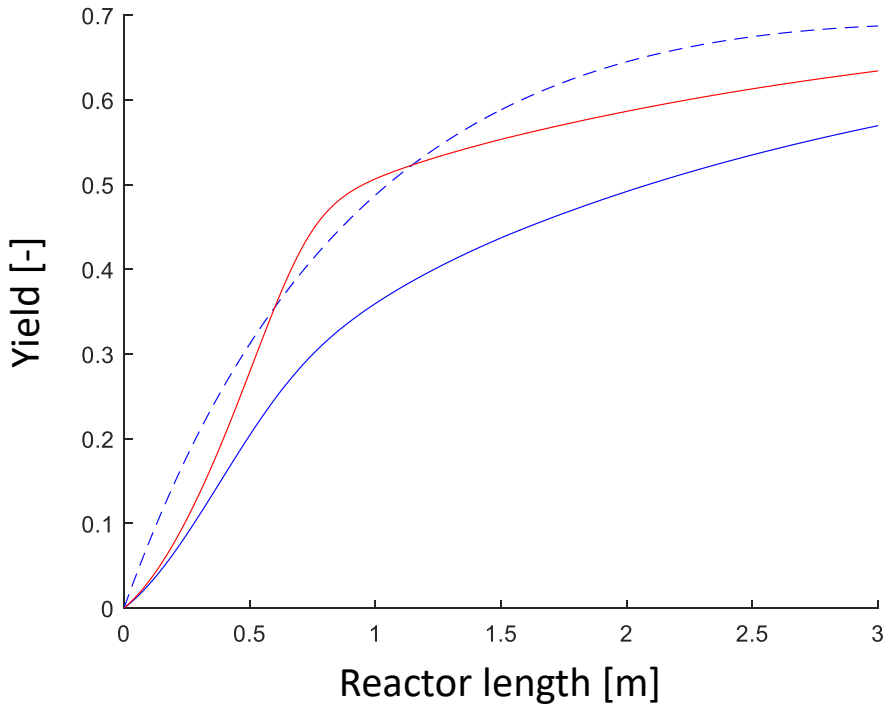
(2) $C_6H_4(CH_3)_2 + 10.5O_2 \rightarrow 8CO_2 + 5H_2O$ $\Delta H$=-4564000 kJ/kmol

(3) $C_6H_4(CO)_2O + +7.5\ O_2 \rightarrow 8CO_2 + 2H_2O$ $\Delta H$=-3278591 kJ/kmol

# *The importance of the reactor: tailoring the environment.*

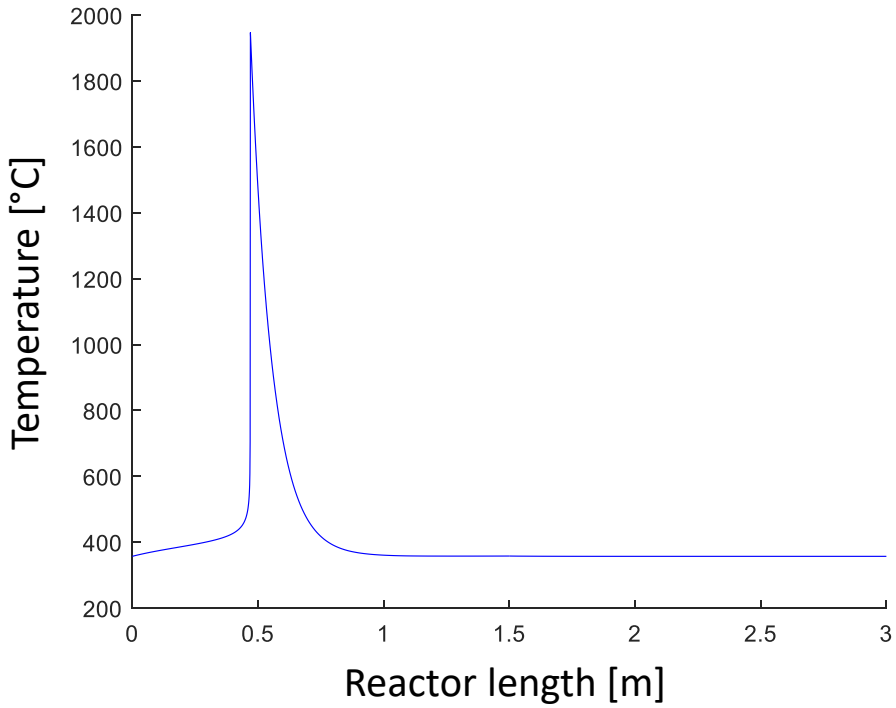
multi-tubular fixed bed, cooled by circulation of **molten salts**

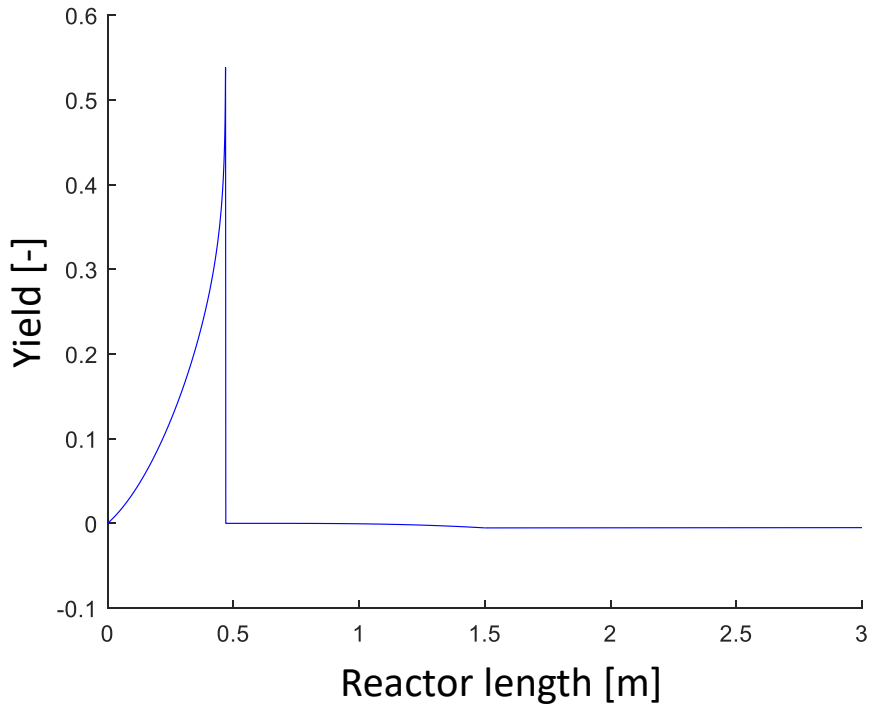


# *The importance of the reactor: tailoring the environment.*

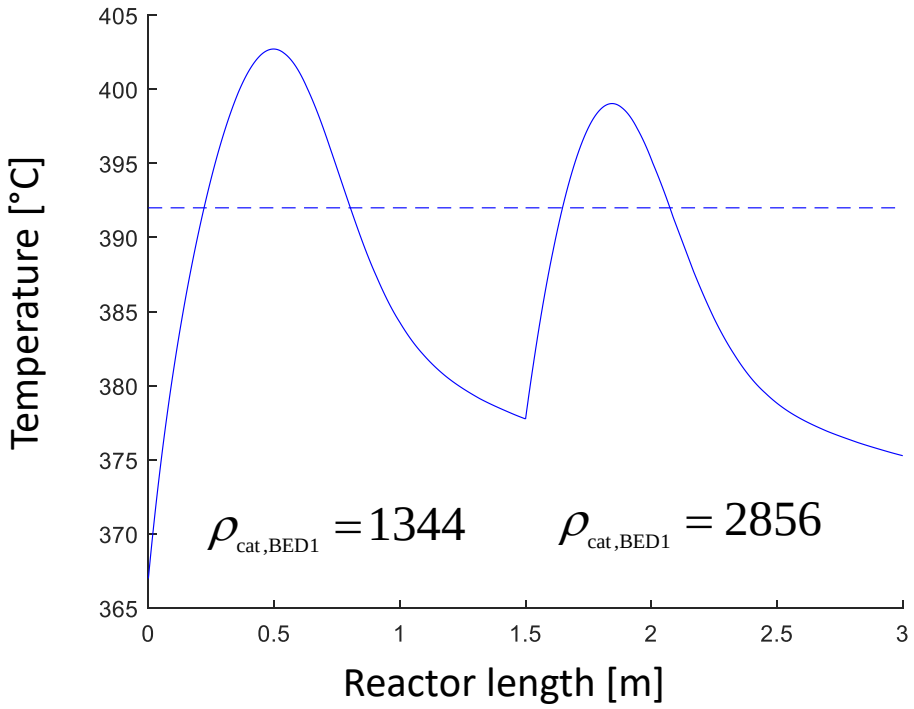
multi-tubular fixed bed, cooled by circulation of **molten salts**

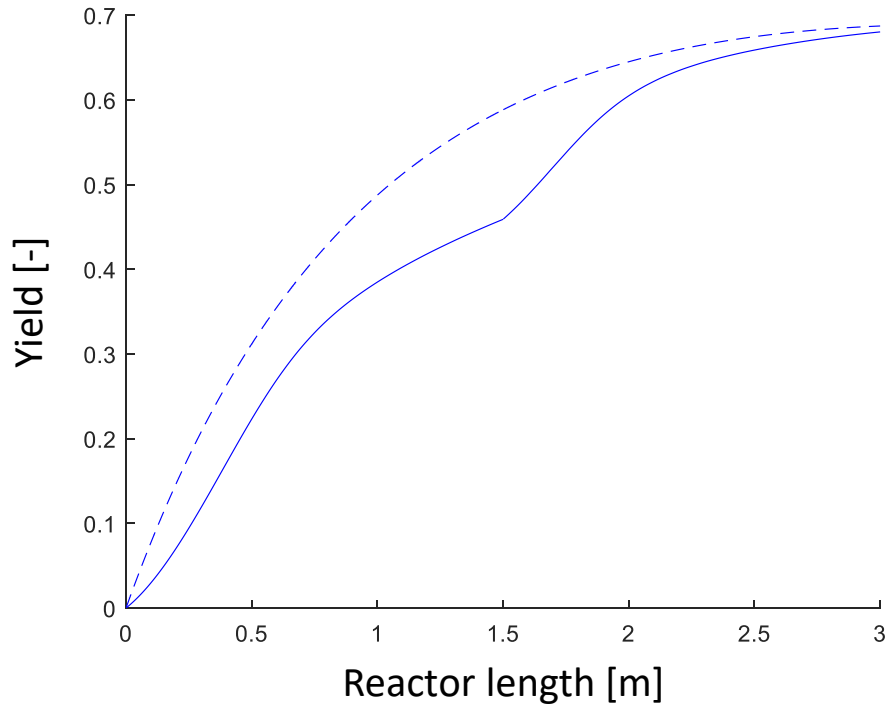


# *The importance of the reactor: tailoring the environment.*

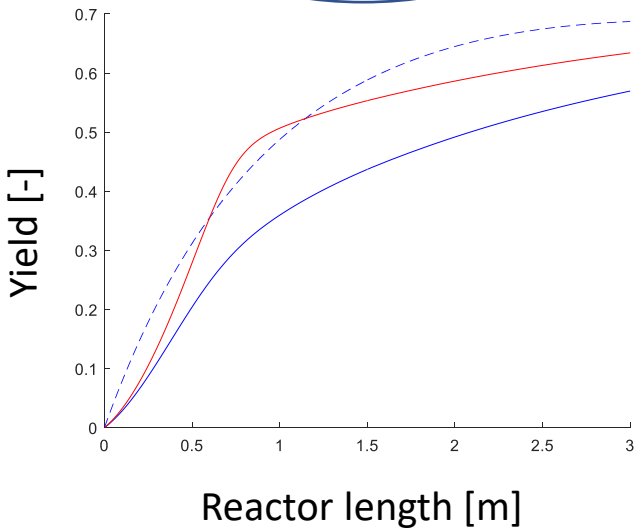
multi-tubular fixed multi-bed, cooled by circulation of **molten salts**



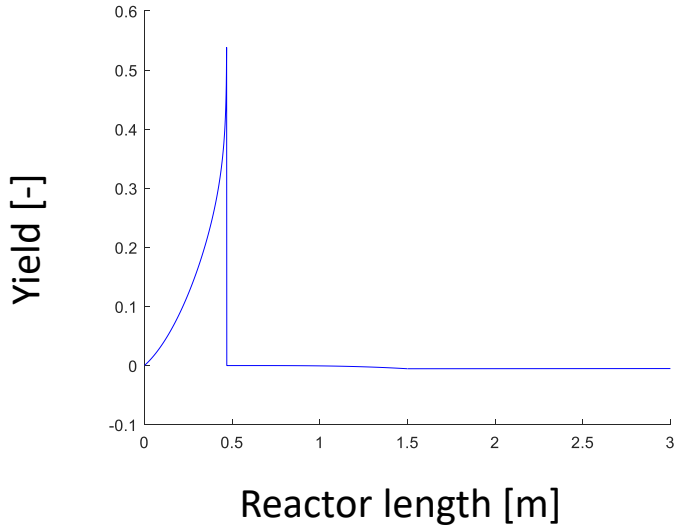



# *The importance of the reactor: tailoring the environment.*

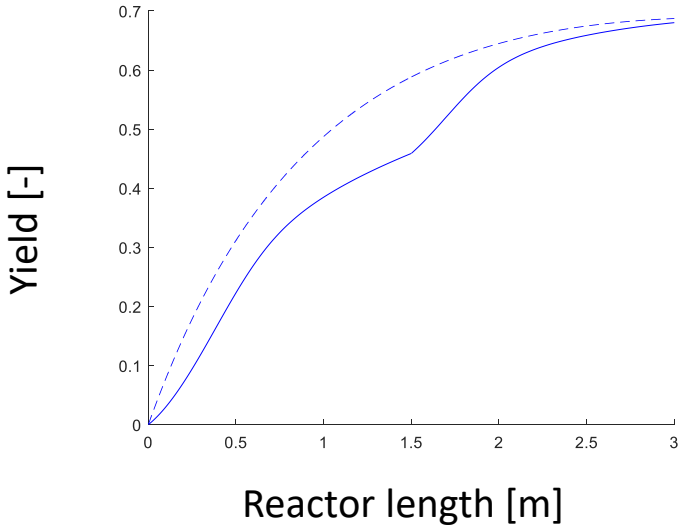
$V_2O_5$ - $TiO_2$




$V_2O_5$ - $TiO_2$

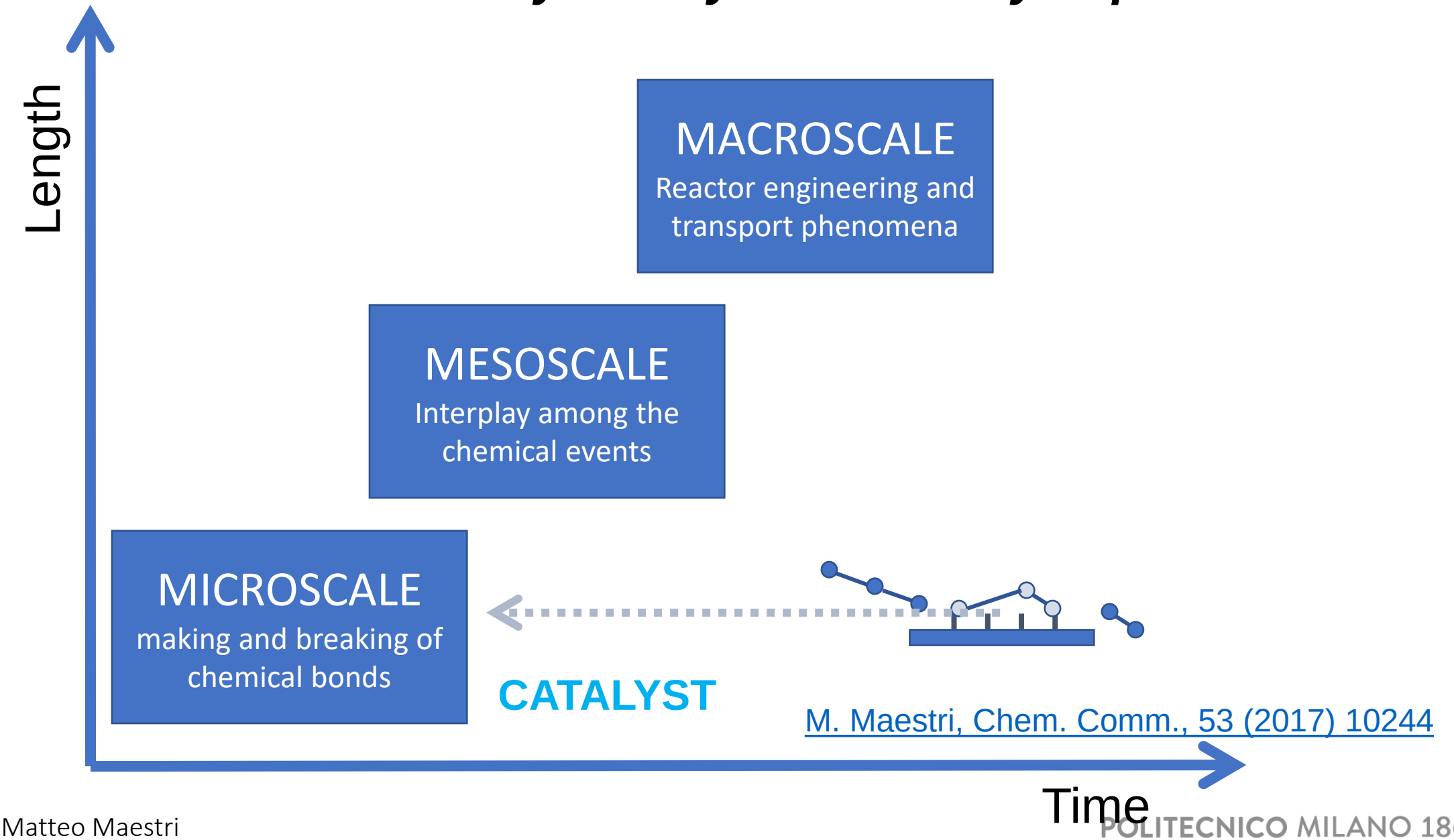



$V_2O_5$ - $TiO_2$



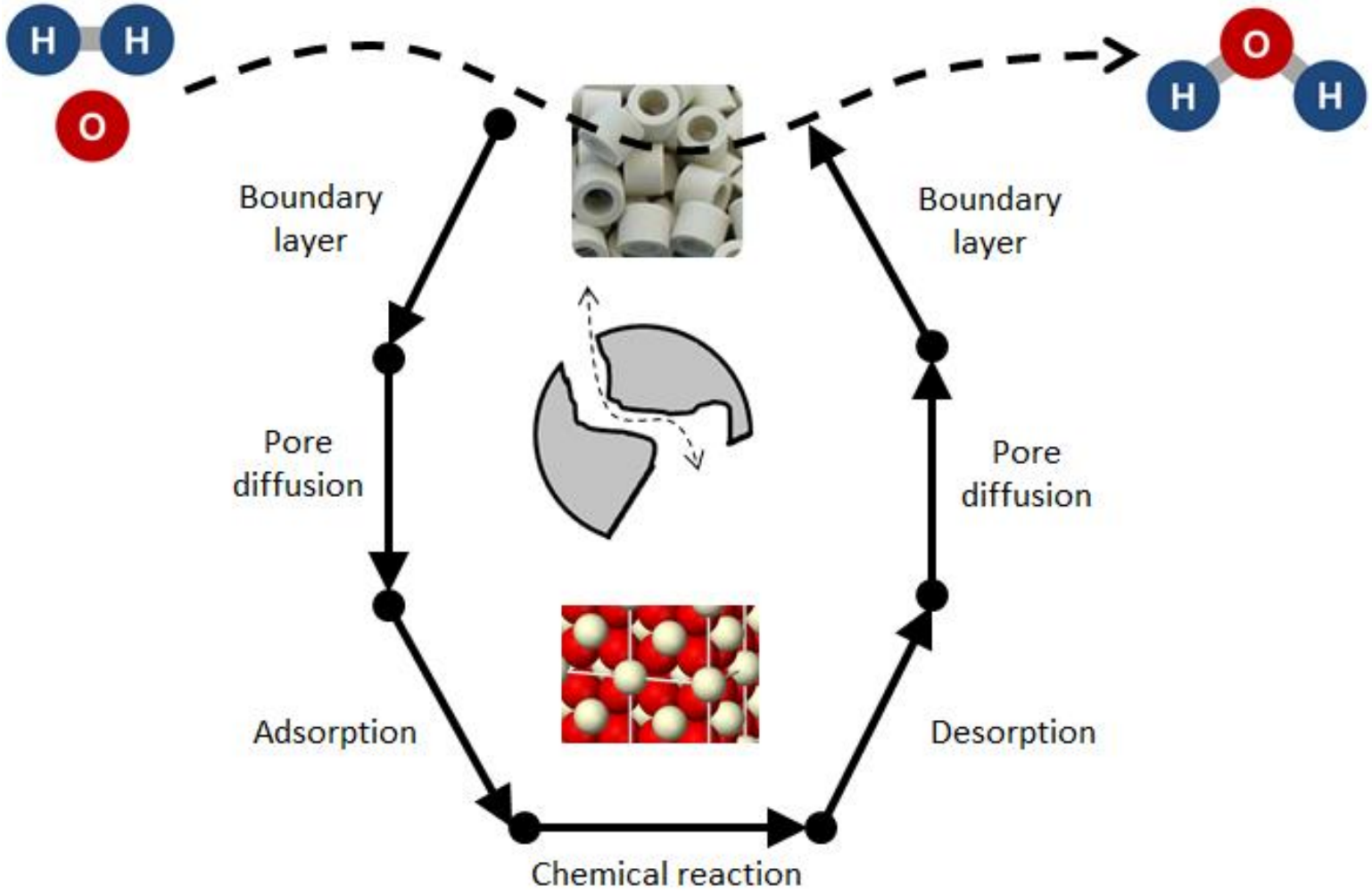

Reactor engineering is required for the design of an «optimal» environment for the active sites

# *A multiscale functionality: catalyst and catalytic process.*

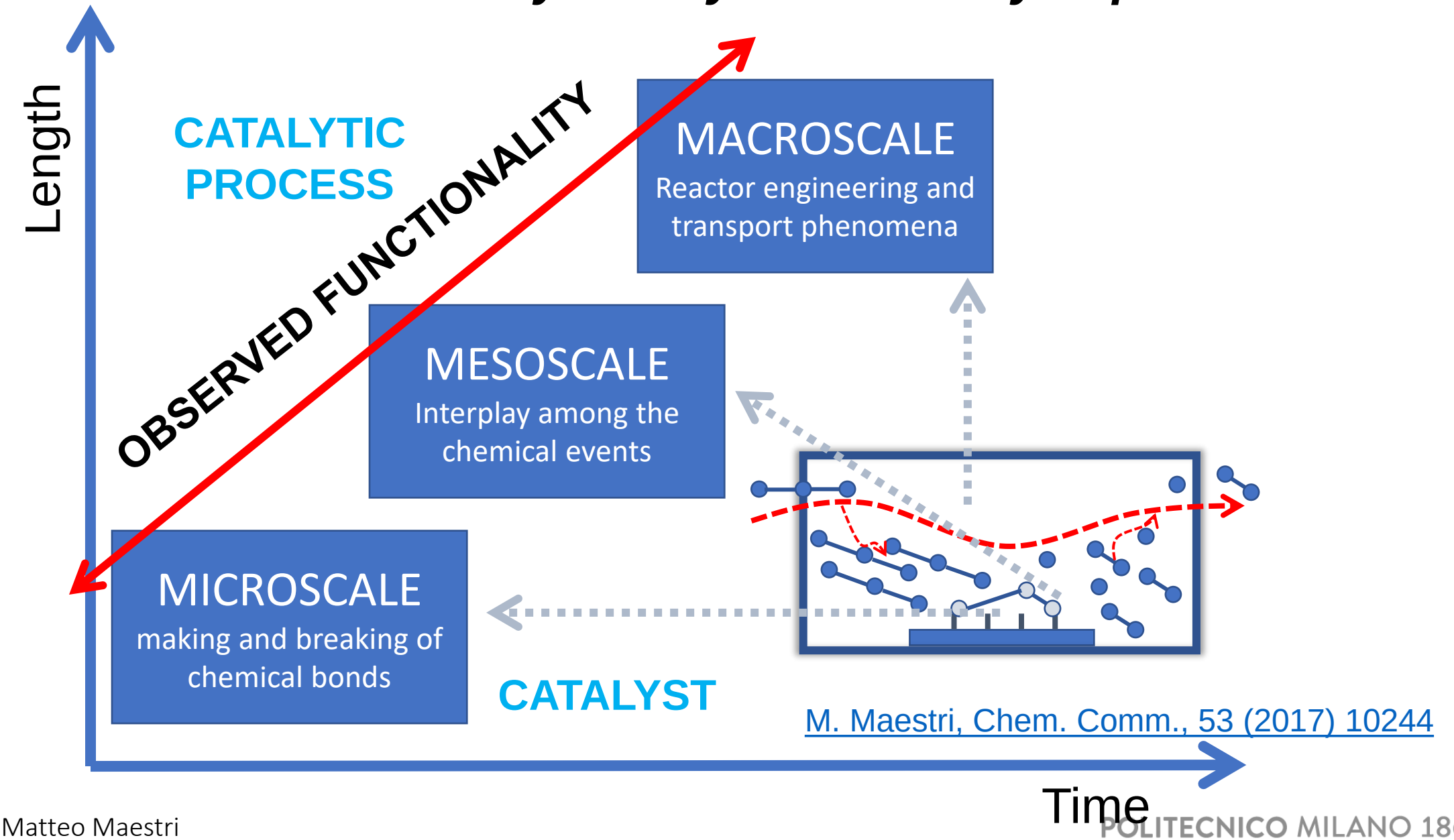



M. Maestri, Chem. Comm., 53 (2017) 10244

# *The long way to the active site*

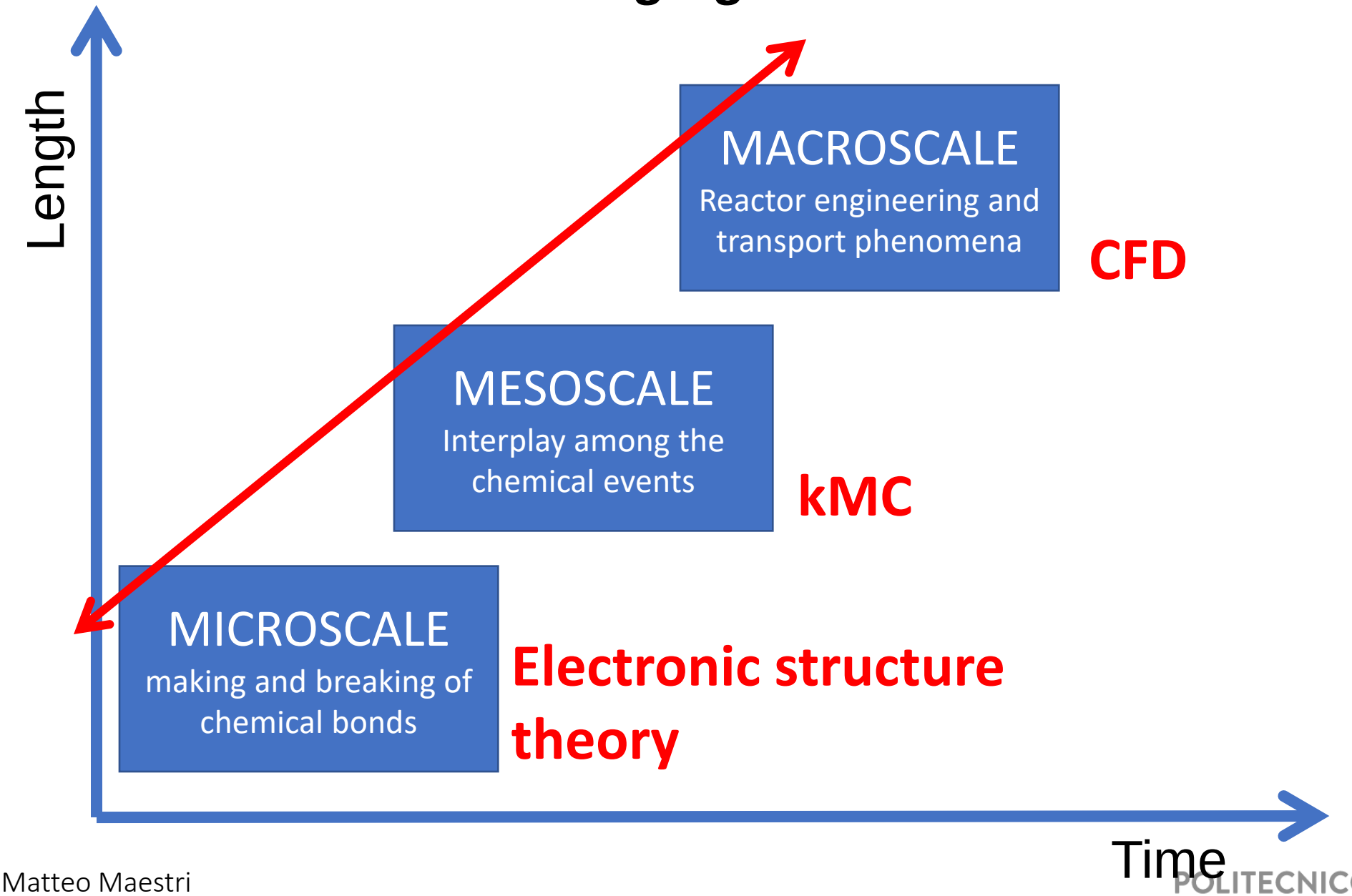


# *A multiscale functionality: catalyst and catalytic process.*

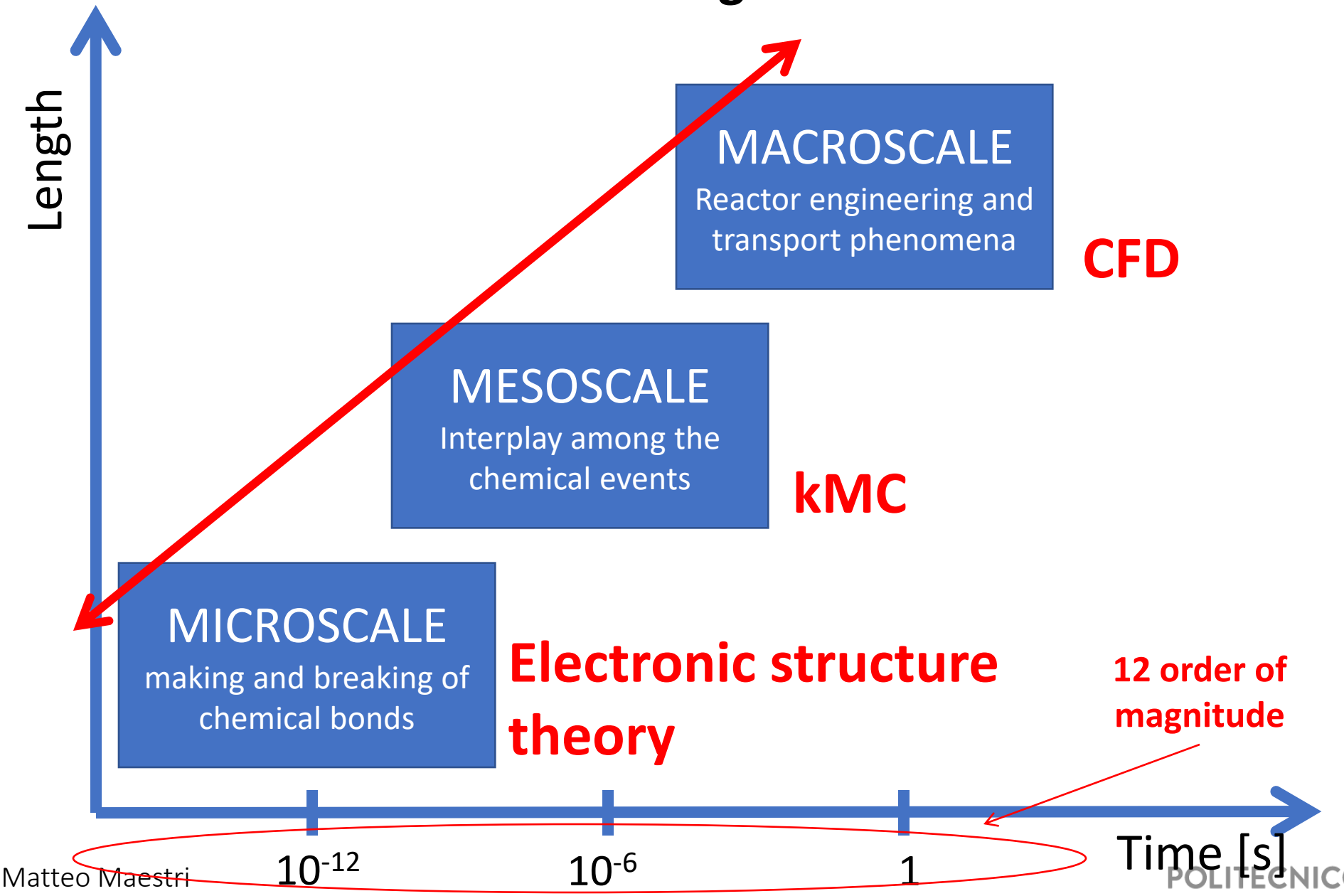



M. Maestri, Chem. Comm., 53 (2017) 10244

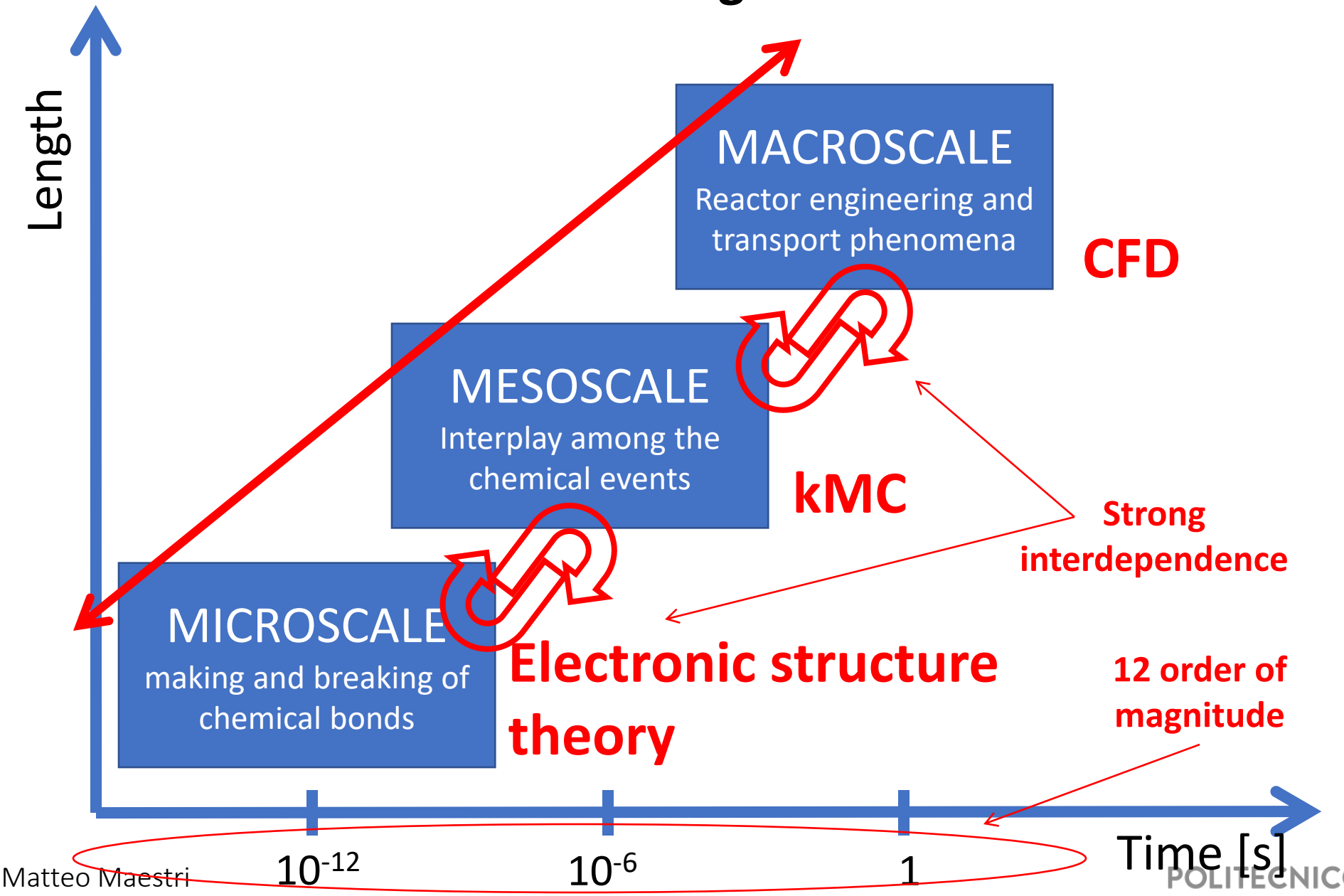
# *Need of bridging the scales.*



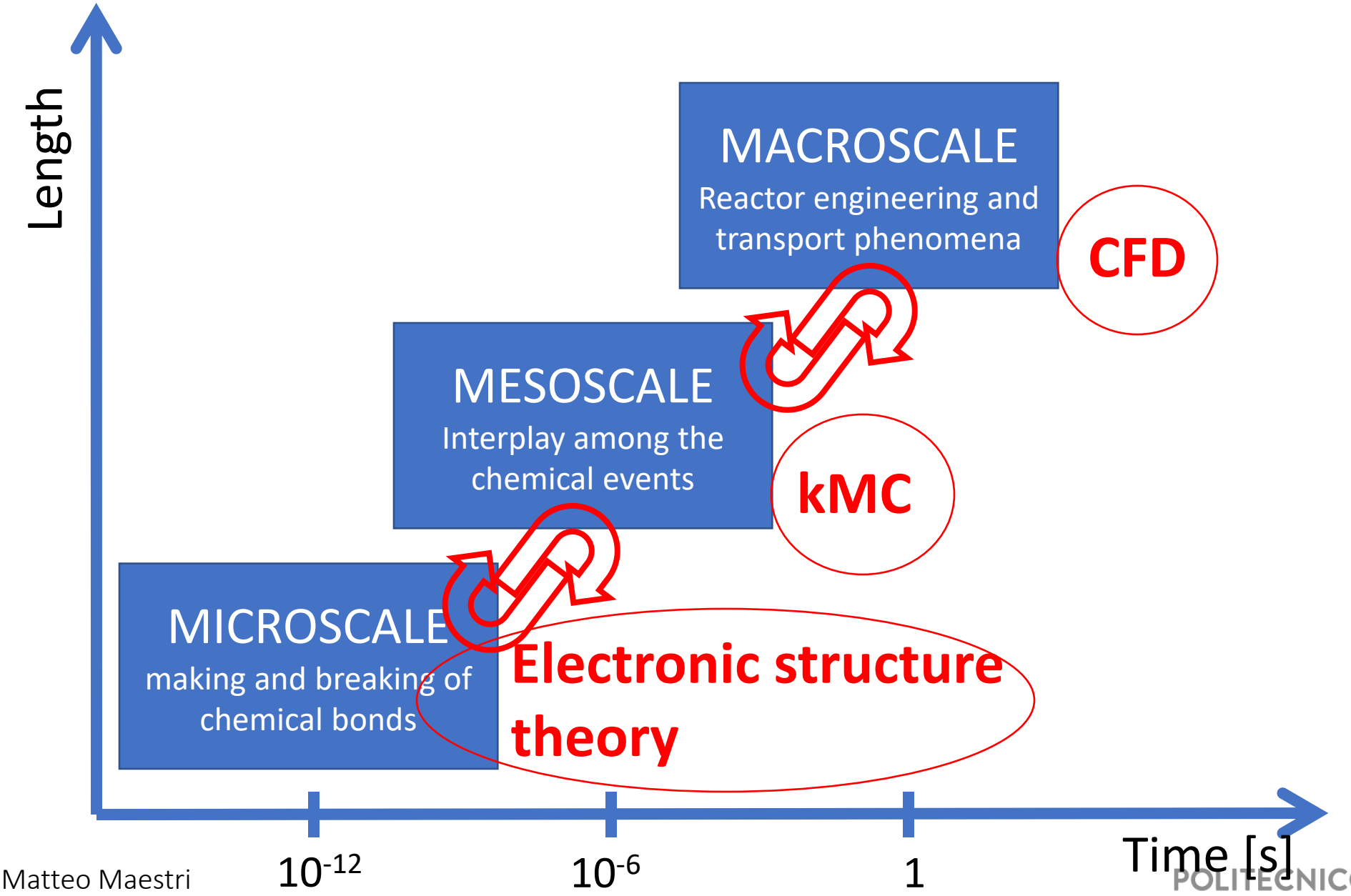
# *Challenges.*


Length
MACROSCALE
Reactor engineering and transport phenomena
CFD
MESOSCALE
Interplay among the chemical events
kMC
MICROSCALE
making and breaking of chemical bonds
Electronic structure theory
12 order of magnitude
$10^{-12}$
$10^{-6}$
1
Time [s]

# *Challenges.*

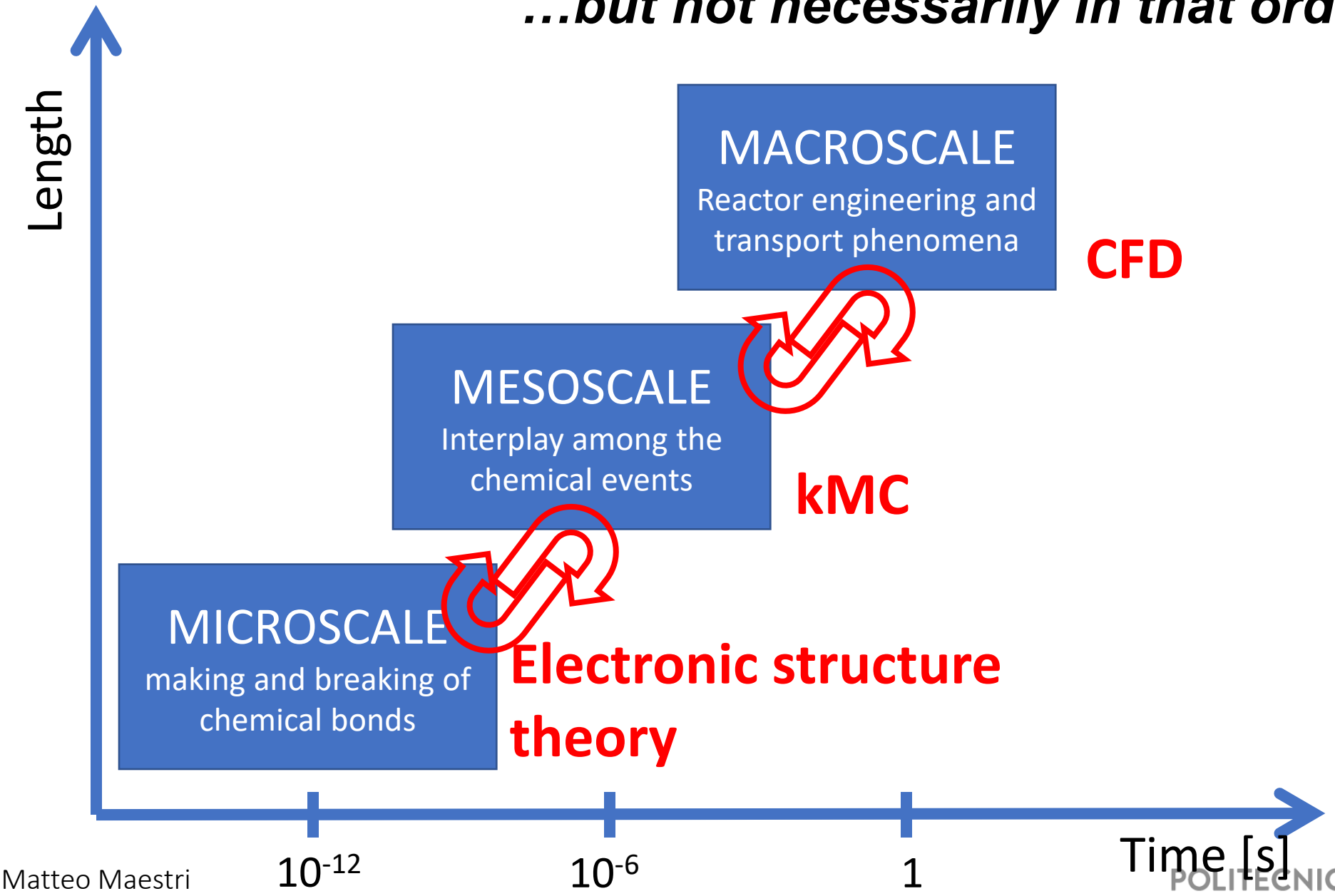

Length
MACROSCALE
Reactor engineering and transport phenomena
CFD
MESOSCALE
Interplay among the chemical events
kMC
Strong interdependence
MICROSCALE
making and breaking of chemical bonds
Electronic structure theory
12 order of magnitude
$10^{-12}$
$10^{-6}$
1
Time [s]

# *First things first…*



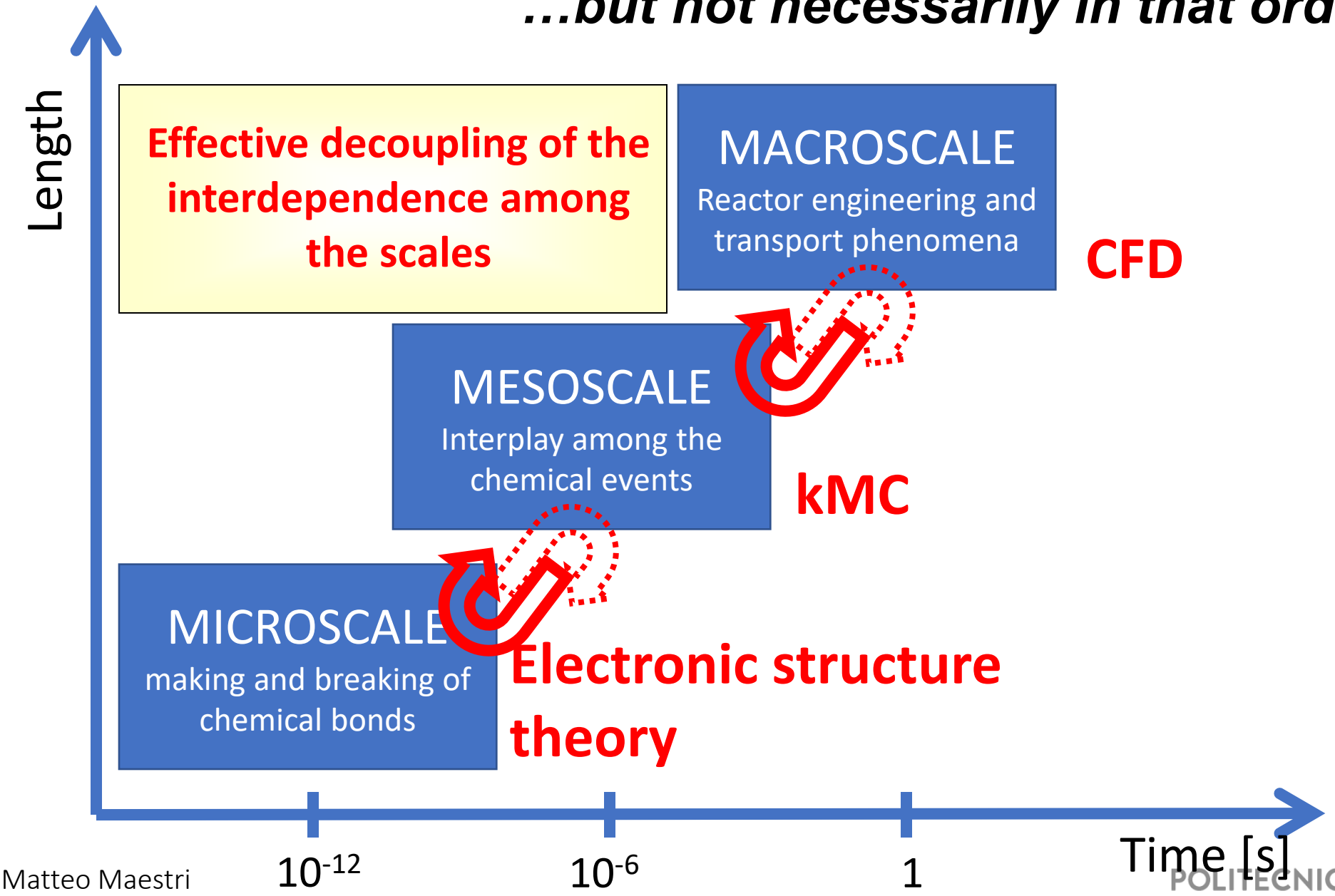
***First things first…***

***…but not necessarily in that order.***

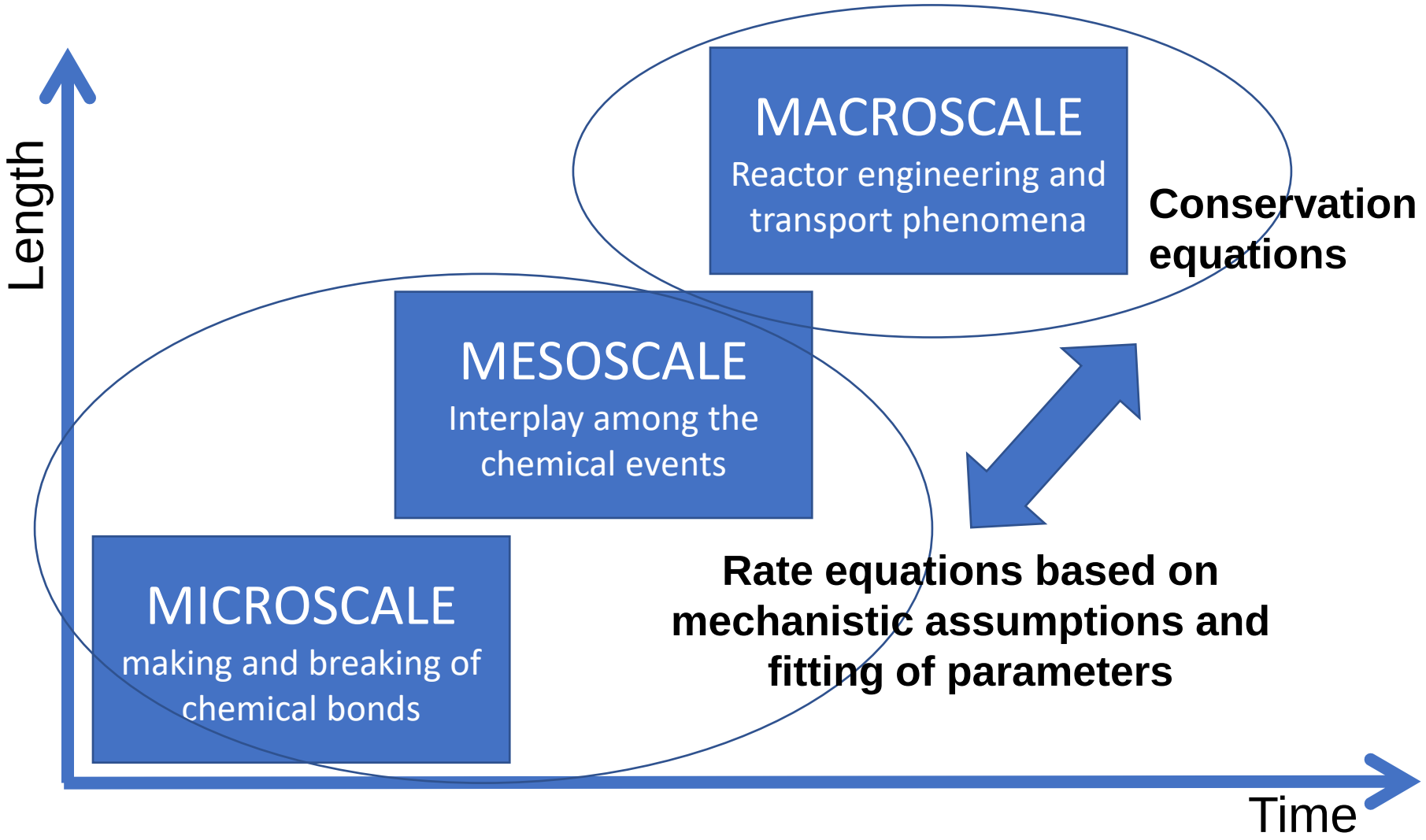


***First things first…***

***…but not necessarily in that order.***

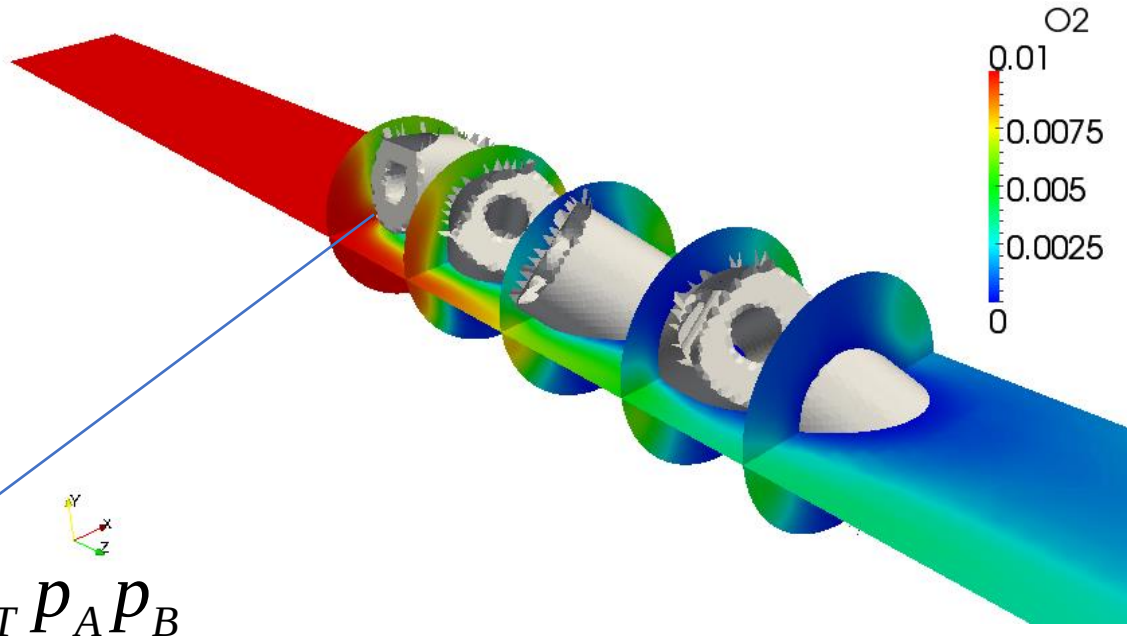


# *Example I: Rate equations and transport.*



# *Example I: Rate equations and transport.*

- ✓ CFD
- ✓ 1D/2D heterogeneous or pseudohomogeneous models

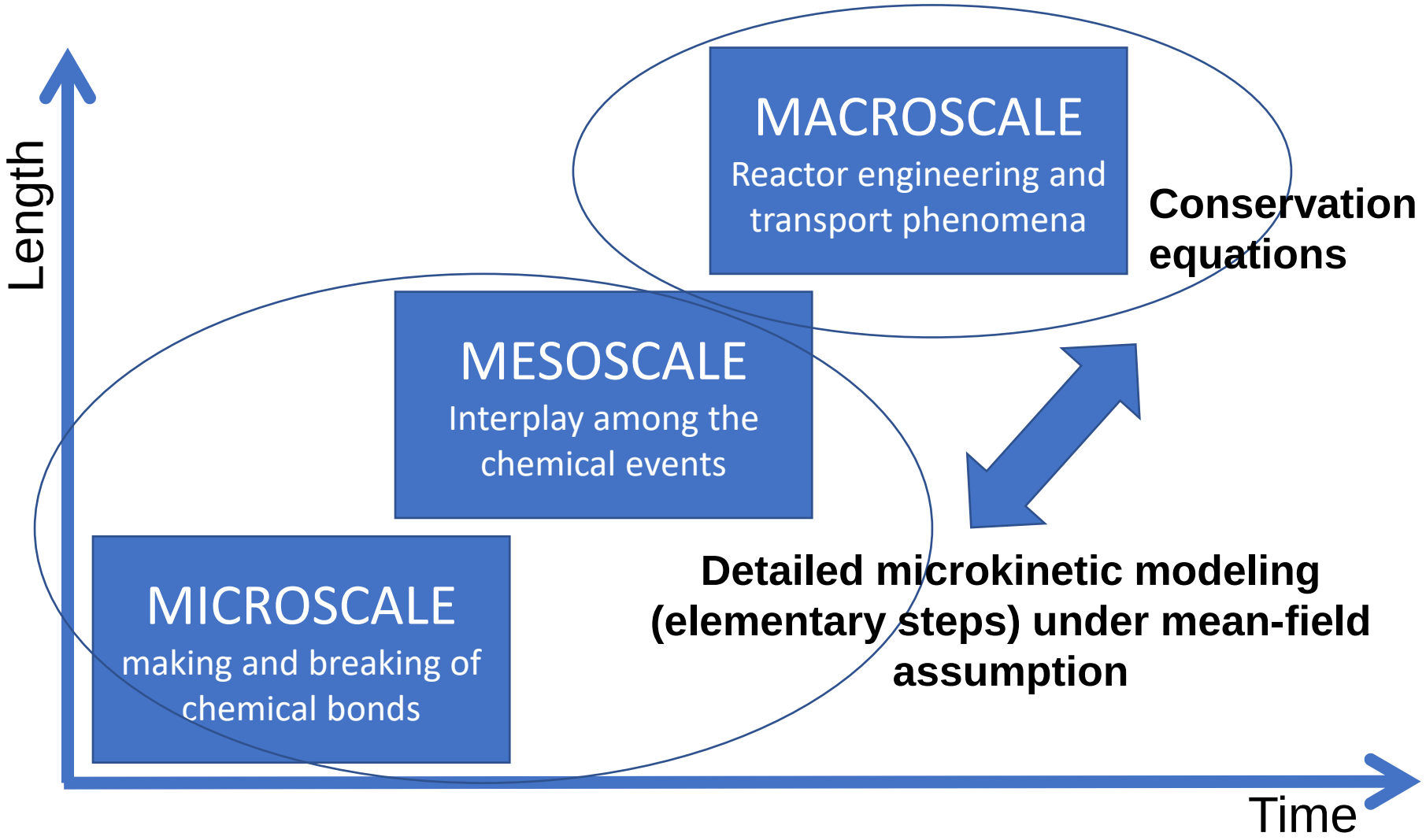



$$r = \frac{k_{TOT}\, p_A p_B}{\left(1 + K_1 p_A + K_2 p_B + K_3 p_C + K_4 p_D\right)^2}$$

Mechanistic information (**RDS**, **MASI**, …) are fully retained in the rate equation and **do not change as a function of the operating conditions** –

**The reaction mechanism is an assumption**

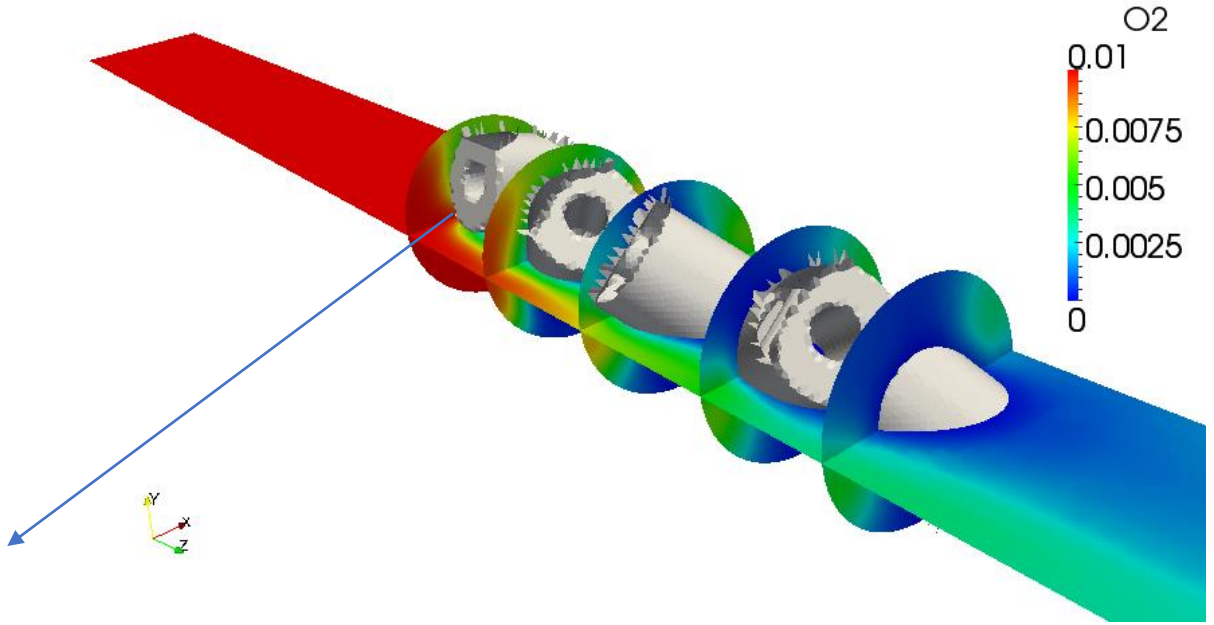
# *Example II: microkinetic modeling and transport.*



# *Example II: microkinetic modeling and transport.*

- ✓ CFD
- ✓ 1D/2D heterogeneous or pseudohomogeneous models

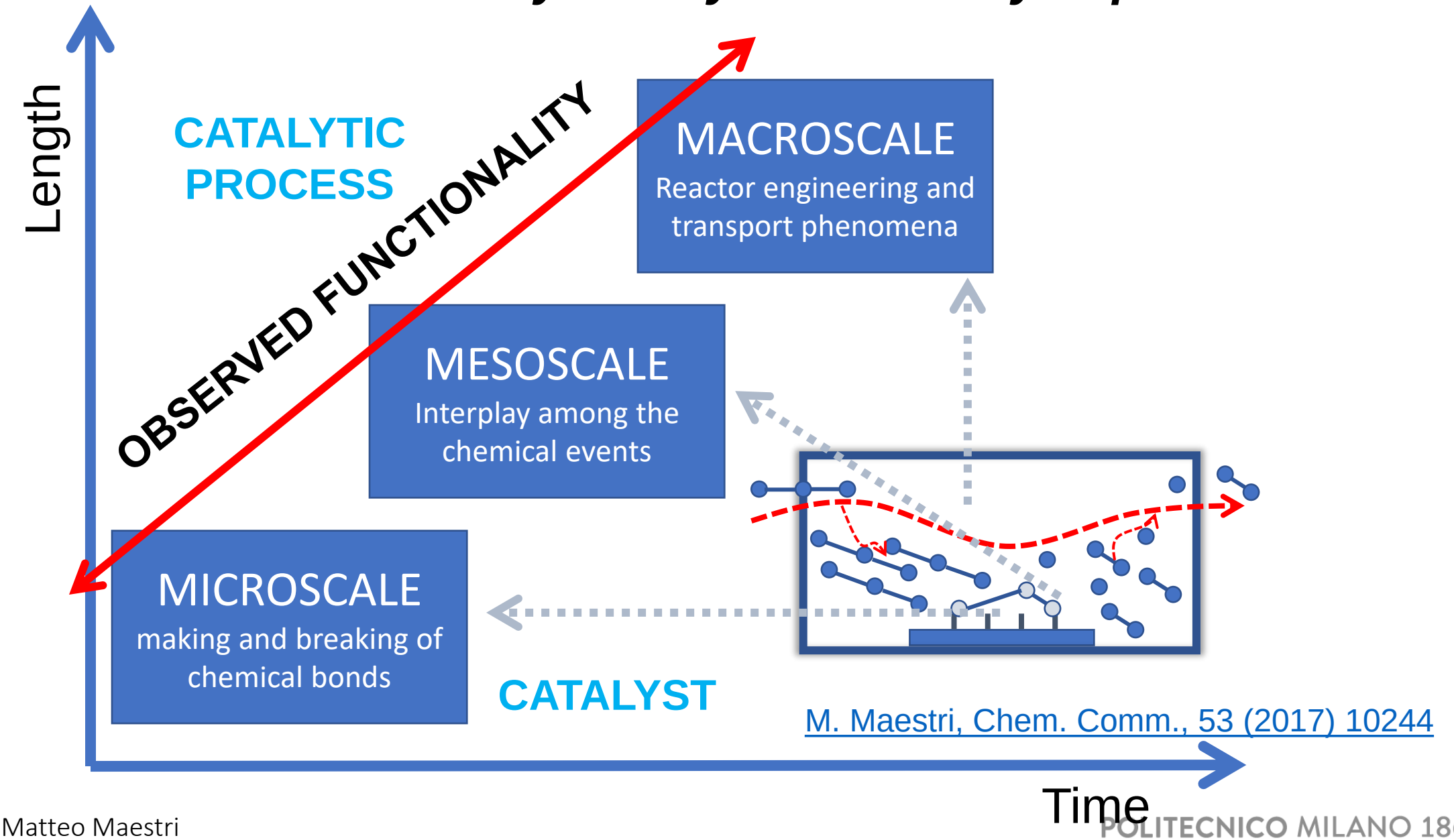
$COOH^*+^* \rightarrow CO^*+OH^*$
$CO^*+OH^* \rightarrow COOH^*+^*$
$COOH^*+^* \rightarrow CO_2^*+H^*$
$CO_2^*+H^* \rightarrow COOH^*+^*$
$CO_2^* + H_2O^* \rightarrow COOH^* + OH^*$
$COOH^* + OH^* \rightarrow CO_2^* + H_2O^*$
$CO_2^*+H^* \rightarrow HCOO^{**}$
$HCOO^{**} \rightarrow CO_2^*+H^*$
$CO_2^* + OH^* + ^* \rightarrow HCOO^{**} + O^*$
$HCOO^{**} + OH^* \rightarrow CO_2^* + H_2O^*$
$CH^* + H^* \rightarrow CH_2^* + ^*$
$CH^* + ^* \rightarrow C^* + H^*$
$C^* + H^* \rightarrow CH^* + ^*$
$CH_3^* + O^* \rightarrow CH_2^* + OH^*$
$CH_2^* + OH^* \rightarrow CH_3^* + O^*$
$CH^* + OH^* \rightarrow CH_2^* + O^*$
$CH_2^* + O^* \rightarrow CH^* + OH^*$
…




**Elementary-like steps** are accounted for explicitly – **RDS and MASI can change** with respect to the different conditions

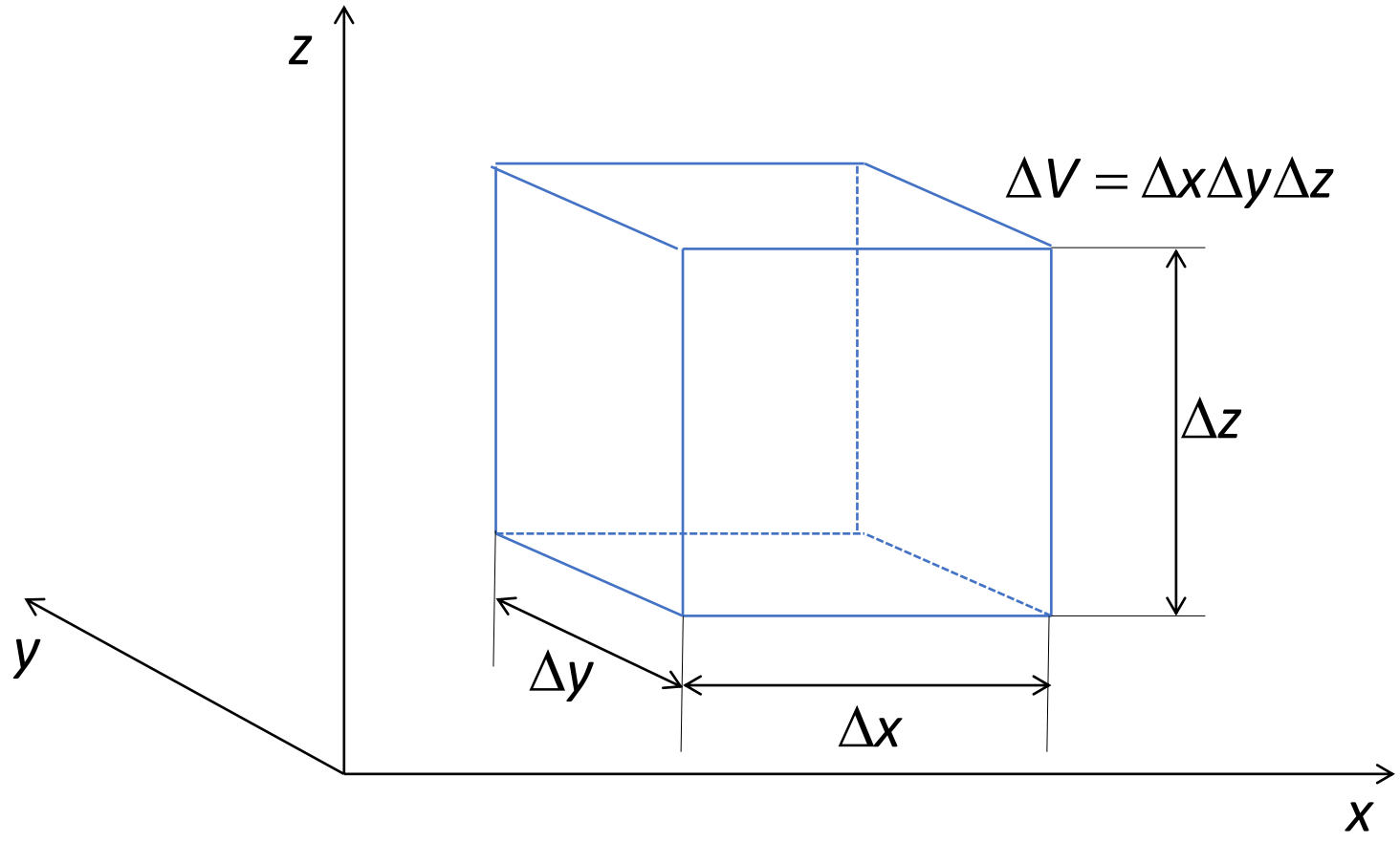
The **reaction mechanism is a result of the simulation**, not an assumption

# *A multiscale functionality: catalyst and catalytic process.*

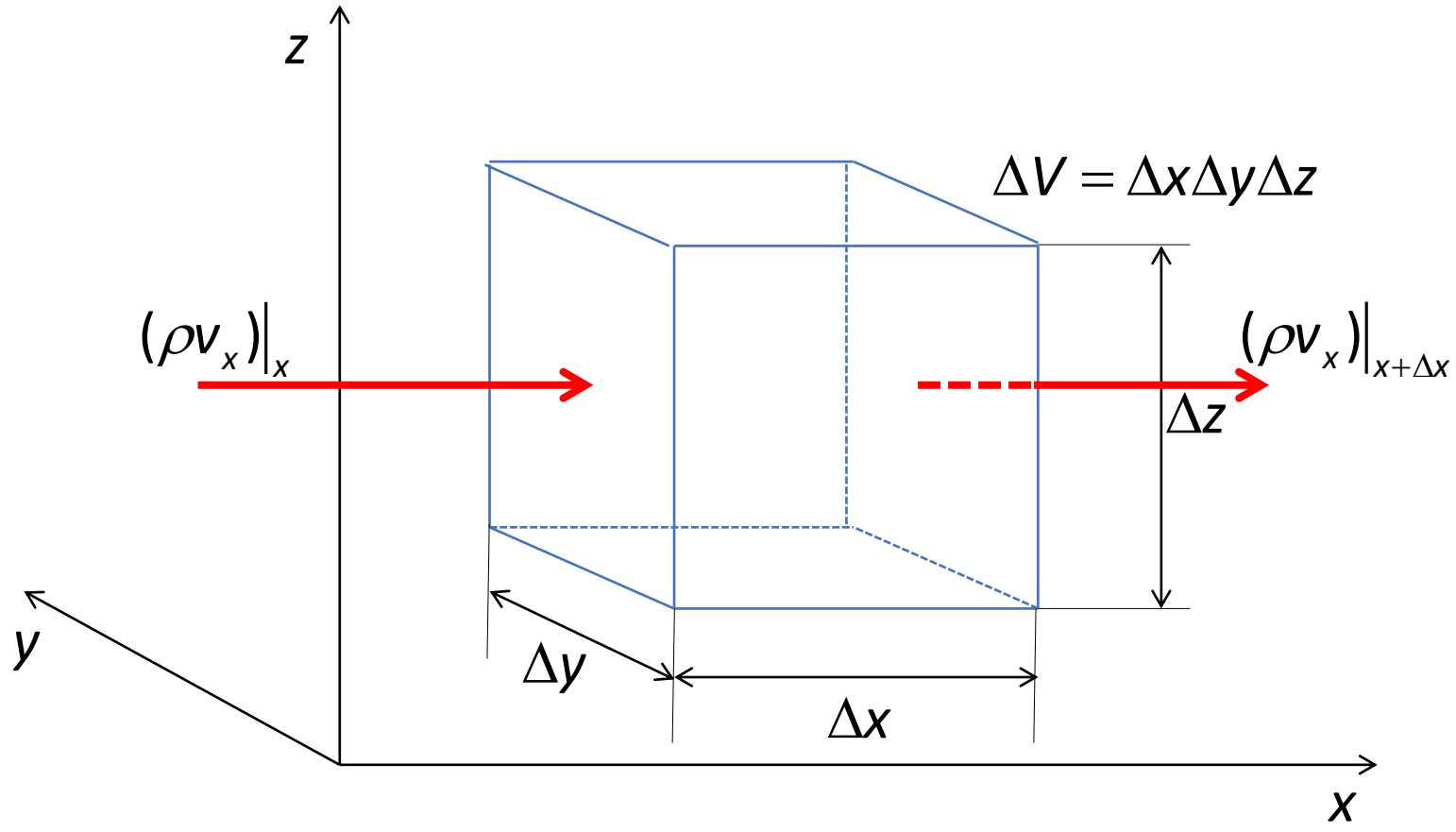



M. Maestri, Chem. Comm., 53 (2017) 10244

# *Governing equation at the macroscale.*

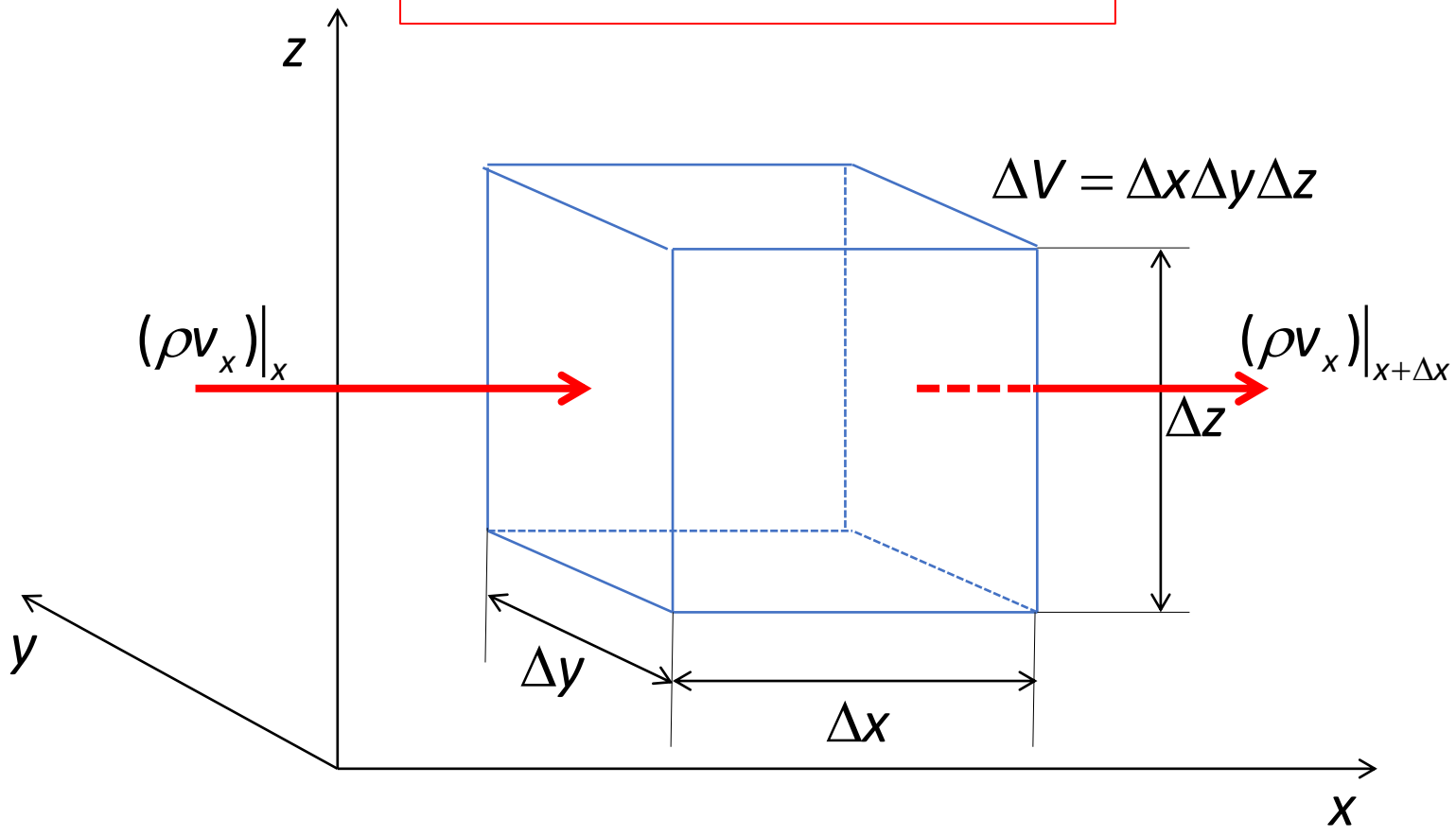


# Derivation of the continuity equation.



# *Derivation of the continuity equation.*

IN – OUT + PROD = ACC



# *Derivation of the continuity equation.*

IN – OUT + PROD = ACC

$$\Delta y \Delta z \left[ (\rho v_x)\big|_x - (\rho v_x)\big|_{x+\Delta x} \right] + \Delta y \Delta x \left[ (\rho v_z)\big|_z - (\rho v_z)\big|_{z+\Delta z} \right] +$$

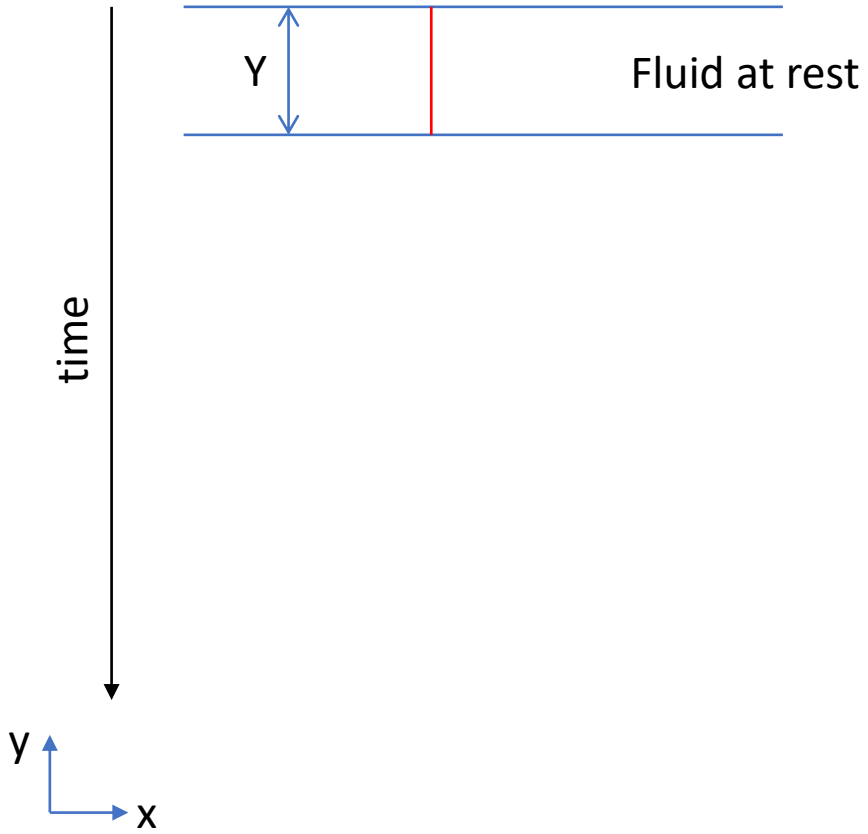
$$+ \Delta x \Delta z \left[ (\rho v_y)\big|_y - (\rho v_y)\big|_{y+\Delta y} \right] = \Delta x \Delta y \Delta z \frac{\partial \rho}{\partial t}$$

By dividing by V and taking the limit as x,y, and z go to zero:

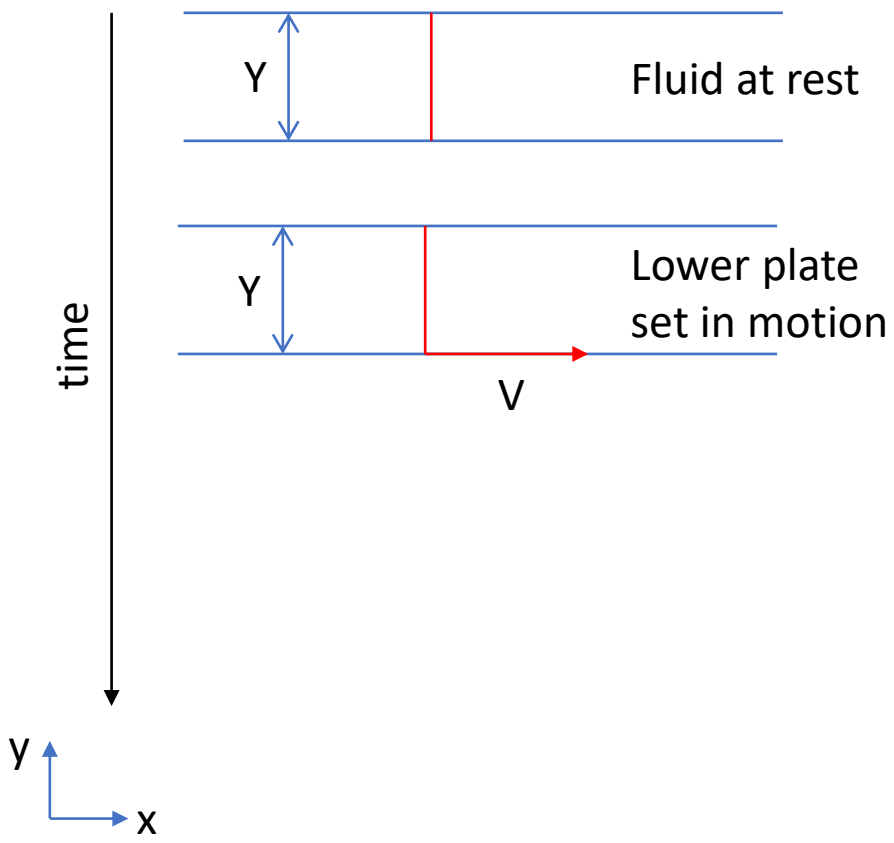
$$\frac{\partial \rho}{\partial t} = -\left( \frac{\partial}{\partial x} \rho v_x + \frac{\partial}{\partial y} \rho v_y + \frac{\partial}{\partial z} \rho v_z \right)$$

$$\frac{\partial \rho}{\partial t} = -(\nabla \cdot \rho \mathbf{v})$$

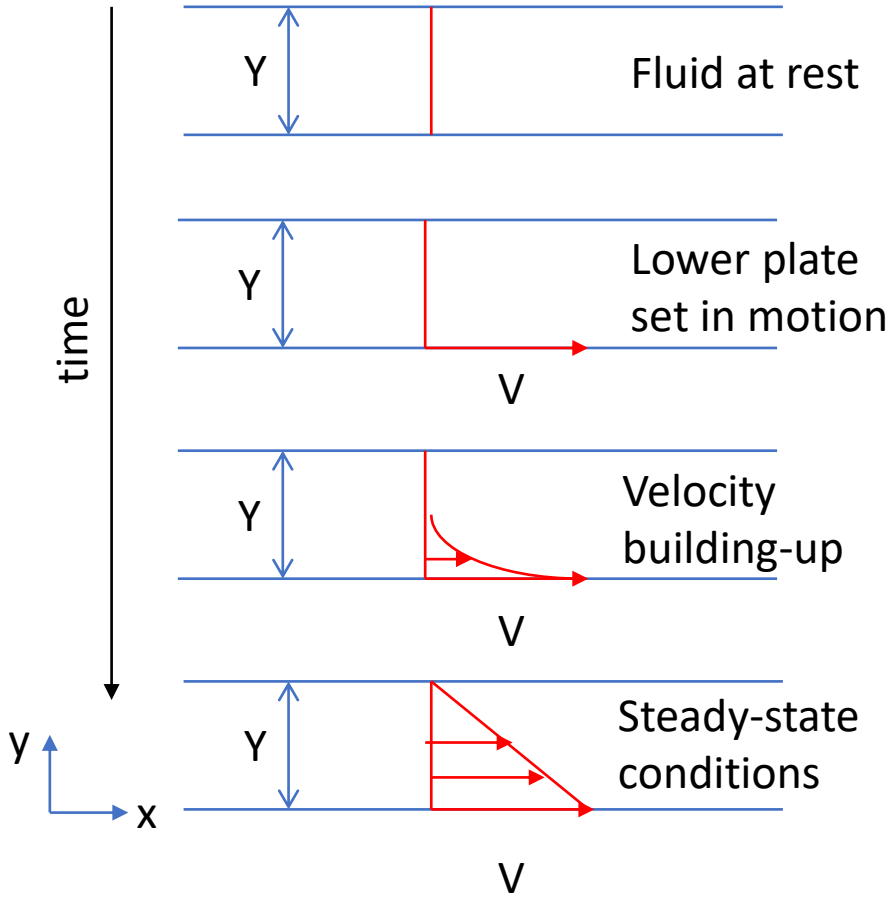
# *Momentum flux*



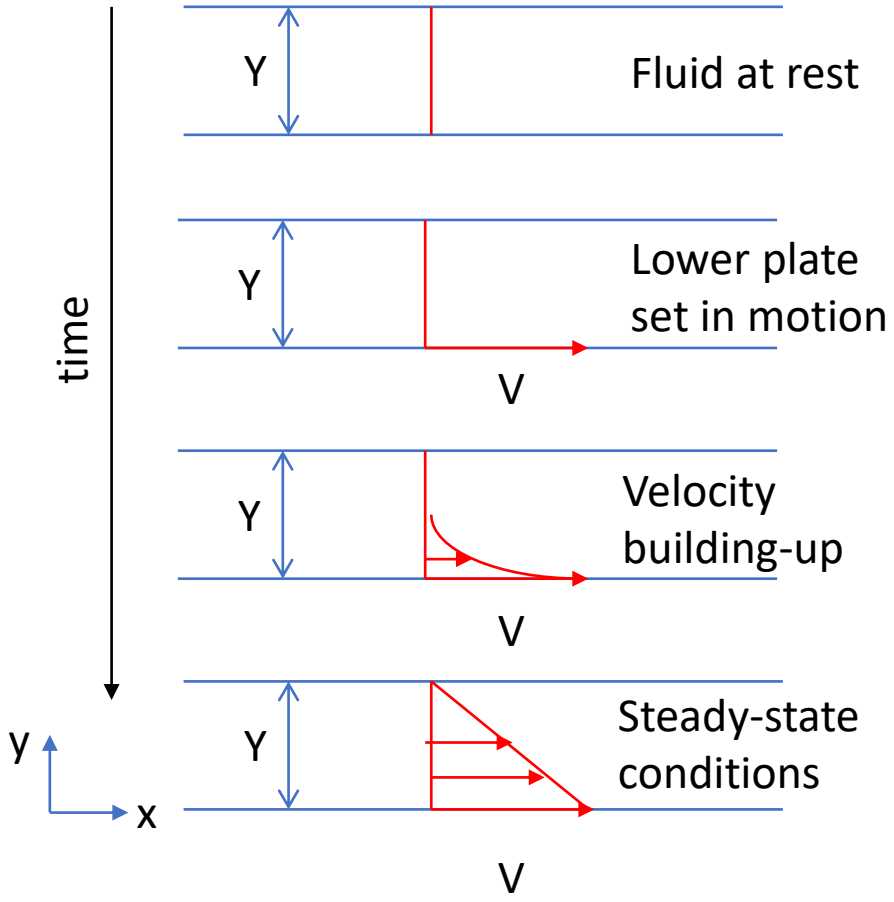
# *Momentum flux*



# *Momentum flux*

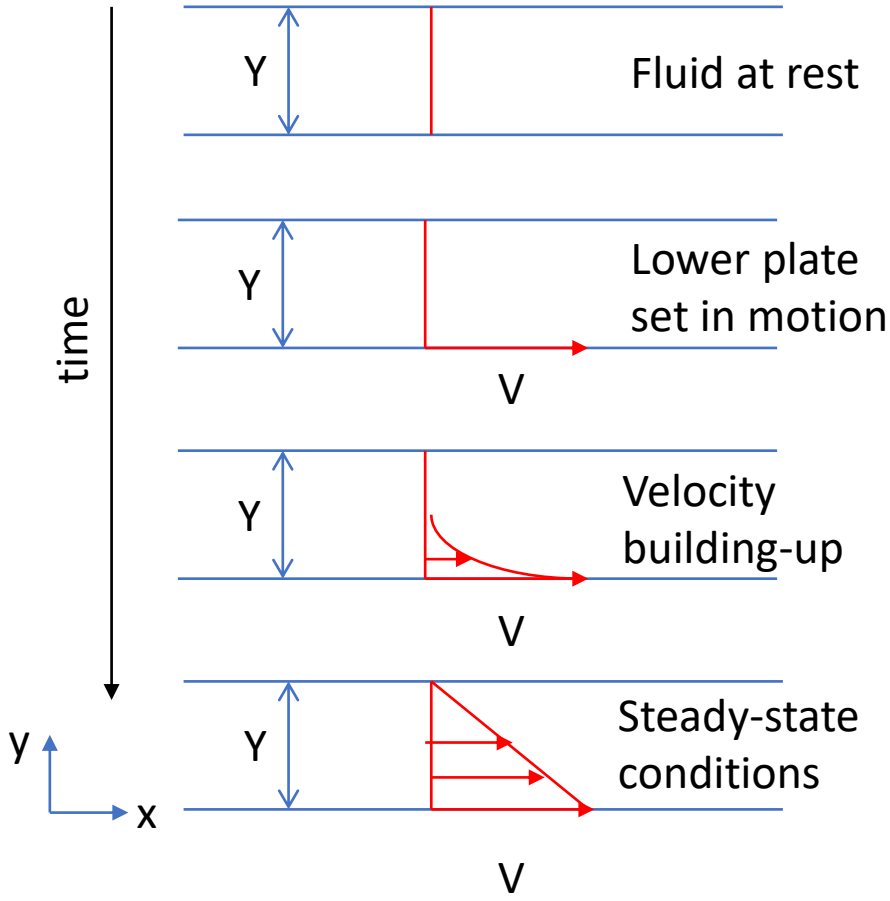


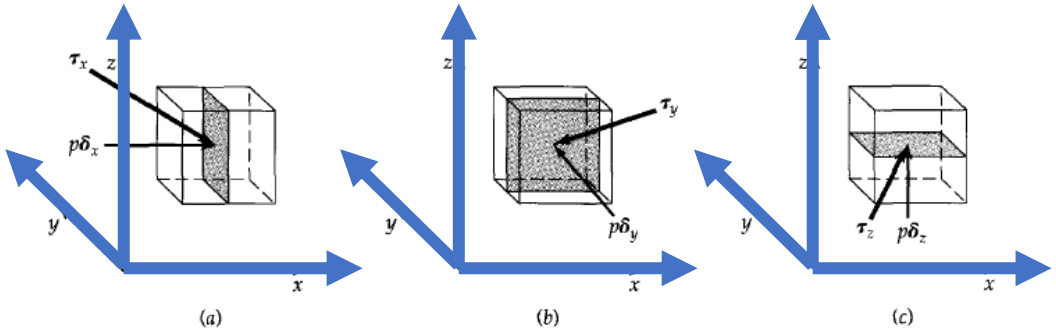
# *Momentum flux*




$$\frac{F}{A} = \mu \frac{V}{Y} \rightarrow \tau_{yx} = -\mu \frac{dv_x}{dy}$$

# Momentum flux

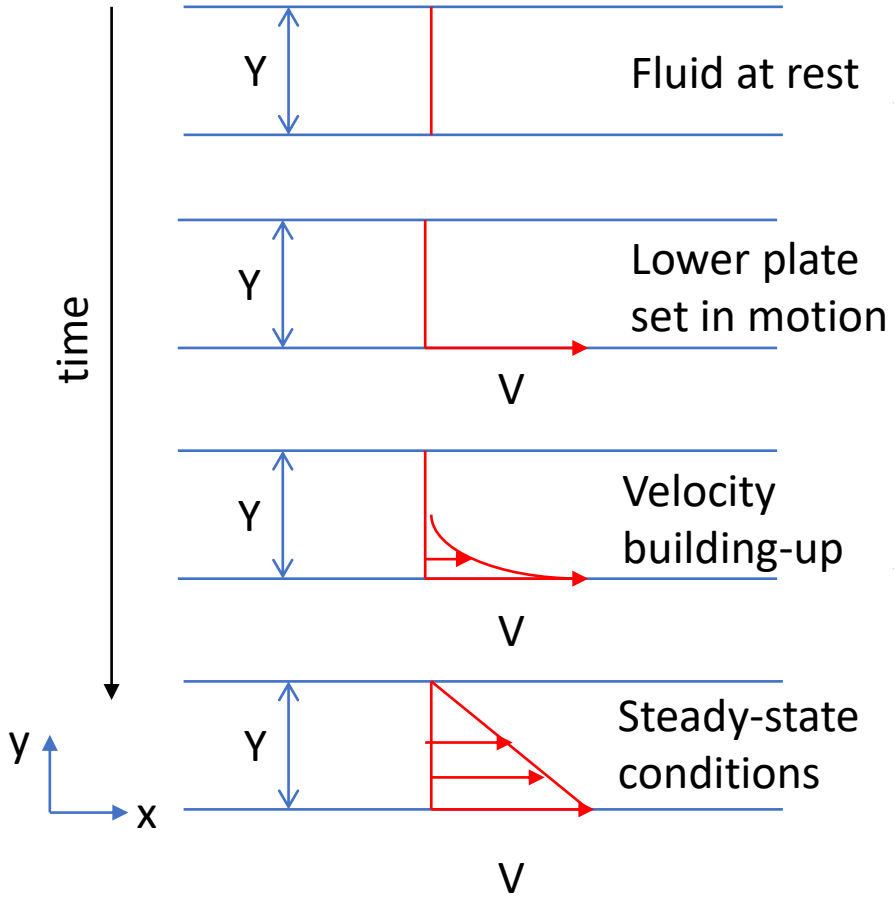


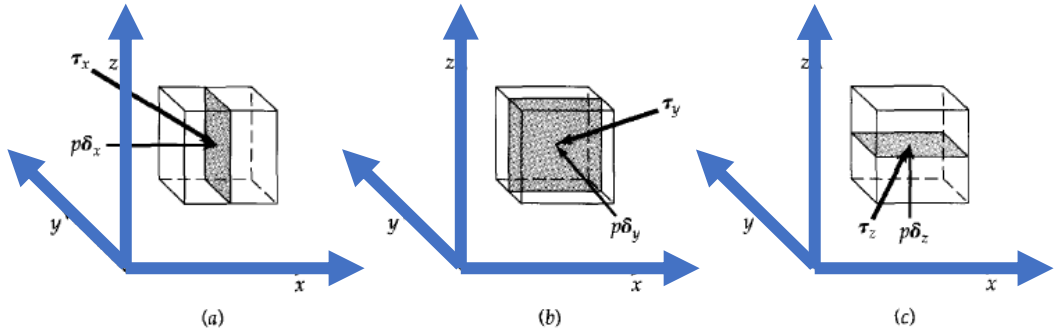

$$\pi_{ij} = p\delta_{ij} + \tau_{ij}$$

$$\frac{F}{A} = \mu\frac{V}{Y} \rightarrow \tau_{yx} = -\mu\frac{dv_x}{dy}$$

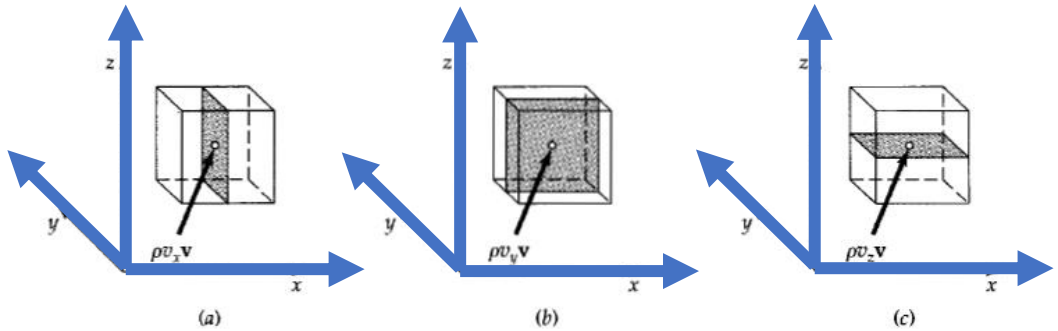
# Momentum flux




$$\frac{F}{A} = \mu \frac{V}{Y} \rightarrow \tau_{yx} = -\mu \frac{dv_x}{dy}$$

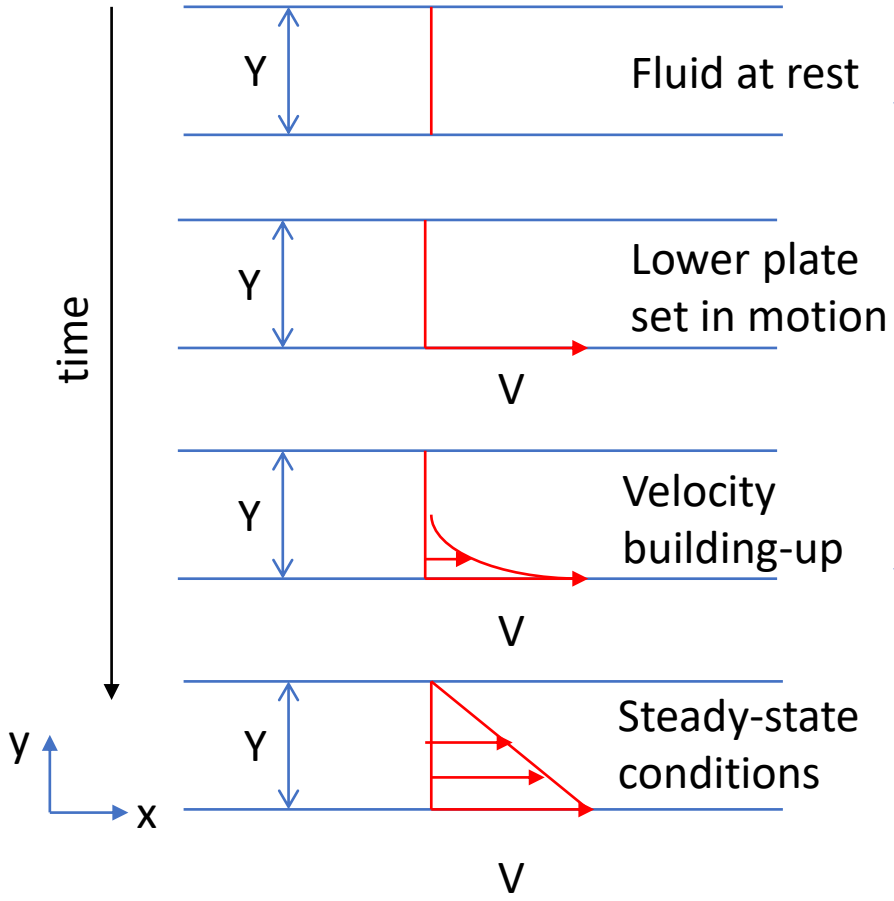



$$\pi_{ij} = p\delta_{ij} + \tau_{ij}$$

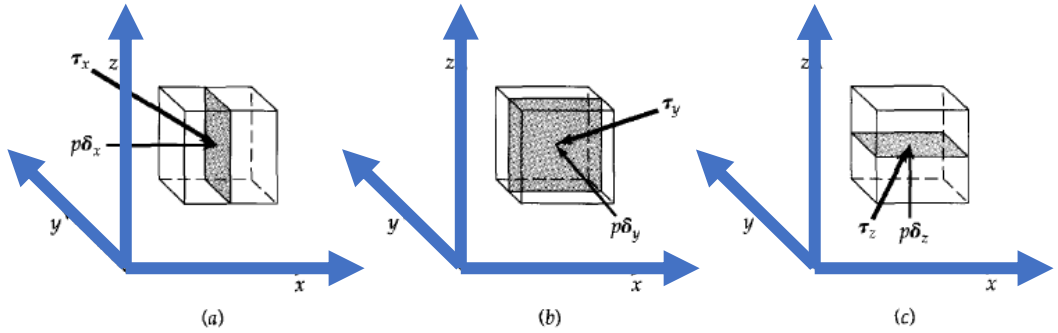



$$\rho \mathbf{vv} = \sum_i (\rho \delta_i v_i) \mathbf{v}$$

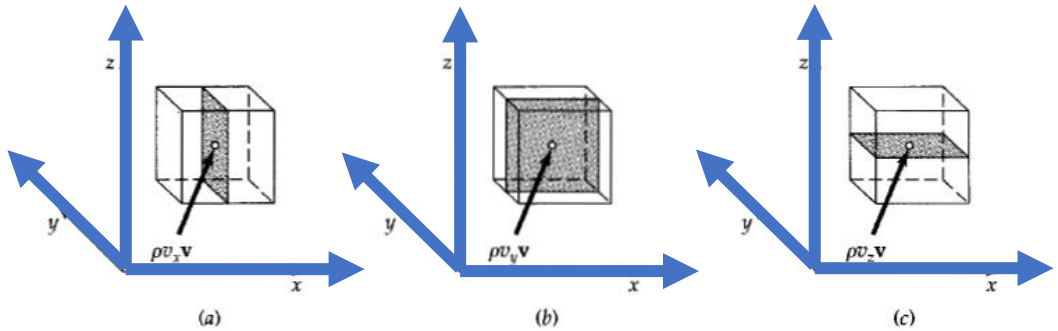
# Momentum flux




$$\frac{F}{A} = \mu \frac{V}{Y} \rightarrow \tau_{yx} = -\mu \frac{dv_x}{dy}$$




$$\pi_{ij} = p\delta_{ij} + \tau_{ij}$$



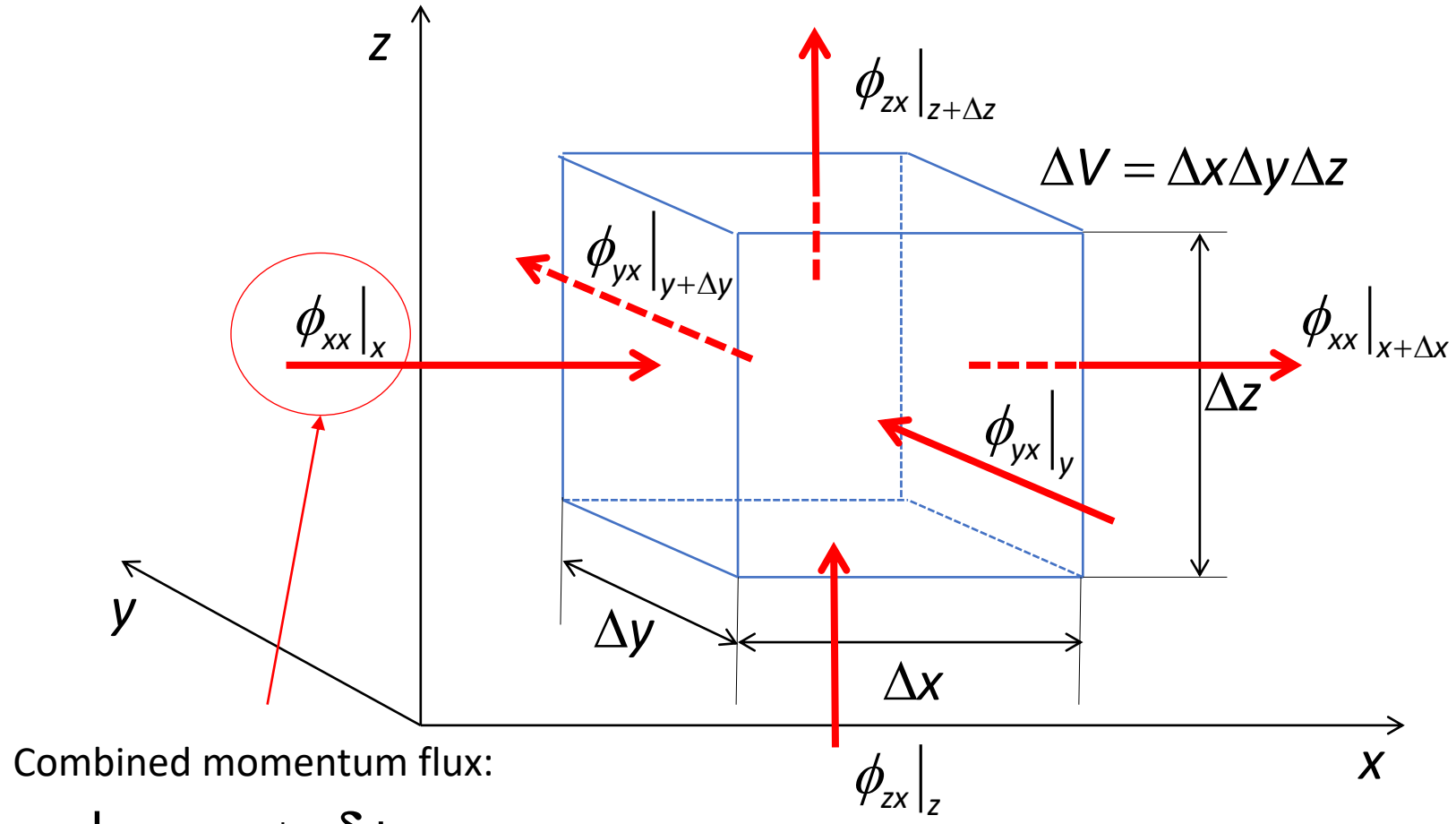

$$\rho \mathbf{vv} = \sum_i (\rho \delta_i v_i) v$$

$$\phi = \pi + \rho \mathbf{vv} = p\delta + \tau + \rho \mathbf{vv}$$

**DETAILS IN:**
**Bird-Stewart-Lightfoot (2002).**
**Transport Phenomena, 2nd edition. J. Wiley & Sons.**

# *Derivation of the equation of motion: momentum balance.*

Flux of x-momentum through the surfaces




Combined momentum flux:

$$\boldsymbol{\phi} = \rho \mathbf{v}\mathbf{v} + p\boldsymbol{\delta} + \boldsymbol{\tau}$$

# *Derivation of the equation of motion.*

IN – OUT + PROD = ACC

Terms to be included:

- ✓ Flux of momentum (x,y,z) across all the surfaces of the volume
- ✓ External forces (e.g., gravitational force)

By proceeding in analogy to what done for the continuity equation:

$$\frac{\partial}{\partial t}\rho v_x = -\left(\frac{\partial}{\partial x}\phi_{xx} + \frac{\partial}{\partial y}\phi_{yx} + \frac{\partial}{\partial z}\phi_{zx}\right) + \rho g_x$$

$$\frac{\partial}{\partial t}\rho v_y = -\left(\frac{\partial}{\partial x}\phi_{xy} + \frac{\partial}{\partial y}\phi_{yy} + \frac{\partial}{\partial z}\phi_{zy}\right) + \rho g_y$$

$$\frac{\partial}{\partial t}\rho v_z = -\left(\frac{\partial}{\partial x}\phi_{xz} + \frac{\partial}{\partial y}\phi_{yz} + \frac{\partial}{\partial z}\phi_{zz}\right) + \rho g_z$$

# *Derivation of the equation of motion.*

By using vector-tensor notation:

$$\frac{\partial}{\partial t}\rho\mathbf{v} = -\left[\nabla\cdot\boldsymbol{\phi}\right] + \rho\mathbf{g}$$

Recalling that:

$$\boldsymbol{\phi} = \rho\mathbf{vv} + p\boldsymbol{\delta} + \boldsymbol{\tau}$$

We obtain:

$$\frac{\partial}{\partial t}\rho\mathbf{v} = \quad -\left[\nabla\cdot\rho\mathbf{vv}\right] \quad -\nabla p - \left[\nabla\cdot\boldsymbol{\tau}\right] \quad + \rho\mathbf{g}$$

| Rate of increase of momentum per unit volume | Rate of momentum addition by convection per unit volume | Rate of momentum addition by molecular transport per unit volume | External force on fluid per unit volume |
|---|---|---|---|

# *Navier-Stokes equation.*

$$\frac{\partial}{\partial t}\rho\mathbf{v} = -\left[\nabla\cdot\rho\mathbf{v}\mathbf{v}\right] - \nabla p - \left[\nabla\cdot\boldsymbol{\tau}\right] + \rho\mathbf{g}$$

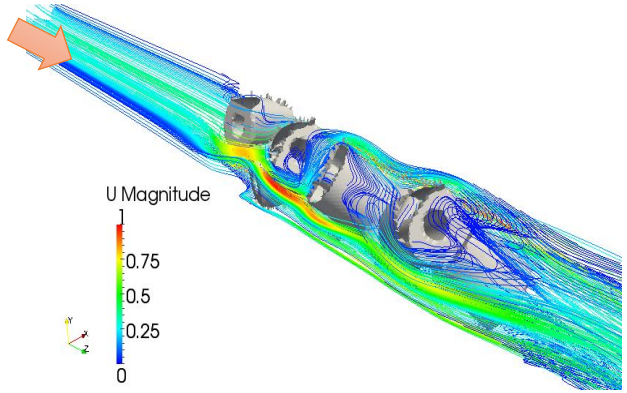
Insertion of the Newtonian expression for $\tau$ into the equation of motion leads to the Navier-Stokes equation:

$$\boldsymbol{\tau} = -\mu\left(\nabla\mathbf{v} + \left(\nabla\mathbf{v}\right)^{T}\right) + \left(\frac{2}{3}\mu - \kappa\right)(\nabla\cdot\mathbf{v})\boldsymbol{\delta}$$

$$\frac{\partial}{\partial t}\rho\mathbf{v} = -\left[\nabla\cdot\rho\mathbf{v}\mathbf{v}\right] - \nabla p + \nabla\cdot\left[\mu\left(\nabla\mathbf{v} + \nabla\mathbf{v}^{T}\right) - \left(\frac{2}{3}\mu - \kappa\right)(\nabla\cdot\mathbf{v})\boldsymbol{\delta}\right] + \rho\mathbf{g}$$

# *Solution of the equation of motion.*

- The equations of fluid mechanics are solvable (analytical solution) only for a limited number of conditions.
- Numerical solution of the equations is necessary
- This field is known as computational fluid dynamics (CFD)



# *What is CFD?*

- To obtain a numerical solution, the differential equations are approximated by using a discretization method
- The discretization leads to a system of algebraic equations, which can be solved on a computer. Convergence is an issue!
- The discretization is applied to small domains in space, thus the solution provides results at discrete locations in space and time
- Typically, in solving the discretized equations, iterative methods are used. Convergence is an issue!

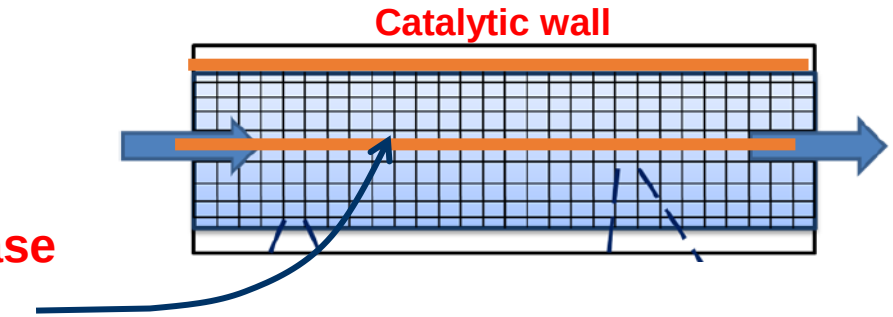
# *Main issues.*

- ✓ Numerical solutions are always approximate
- ✓ In some conditions, even if we solve the equations exactly, the solution may not be correct when «models» are used to describe particular conditions (e.g., turbulence, see later)
- ✓ Compromises are always necessary (grid Vs. discretization schemes, …)

*«Wonderful color pictures make great impressions but are of no value if they are not quantitatively correct. Results must be examined very critically»*
(Ferziger & Peric)

***Computational Fluid Dynamics* Vs Colorful Fluid Dynamics**

# *Reacting flows: governing equations.*




$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0$$ continuity

$$\frac{\partial}{\partial t}(\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) = -\nabla p + \nabla \cdot \left[ \mu \left( \nabla \mathbf{v} + \nabla \mathbf{v}^T \right) - \frac{2}{3} \mu (\nabla \cdot \mathbf{v}) \delta \right] + \rho \mathbf{g}$$ momentum

$$\frac{\partial}{\partial t}(\rho \omega_k) + \nabla \cdot (\rho \omega_k \mathbf{v}) = -\nabla \cdot (\rho \omega_k \mathbf{V}_k) + \dot{\Omega}_k^{\text{hom}} \qquad k = 1,\ldots,NG$$ gas-phase species

$$\rho \hat{C}_P \frac{\partial T}{\partial t} + \rho \hat{C}_P \mathbf{v} \nabla T = \nabla \cdot (\lambda \nabla T) - \rho \sum_{k=1}^{NG} \hat{C}_{P,k} \omega_k \mathbf{V}_k - \sum_{k=1}^{NG} \hat{H}_k^{\text{hom}} \dot{\Omega}_k^{\text{hom}}$$ gas-phase energy

# Governing equations

## Non-catalytic walls

$$\nabla \omega_k \big|_{inert} = 0$$

$$T\big|_{inert} = f\left(t, T\right)$$

$$\nabla T\big|_{inert} = g\left(t, T\right)$$

## Catalytic walls

$$\rho \Gamma_{k,mix} \left(\nabla \omega_k\right)\big|_{catalytic} = -\alpha_{cat}\, \dot{\Omega}_k^{het} \qquad k = 1,\ldots,NG$$

$$\lambda \left(\nabla T\right)\big|_{catalytic} = -\alpha_{cat} \sum_{j=1}^{NR} \Delta H_j^{het} \dot{r}_j^{het}$$

$$\sigma_{cat} \frac{\partial \theta_i}{\partial t} = \dot{\Omega}_i^{het} \qquad i = 1,\ldots,NS$$

**Adsorbed (surface) species**

## Detailed microkinetic models

$COOH^* + * \rightarrow CO^* + OH^*$

$CO^* + OH^* \rightarrow COOH^* + *$

$COOH^* + * \rightarrow CO_2^* + H^*$

$CO_2^* + H^* \rightarrow COOH^* + *$

$CO_2^* + H_2O^* \rightarrow COOH^* + OH^*$

$COOH^* + OH^* \rightarrow CO_2^* + H_2O^*$

$CO_2^* + H^* \rightarrow HCOO^{**}$

$HCOO^{**} \rightarrow CO_2^* + H^*$

$CO_2^* + OH^* + * \rightarrow HCOO^{**} + O^*$

$HCOO^{**} + OH^* \rightarrow CO_2^* + H_2O^*$

$CH^* + H^* \rightarrow CH_2^* + *$

$CH^* + * \rightarrow C^* + H^*$

$C^* + H^* \rightarrow CH^* + *$

$CH_3^* + O^* \rightarrow CH_2^* + OH^*$

$CH_2^* + OH^* \rightarrow CH_3^* + O^*$

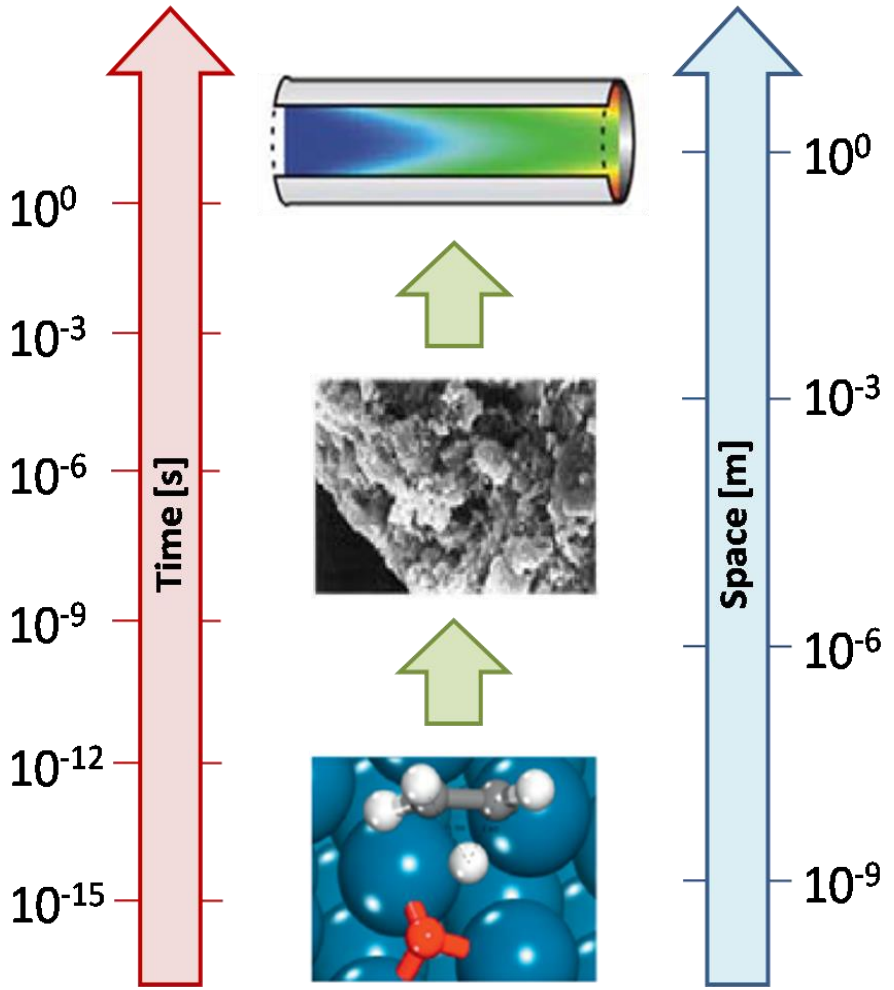
$CH^* + OH^* \rightarrow CH_2^* + O^*$

$CH_2^* + O^* \rightarrow CH^* + OH^*$

…

# *Numerical challenges for reacting flows*

- ✓ **Dimensions of the system**
  - Proportional to the number of species
  - Proportional to the number of cells
- ✓ **Stiffness**
  - Different temporal scales involved
  - Different spatial scales involved
- ✓ **Non-linearity**
  - Source term non-linear in concentrations and temperature
  - Coverage dependence of activation energy

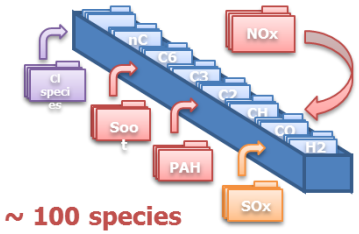


# *Numerical solution*

**Fully segregated algorithms**

☺ easy to implement and computationally efficient
☹ **unfeasible when large, stiff kinetic mechanisms are used**

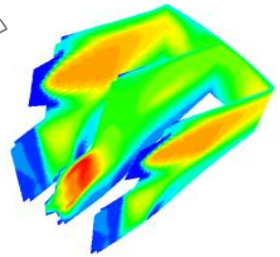
**Detailed kinetic schemes**




**~ 100 species**
**~ 1000 reactions**

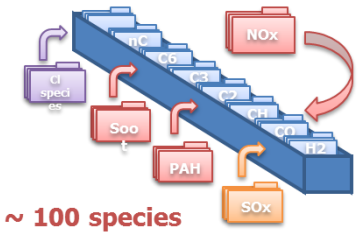
Strong non linearity of reaction terms
High stiffness

**Complex geometries**



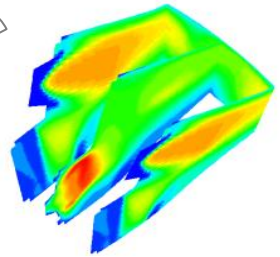
# *Numerical solution*

**Detailed kinetic schemes**




**~ 100 species**
**~ 1000 reactions**

**Complex geometries**



Fully segregated algorithms

☺ easy to implement and computationally efficient
☹ unfeasible when large, stiff kinetic mechanisms are used

Strong non linearity of reaction terms
High stiffness

Fully coupled algorithms

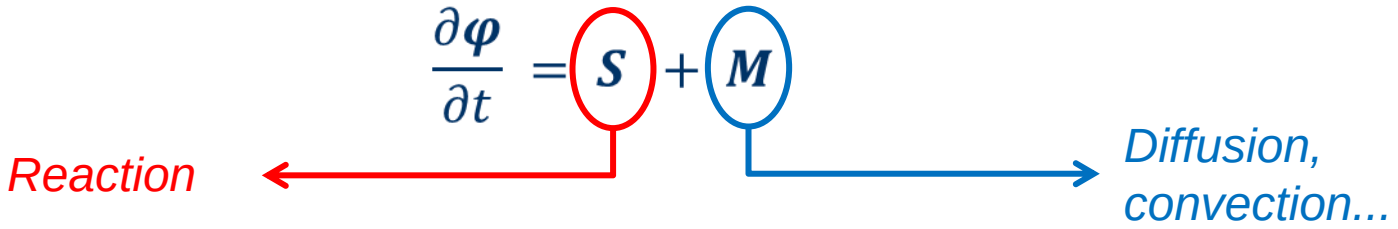
☺ **all the processes and their interactions are considered simultaneously**
☺ natural way to treat problems with multiple stiff processes
☹ the resulting system of equations can be extremely large and the computational cost prohibitive

Operator-splitting methods

☺ usually avoid many costly matrix operations
☺ **allow the best numerical method to be used for each type of term or process**
☹ the resulting algorithms can be very complex and usually differ from term to term

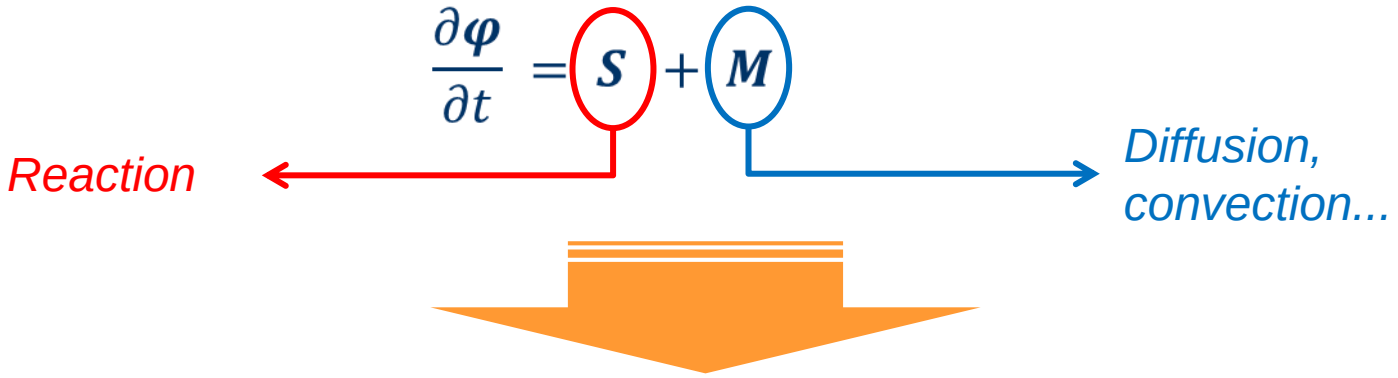
CatalyticFOAM

# Operator splitting algorithm




*Second-order splitting schemes for a class of reactive systems – Z. Ren, S. B. Pope – Journal of Computational Physics - 2008*

# Operator splitting algorithm




Splitting scheme

*Second-order splitting schemes for a class of reactive systems – Z. Ren, S. B. Pope – Journal of Computational Physics - 2008*

# Operator splitting algorithm

$$\frac{\partial \varphi}{\partial t} = S + M$$

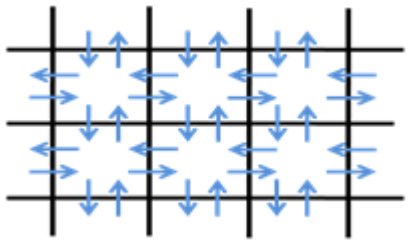
*Reaction*

*Diffusion, convection...*

Splitting scheme

**PREDICTOR**

$$\frac{\partial \varphi}{\partial t} = M$$



*Second-order splitting schemes for a class of reactive systems – Z. Ren, S. B. Pope – Journal of Computational Physics - 2008*

# Operator splitting algorithm

$$\frac{\partial \varphi}{\partial t} = S + M$$

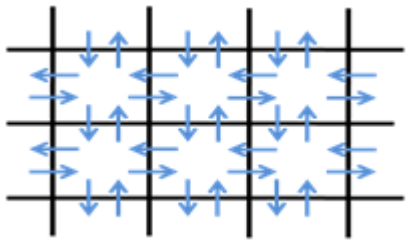
*Reaction*

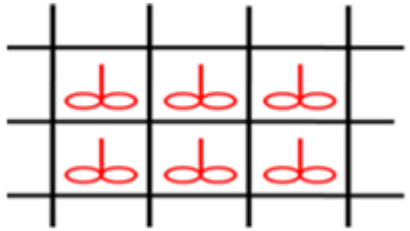
*Diffusion, convection...*

Splitting scheme

**PREDICTOR**

$$\frac{\partial \varphi}{\partial t} = M$$





$$\frac{\partial \varphi}{\partial t} = S$$

**CORRECTOR**

*Second-order splitting schemes for a class of reactive systems – Z. Ren, S. B. Pope – Journal of Computational Physics - 2008*

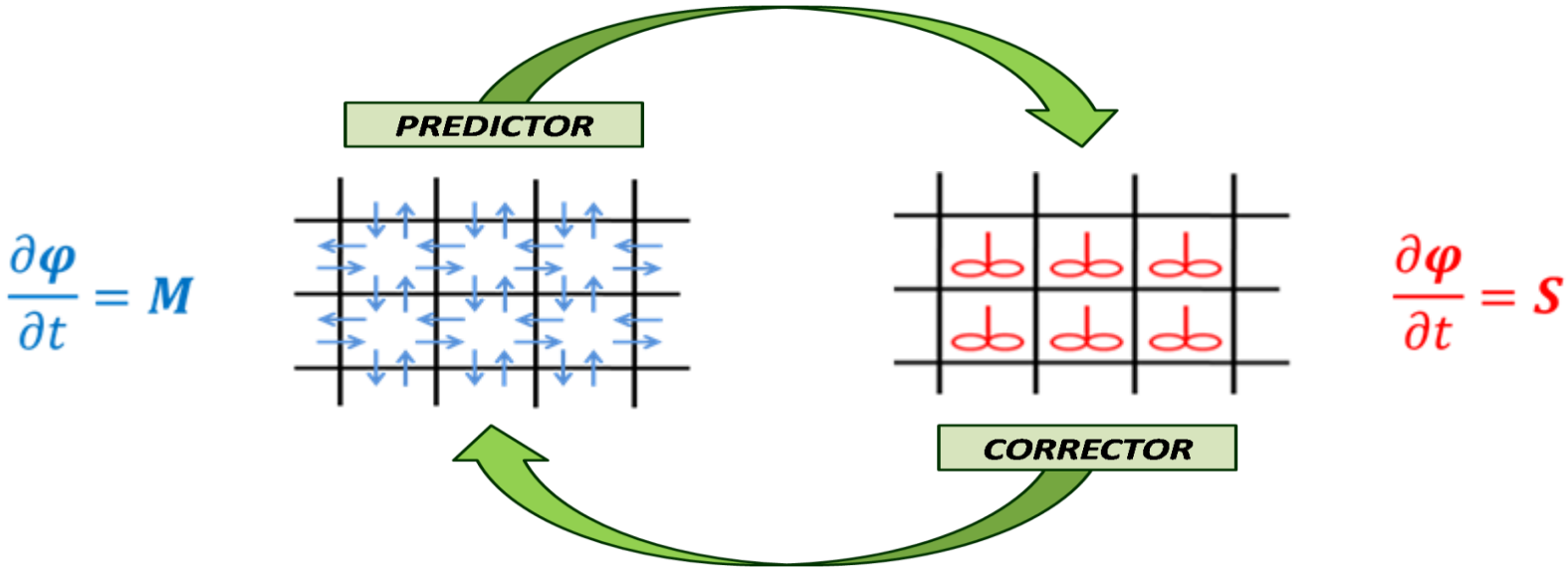
# Operator splitting algorithm

$$\frac{\partial \varphi}{\partial t} = S + M$$

*Reaction*

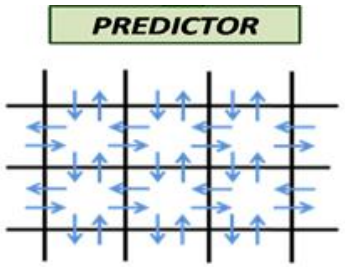
*Diffusion, convection...*

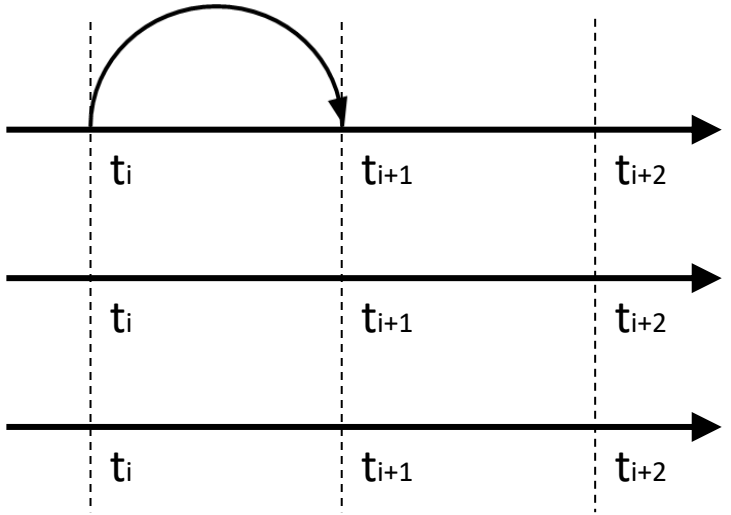
Splitting scheme




*Second-order splitting schemes for a class of reactive systems – Z. Ren, S. B. Pope – Journal of Computational Physics - 2008*

# *Operator splitting algorithm*

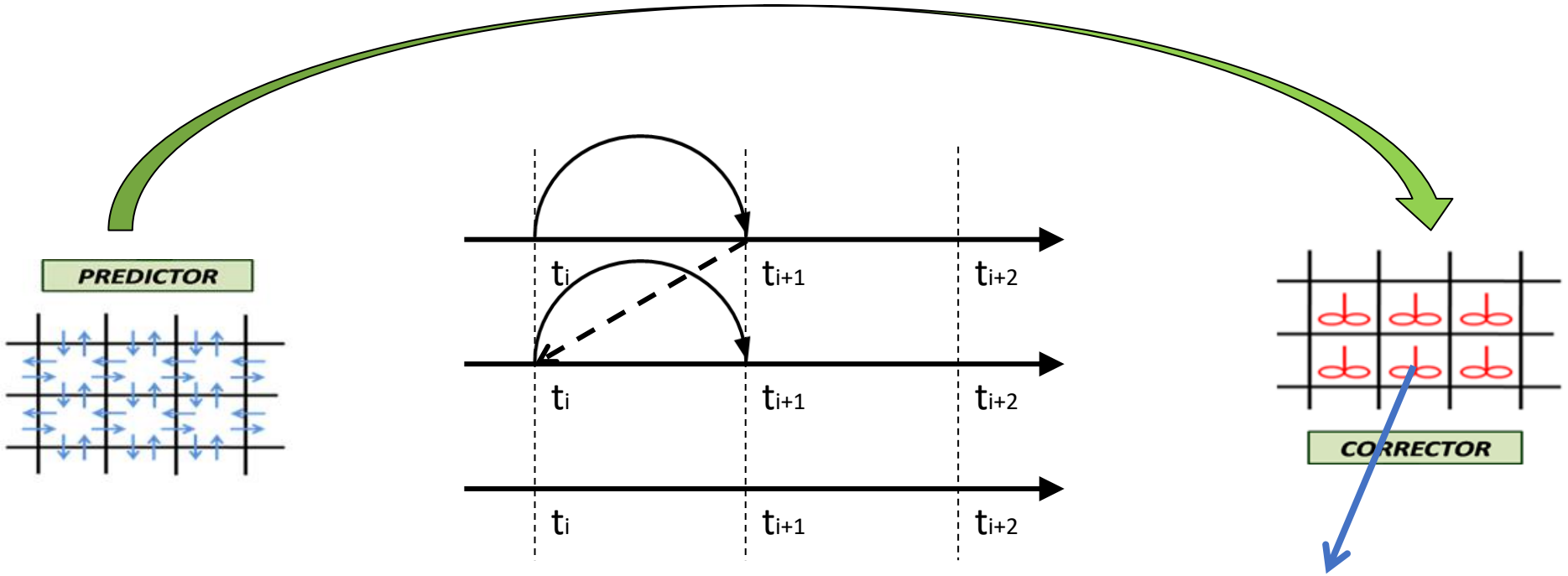



The integration of the **transport term** is performed over the *i*th time step

*Second-order splitting schemes for a class of reactive systems – Z. Ren, S. B. Pope – Journal of Computational Physics - 2008*

# *Operator splitting algorithm*

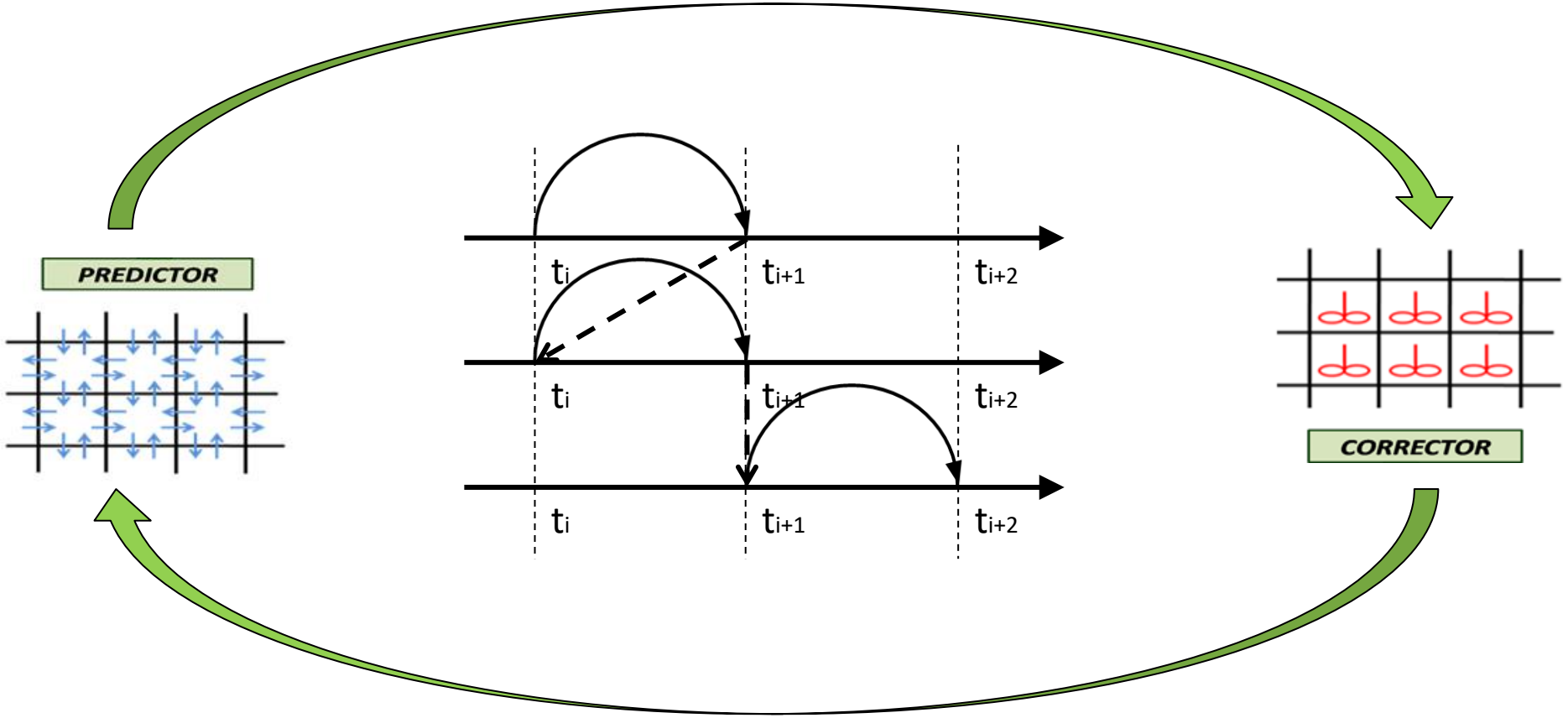



**Each cell of the computational domain is treated as a batch reactor**

The integration of the **reaction term** is performed over the **same time step**

*Second-order splitting schemes for a class of reactive systems – Z. Ren, S. B. Pope – Journal of Computational Physics - 2008*

# *Operator splitting algorithm*

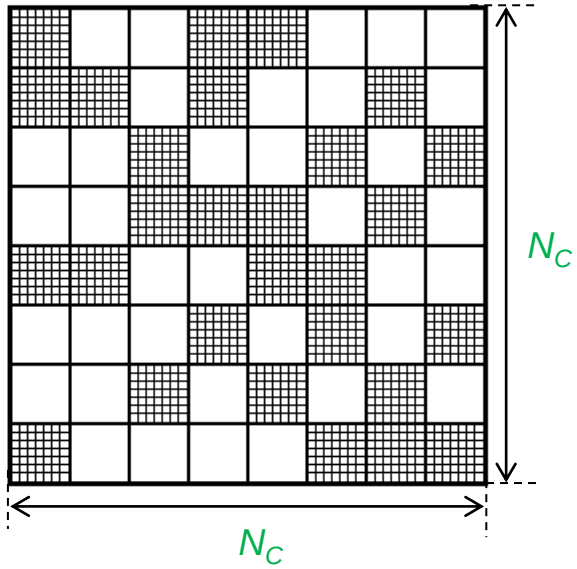



The procedure is iterated on the next time step

*Second-order splitting schemes for a class of reactive systems – Z. Ren, S. B. Pope – Journal of Computational Physics - 2008*

# *Jacobian matrix*

**Global system**



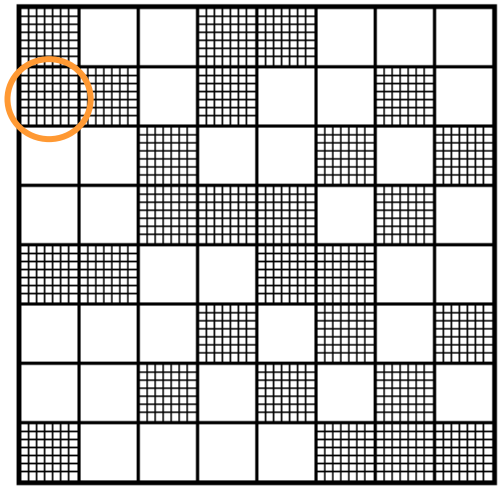

$$\frac{\partial \varphi}{\partial t} = \boldsymbol{M} + \boldsymbol{S}$$

**Jacobian matrix:**

- ✓ **Sparse**
- ✓ **Unstructured**
- ✓ **Blocks**

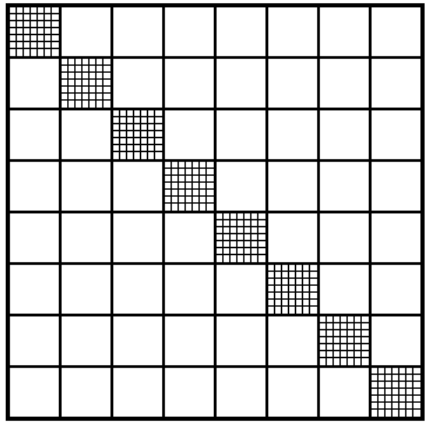
# *Jacobian matrix*

**Global system**



$$\frac{\partial \varphi}{\partial t} = M + S$$

Source term



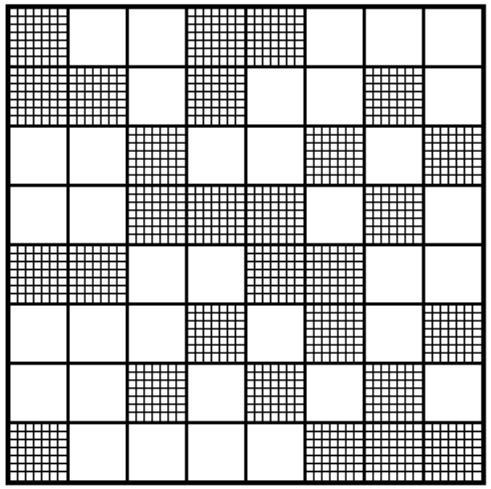
$$\frac{\partial \varphi}{\partial t} = S$$

**Jacobian matrix:**

- ✓ **Sparse**
- ✓ **Diagonal**
- ✓ **Blocks**

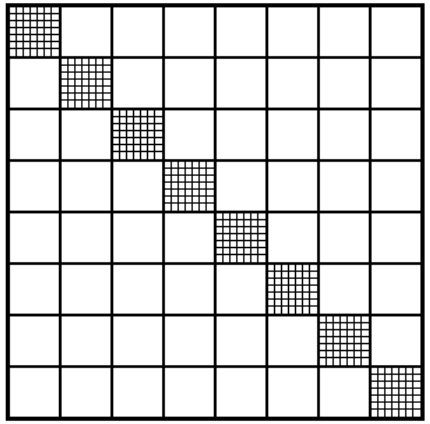
# *Jacobian matrix*

Global system



$$\frac{\partial \varphi}{\partial t} = M + S$$

Source term

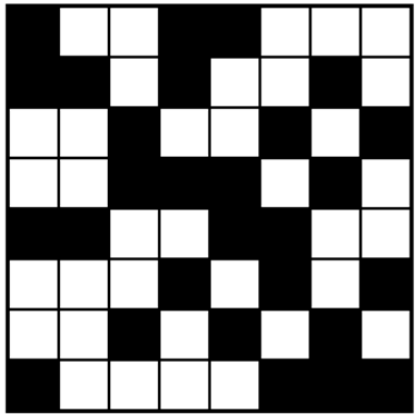


$$\frac{\partial \varphi}{\partial t} = S$$

**Jacobian matrix:**

- ✓ **Sparse**
- ✓ **Unstructured**

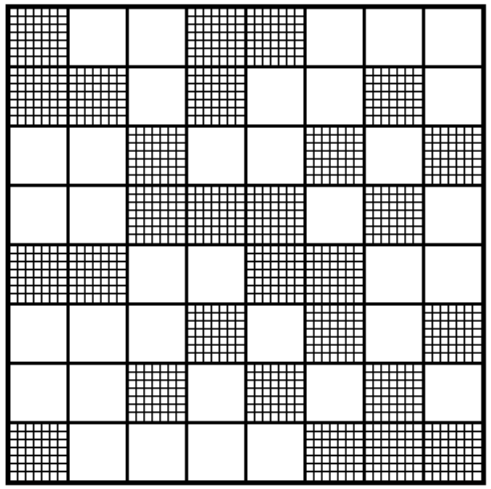
**Transport term**



$$\frac{\partial \varphi}{\partial t} = M$$

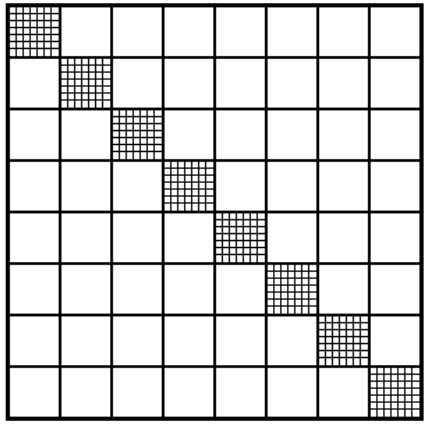
# Operator splitting algorithm

Global system



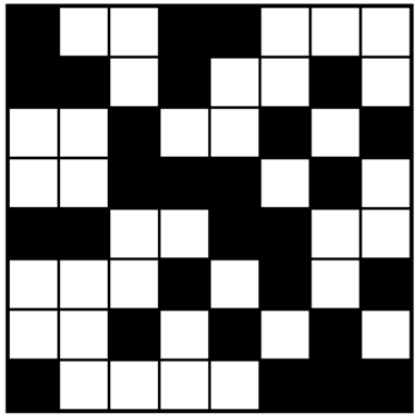
$$\frac{\partial \varphi}{\partial t} = M + S$$

Source term



$$\frac{\partial \varphi}{\partial t} = S$$

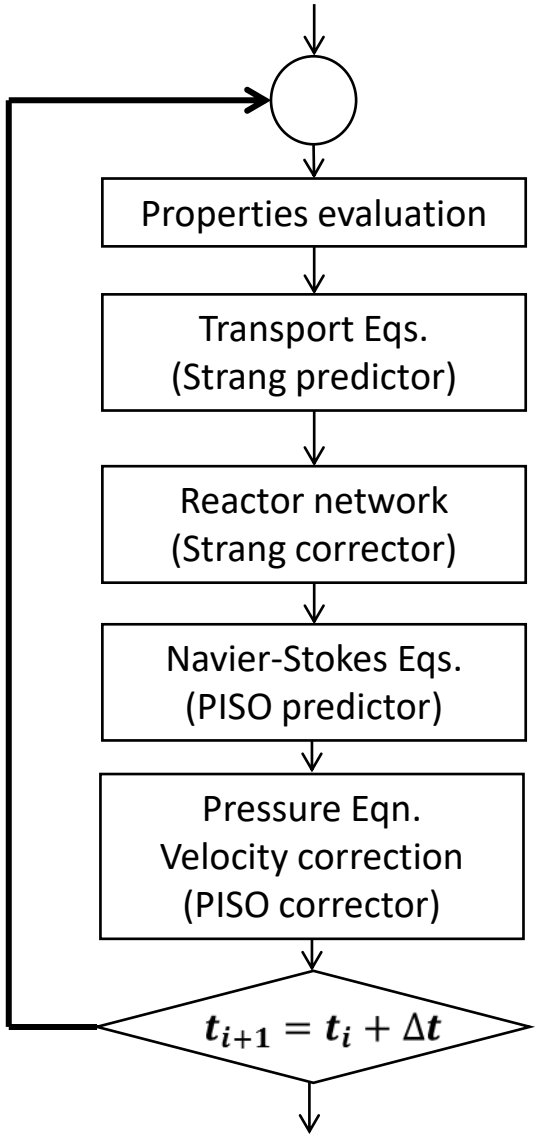
Transport term

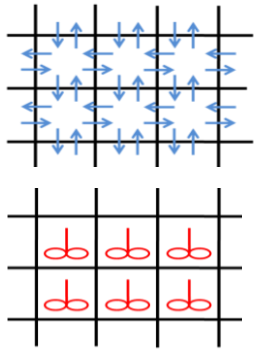


$$\frac{\partial \varphi}{\partial t} = M$$

**Use of a suitable algorithm for each sub-problem**

# *Solution procedure*



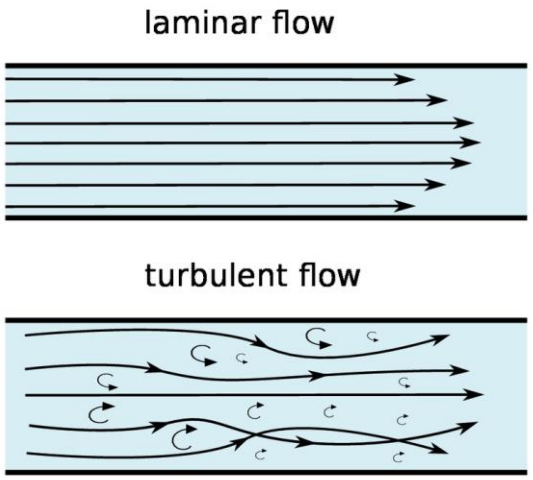
Main features:

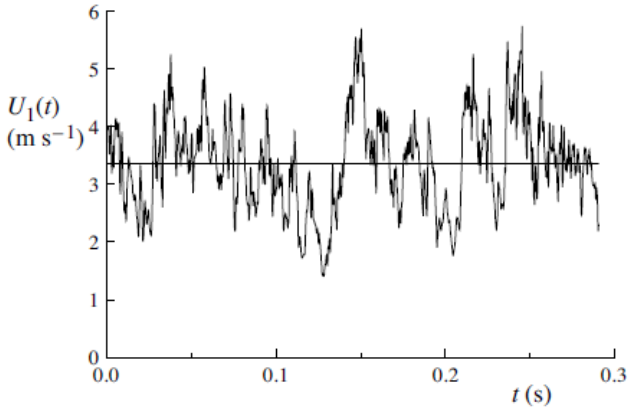
- ✓ Solution of the Navier-Stokes equations (laminar and turbulent regime)
- ✓ No limit to the number of species and reactions
- ✓ No limit in geometry

# *Fluid regimes.*

Two fluid regimes:

- **Laminar flow**
  - Fluid particles move along well-defined stream-lines
- **Turbulent flow**
  - Stream-lines cross and mix
  - Highly unsteady with local pressure and velocity variations in space and **time**
  - Time averaged quantities have a steady value

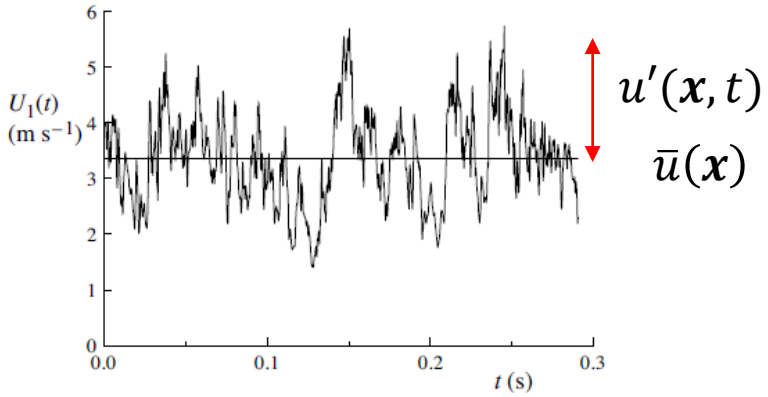


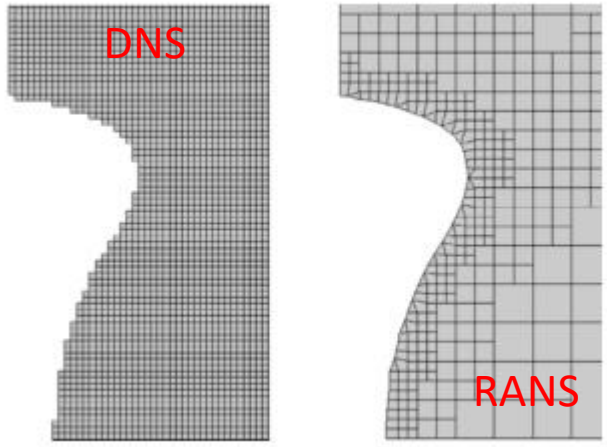

Pope, Turbulent flows, Cambridge University Press (2000)

# *Numerical simulations of turbulent flows.*

- **DNS**
  - Only continuum hypothesis
  - No modelling is involved
  - Highest cost, highest reliability
  - Unfeasible computational cost
- **RANS**
  - Solution of time-averaged NS
    $u(\boldsymbol{x},t) = \bar{u}(\boldsymbol{x}) + u'(\boldsymbol{x},t)$
  - closure models (k-ε,k-ωSST …)
  - Feasible computational cost
- **LES**
  - Based on scale separations
  - Large structures are computed, small scales are modeled
  - Relatively high computational costs

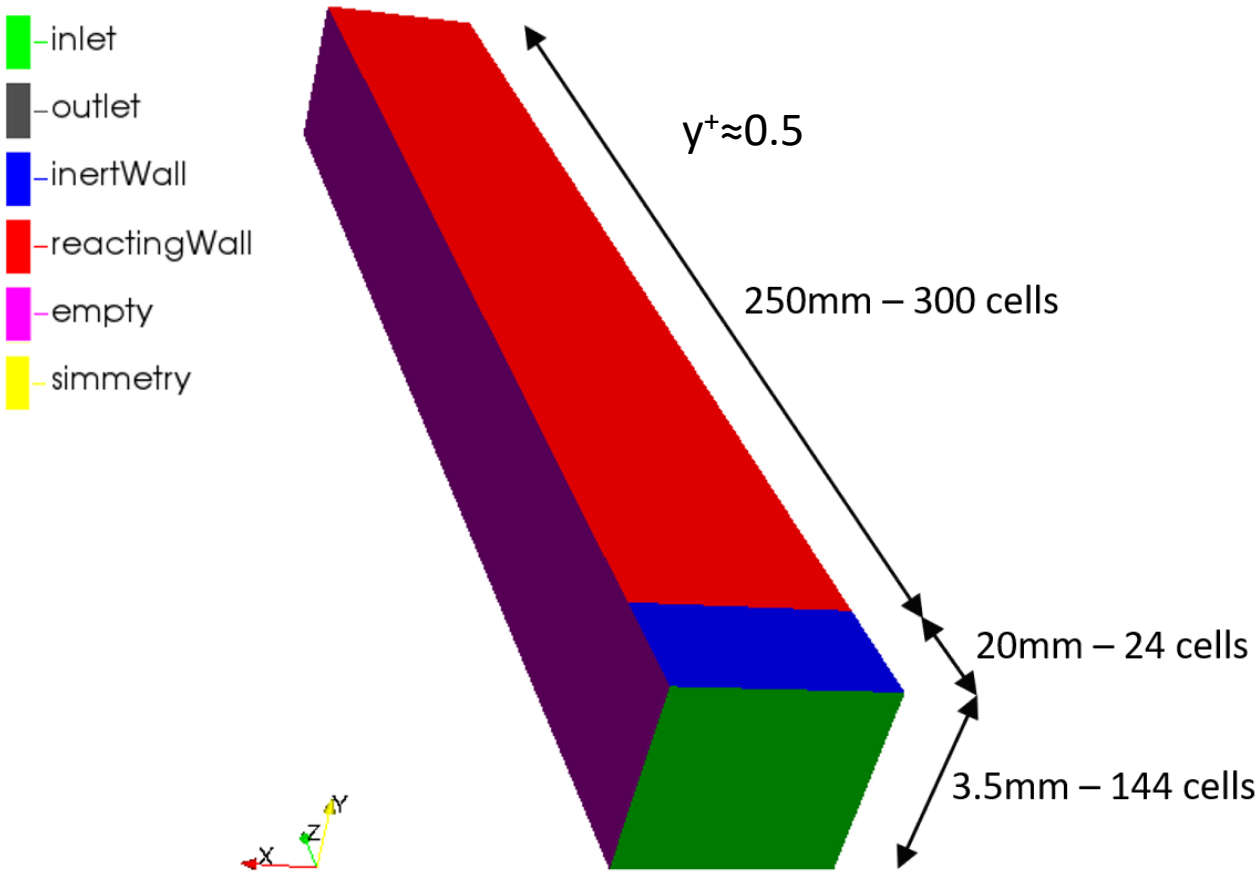



Pope, Turbulent flows, Cambridge University Press (2000)




A. Della Torre et al., Int. J. Heat and Fluid Flow 50 (2014)

# *Turbulence models: assessment.*




The **goal** is the assessment of the implementation and coupling of the **turbulence models** of OpenFOAM with the new solver

| CASE | $\phi$ | *Re* |
|---|---|---|
| 1 | 0.18 | 15000 |
| 2 | 0.24 | 30000 |

$H_2$+0.5$O_2$-> $H_2O$

$T_{in}$=673.15K

RANS with κ-ε AKN

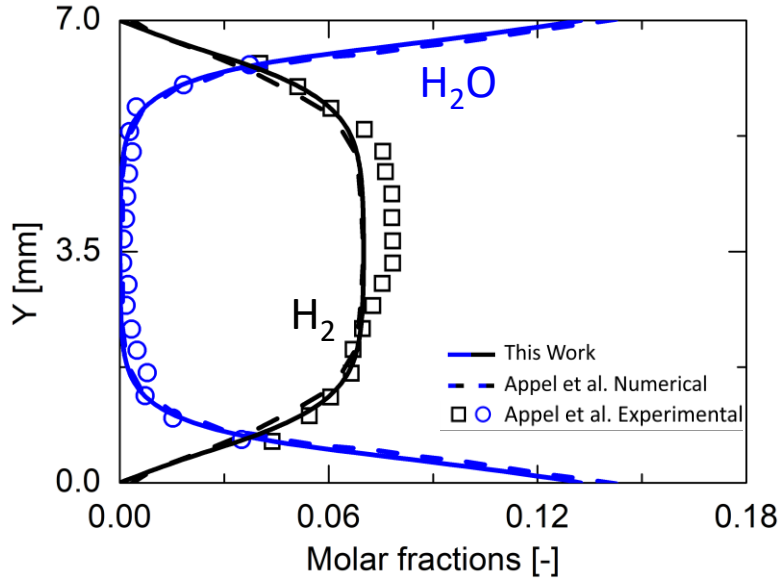
$$\phi = \frac{\left({}^{H_2}/_{O_2}\right)}{\left({}^{H_2}/_{O_2}\right)_{stoichio}}$$

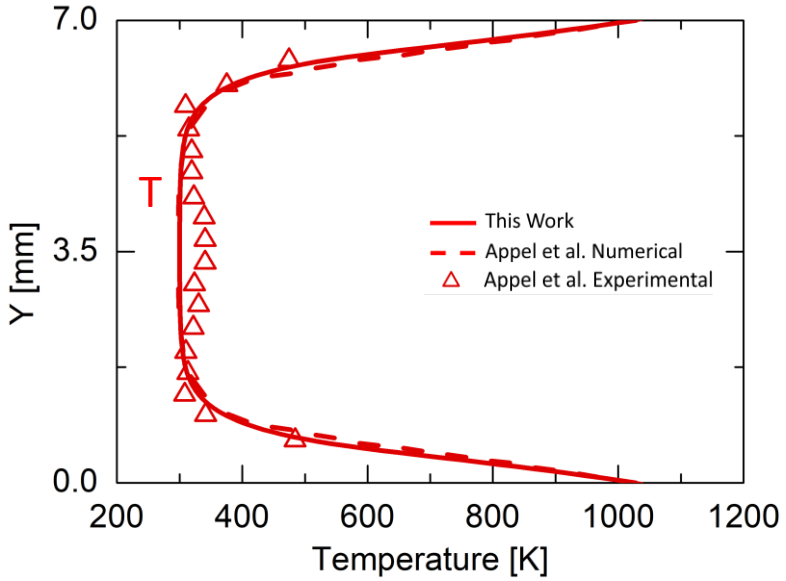
Numerical schemes:

- Divergence terms: Gauss limitedLinear
- Laplacian terms: Gauss linear orthogonal
- Gradient terms: Gauss linear

C. Appel et al., Proceedings of the Combustion Institute, vol. 29, n. 1 ( 2002)

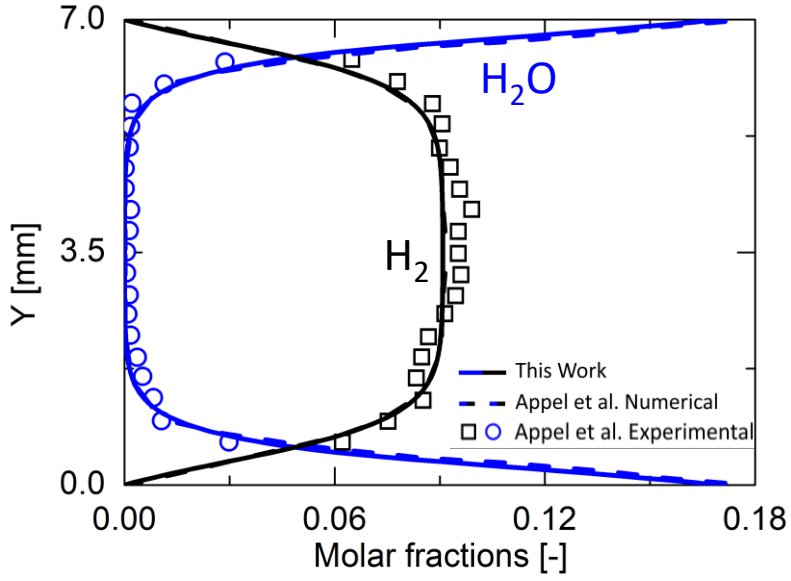
# *Turbulence models: assessment.*

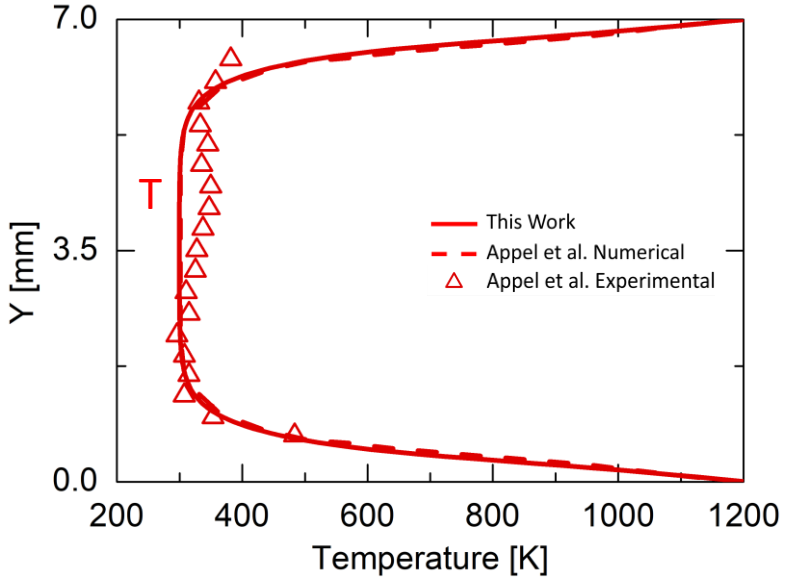



CASE 1 { $Re$ = 15000; $\phi$ = 0.18

Good agreement with experimental data

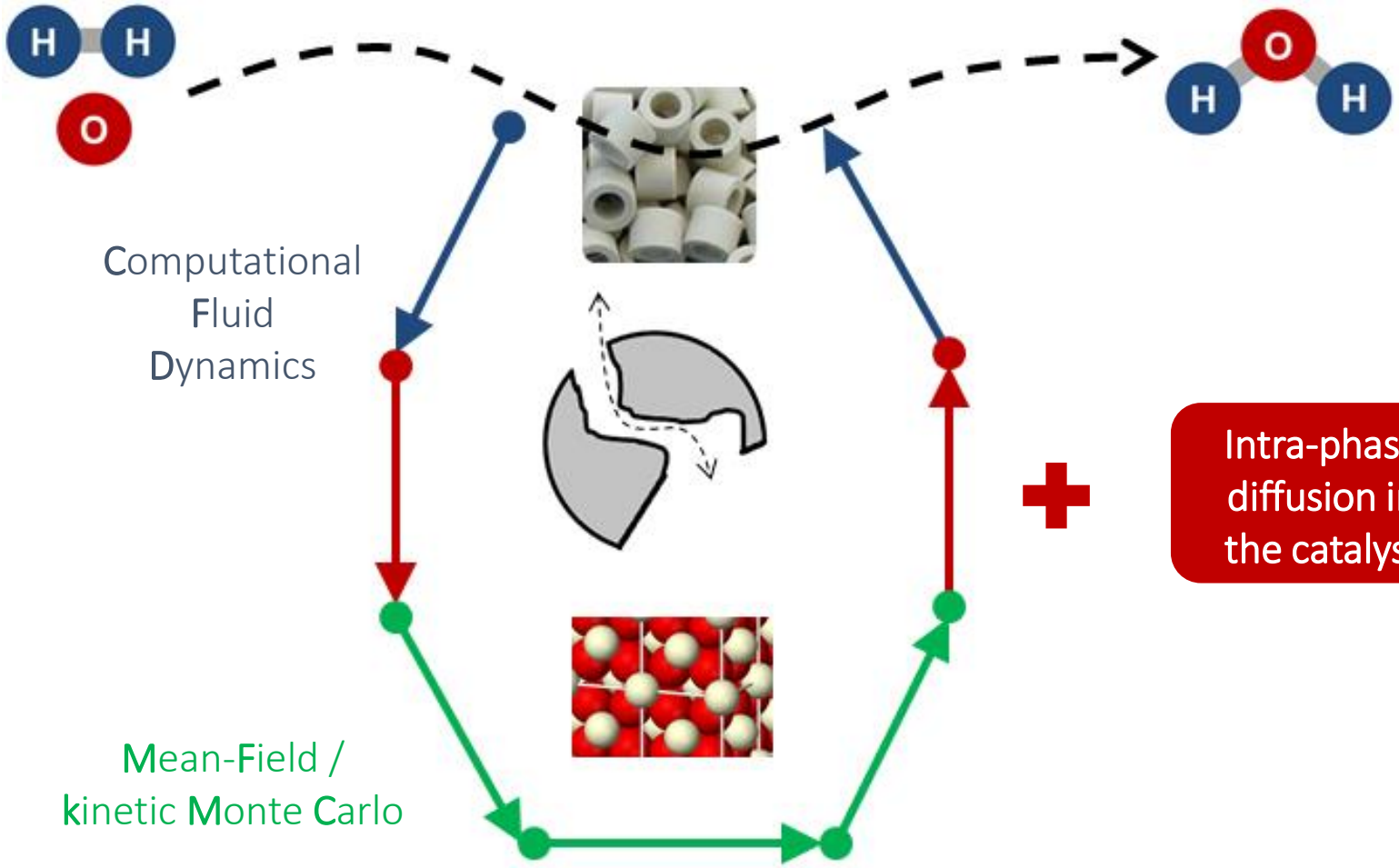



CASE 2 { $Re$ = 30000; $\phi$ = 0.24

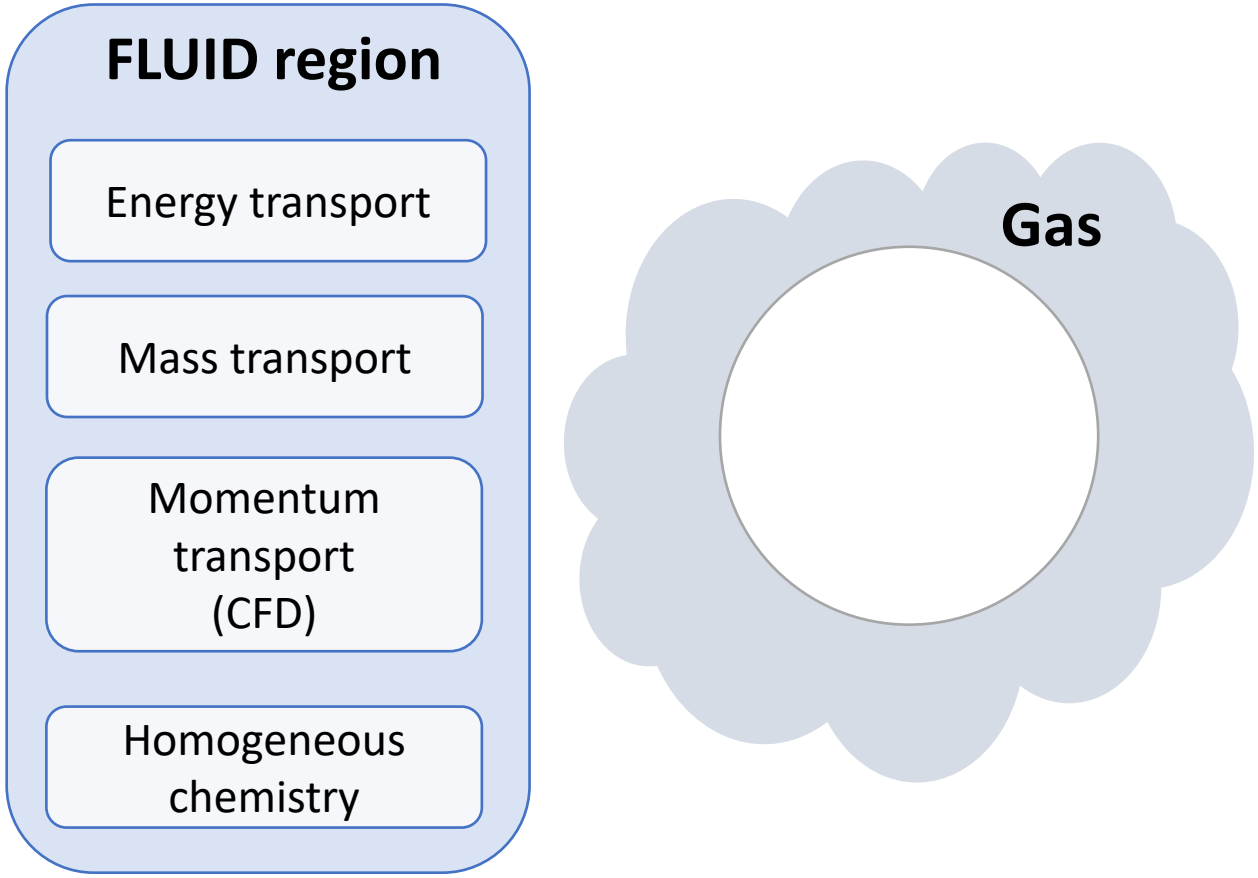
C. Appel et al., Proceedings of the Combustion Institute, 29 (2002)

# *The long way to the active site*

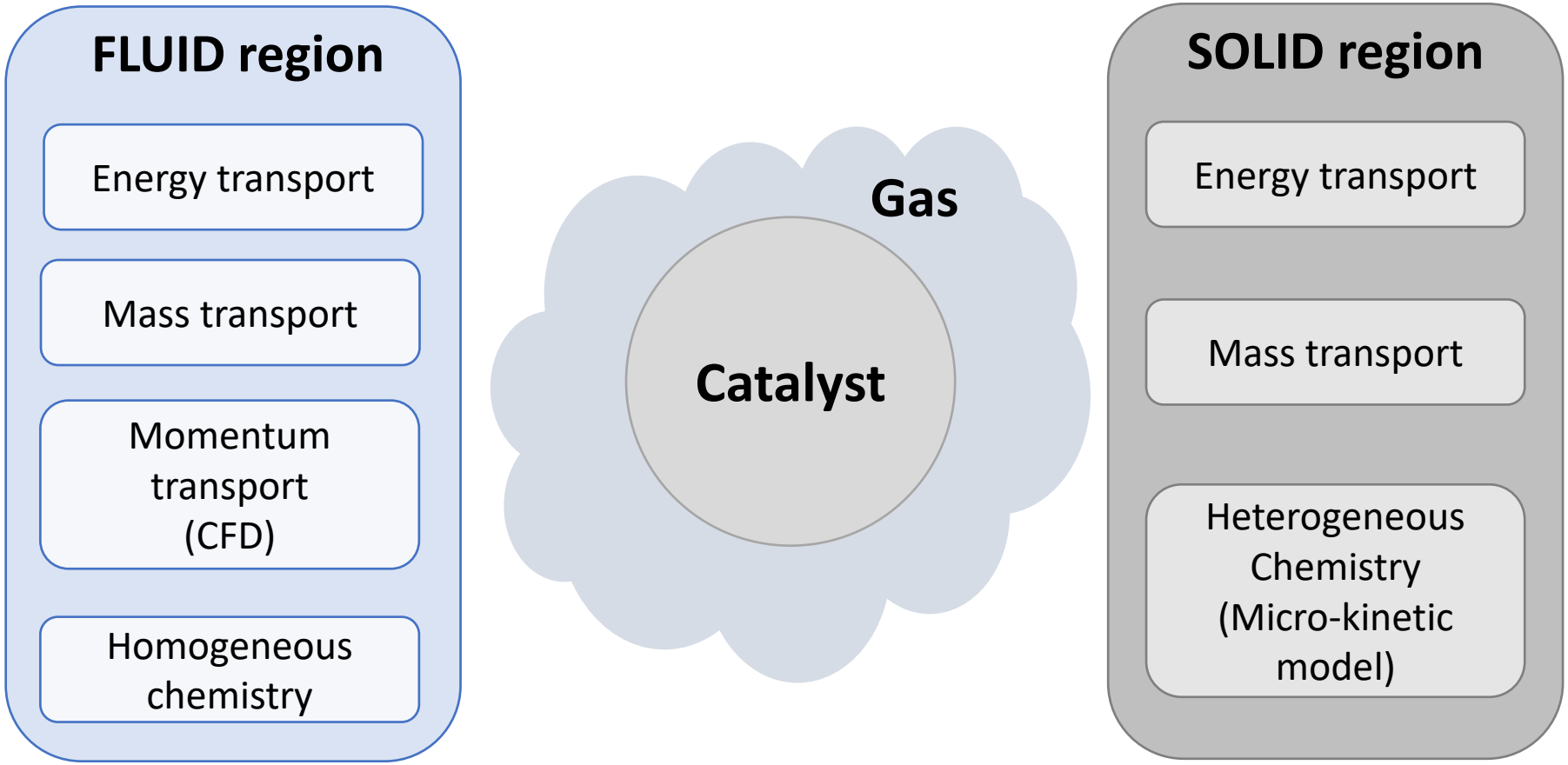



T. Maffei, G. Gentile et al., *Chem. Eng. J., 2016*

# *Coupling CFD with intraphase transport*

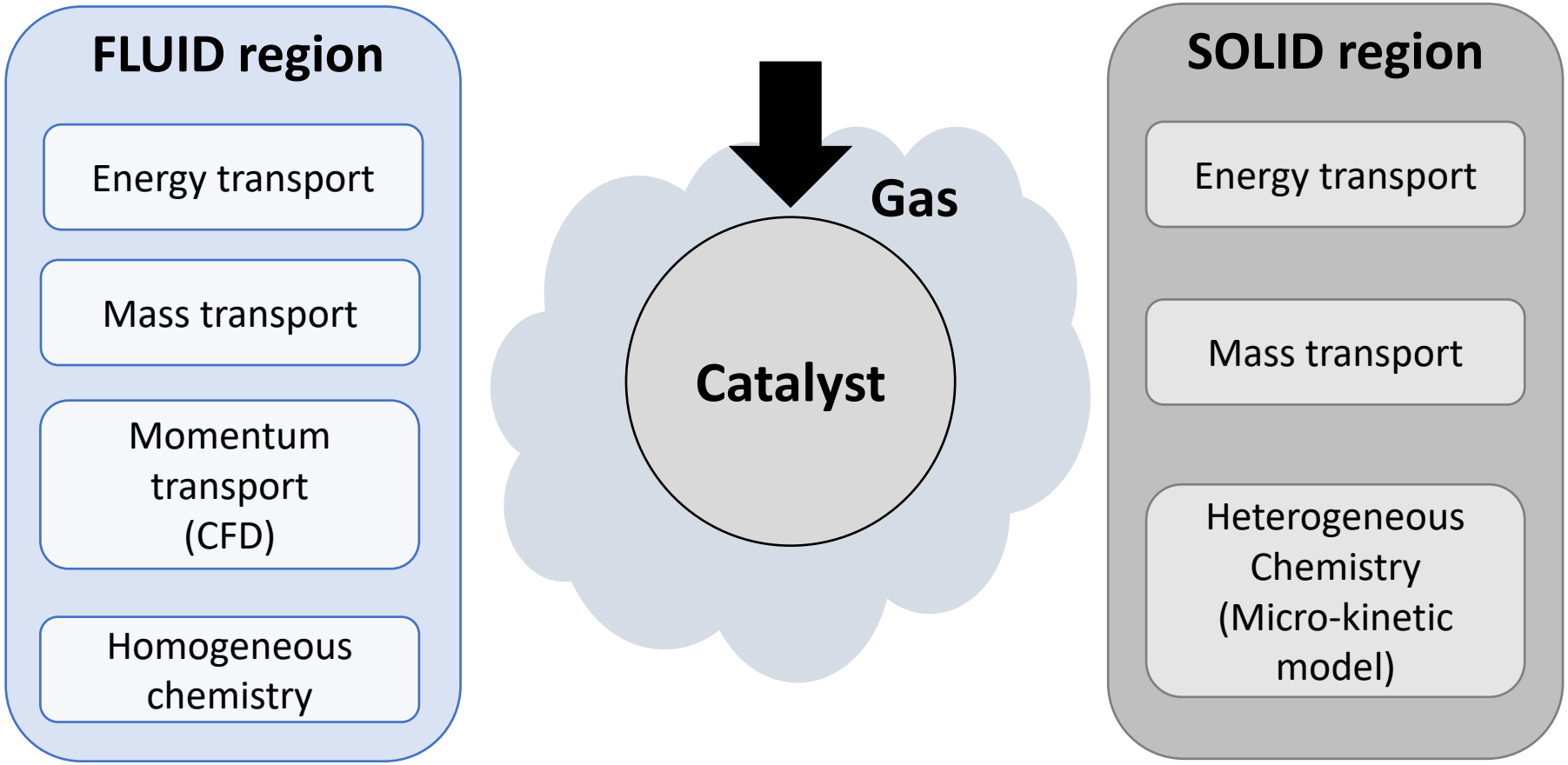


# *Coupling CFD with intraphase transport*




**Two disctinct regions (meshes) with their own equations**

# *Regions connected by the solid-fluid interface*



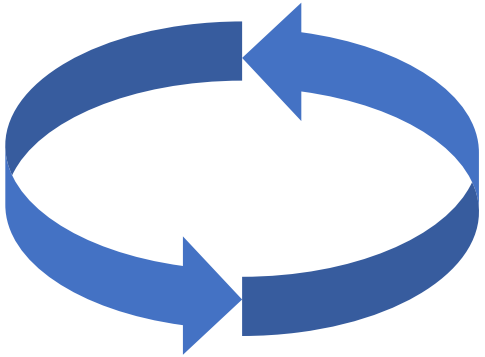

**Two disctinct regions (meshes) with their own equations**

# *Multi-region approach.*

**FLUID region**

- Energy transport
- Mass transport
- Momentum transport (CFD)
- Homogeneous chemistry

**Coupling between regions**

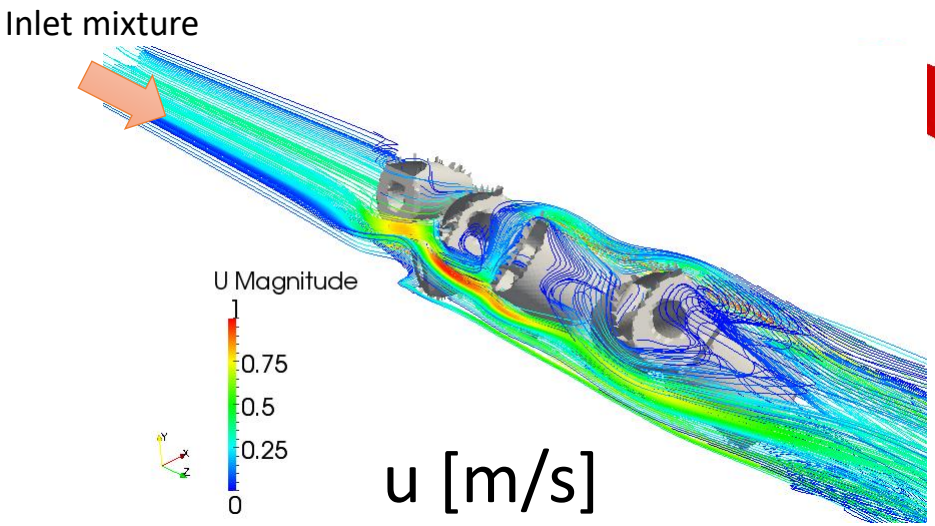


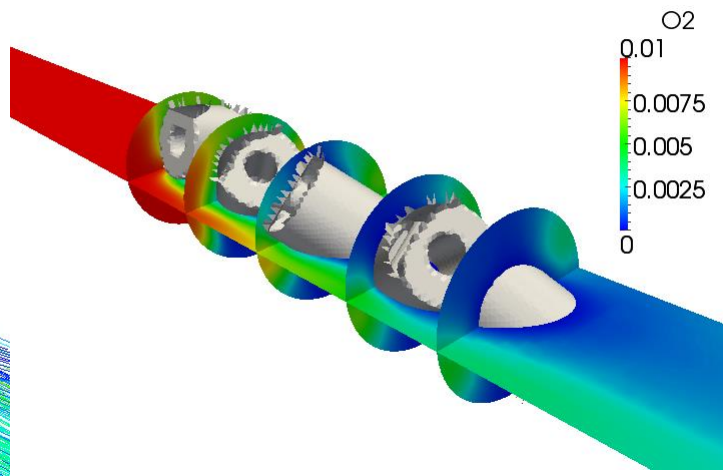
**Effective coupling between the two regions is pivotal**

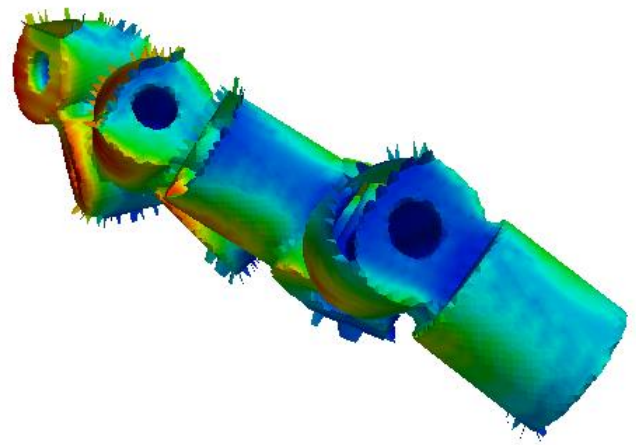
**SOLID region**

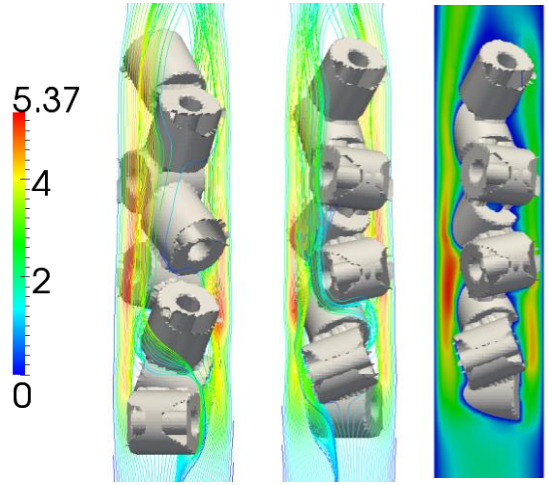
- Energy transport
- Mass transport
- Heterogeneous Chemistry (Micro-kinetic model)

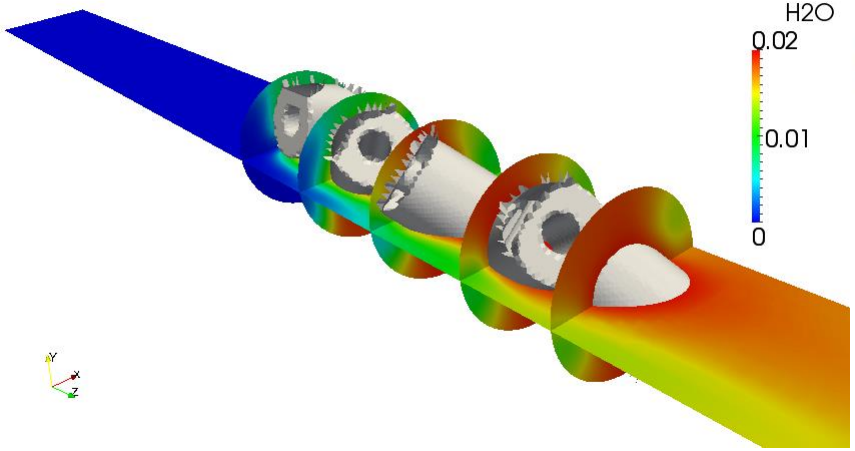
# *Show-case: Rashig rings.*

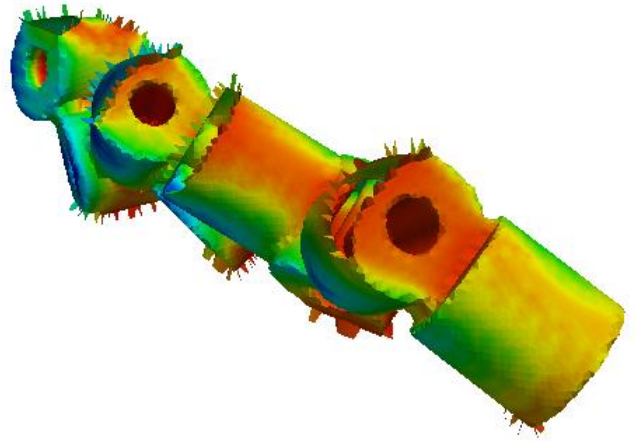


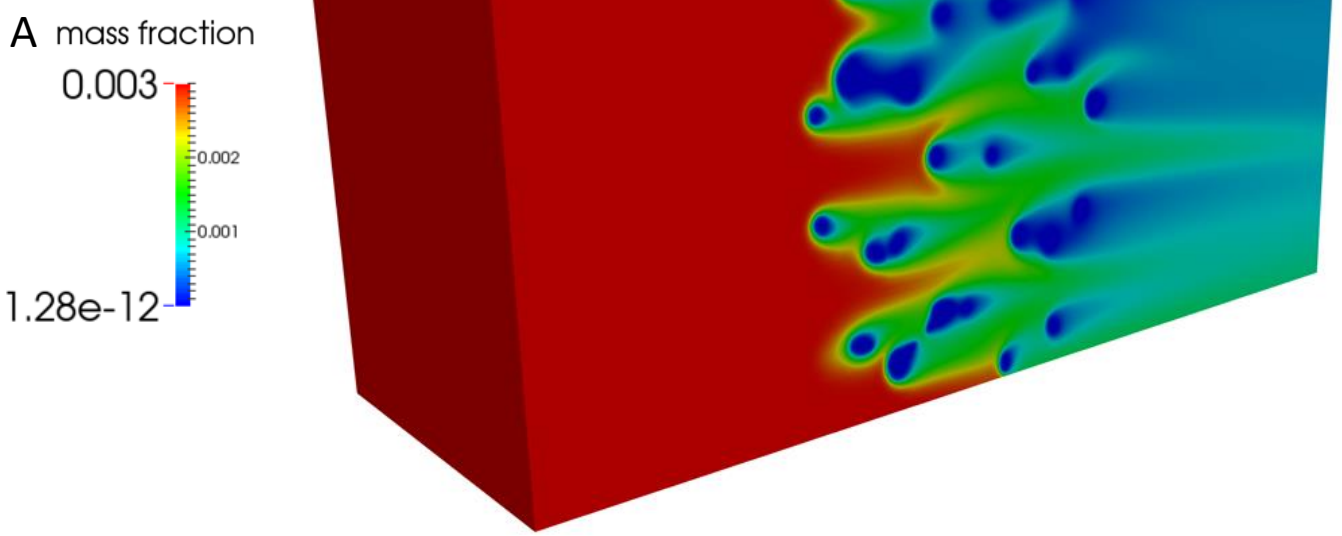



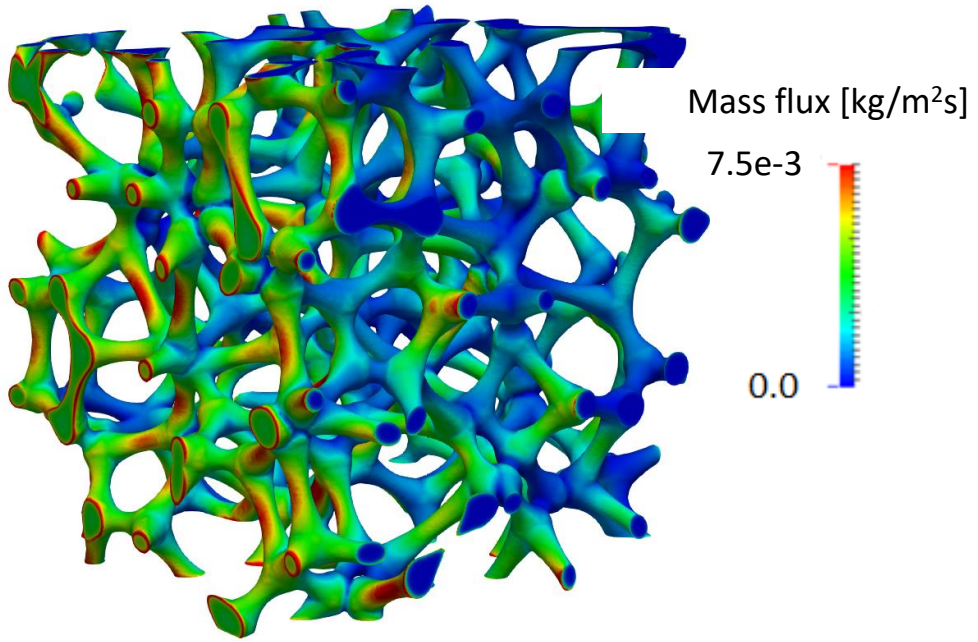




M. Maestri and A. Cuoci, *Chem. Eng. Sci., 2013*

# *Show-case: foams.*



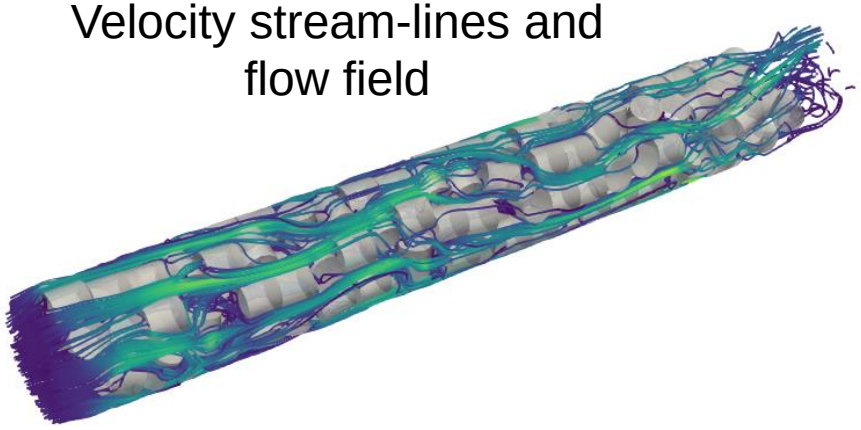



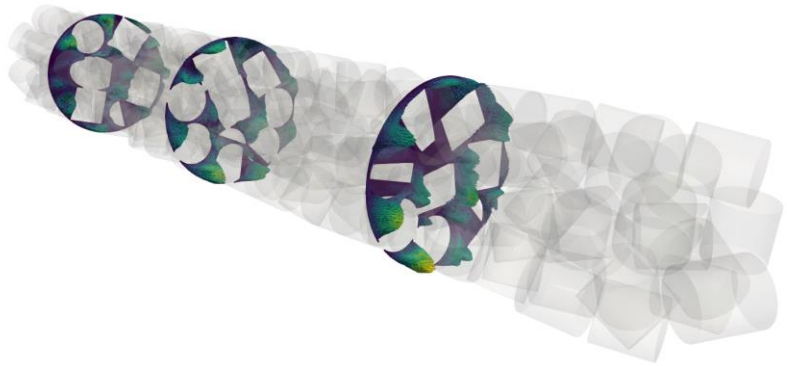

M. Bracconi, M. Ambrosetti, M. Maestri, G. Groppi, E. Tronconi, *Chem. Eng. J., 2017*

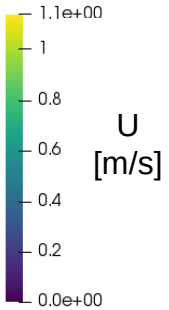
M. Bracconi, M. Ambrosetti, M. Maestri, G. Groppi, E. Tronconi, *Chem. Eng. J., 2018*

# *Show-case: cylinders - methanol synthesis*

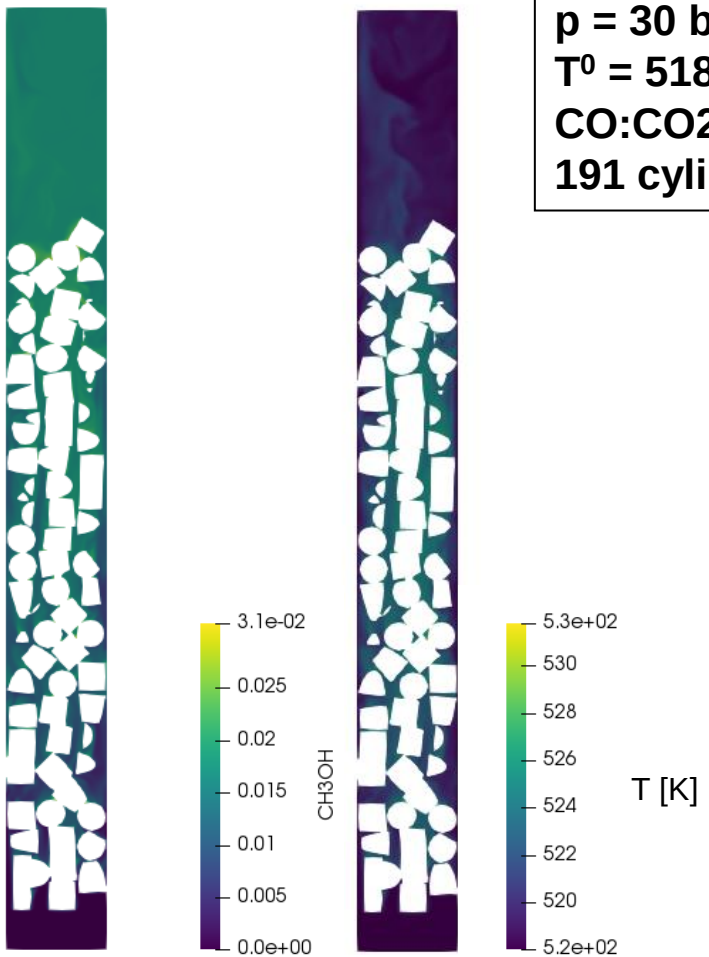
Velocity stream-lines and flow field







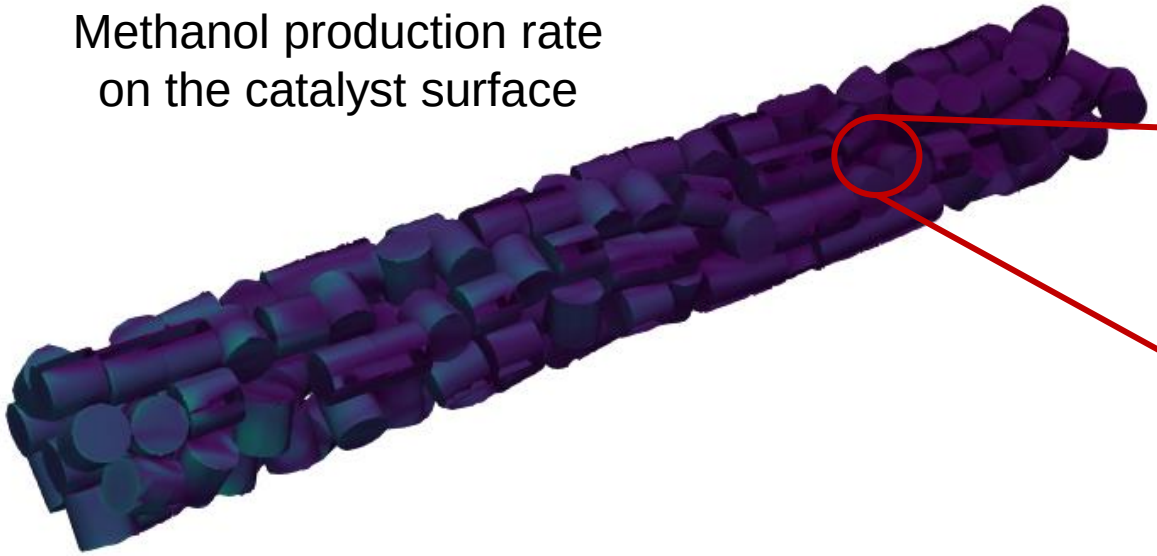

Gas phase maps



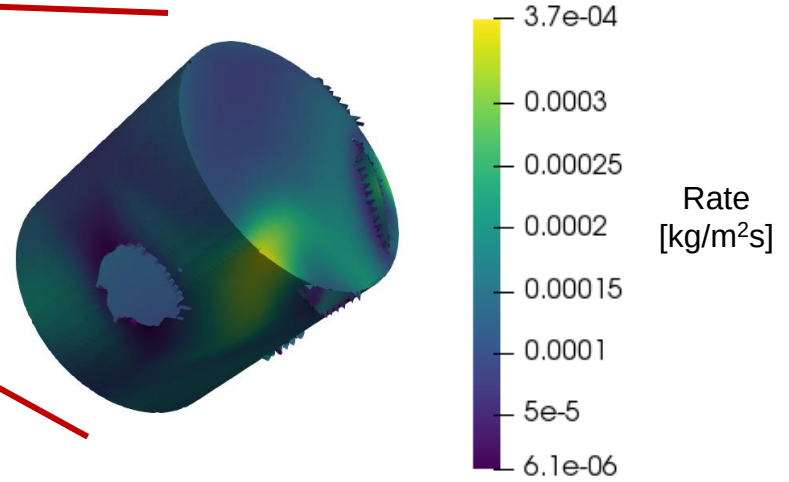

**Methanol synthesis**
**p = 30 bar**
**$T^0$ = 518 K**
**CO:CO2:H2 = 0.05:0.10:0.55**
**191 cylindrical pellets**

# *Show-case: cylinders - methanol synthesis*

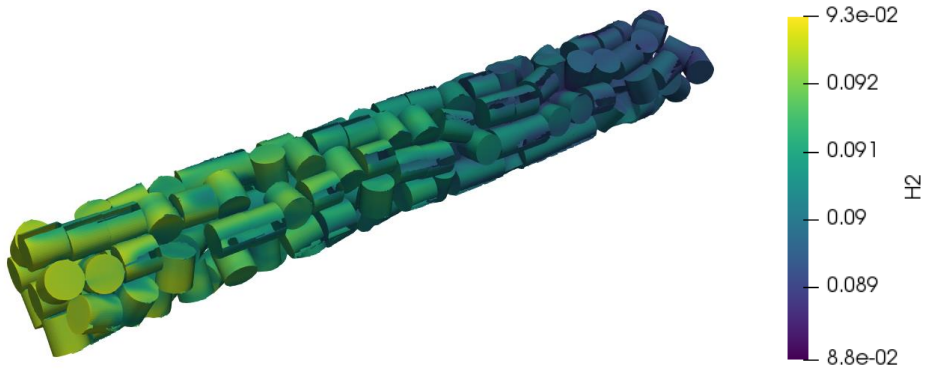
Methanol production rate
on the catalyst surface



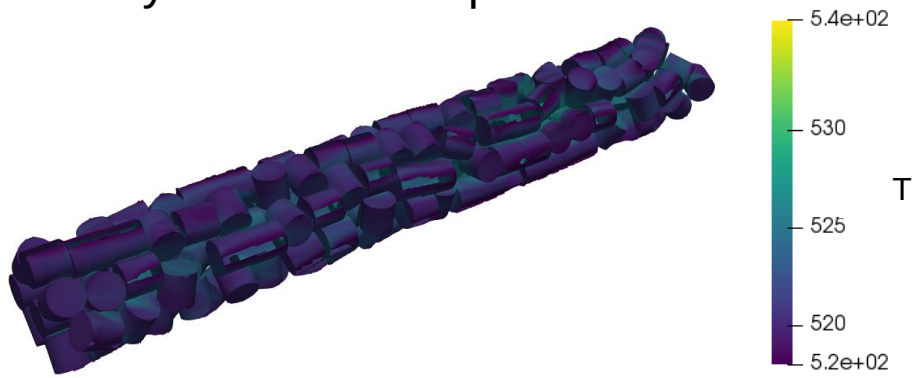
**Possibility to extract information on each single pellet**



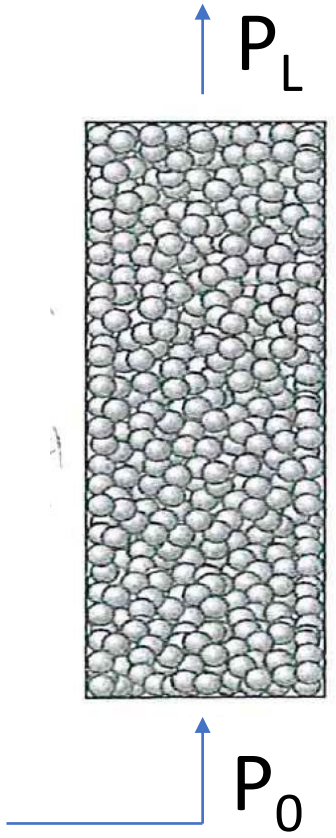

Catalyst surface mass fraction of $H_2$




Catalyst surface temperature



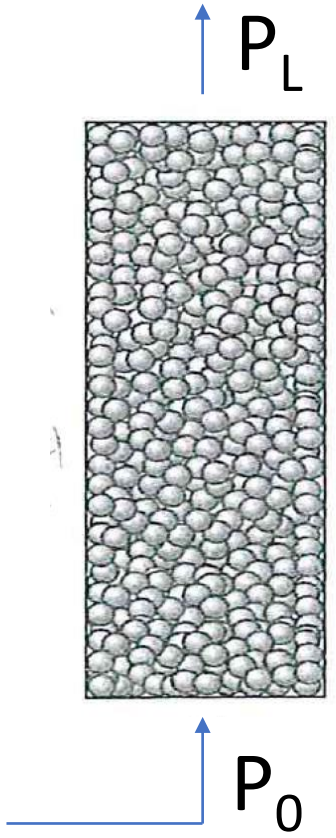
# *From fixed to fluidized bed reactors.*




Pressure drops in packed beds

$$\frac{\left[(P_0 - P_L)\right]}{L} = 150\frac{(1-\varepsilon)^2}{\varepsilon^3}\frac{\mu v_0}{D_p^2} + \frac{7}{4}\frac{\rho v_0^2}{D_p}\frac{(1-\varepsilon)}{\varepsilon^3}$$

# *From fixed to fluidized bed reactors.*



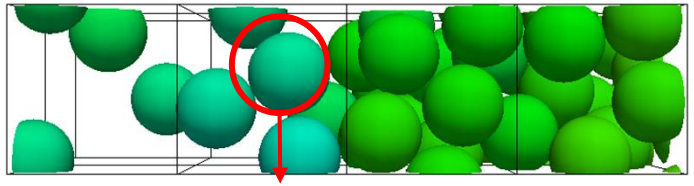

$$\begin{pmatrix}\text{drag force by}\\ \text{upward moving gas}\end{pmatrix} = \begin{pmatrix}\text{weight of}\\ \text{particles}\end{pmatrix}$$

$$\begin{pmatrix}\text{pressure drop}\\ \text{across bed}\end{pmatrix}\begin{pmatrix}\text{cross sectional}\\ \text{area of tube}\end{pmatrix} = \begin{pmatrix}\text{volume of}\\ \text{the bed}\end{pmatrix}\begin{pmatrix}\text{fraction}\\ \text{of solids}\end{pmatrix}\begin{pmatrix}\text{specific weight}\\ \text{of solids}\end{pmatrix}$$

Minimum fluidization velocity

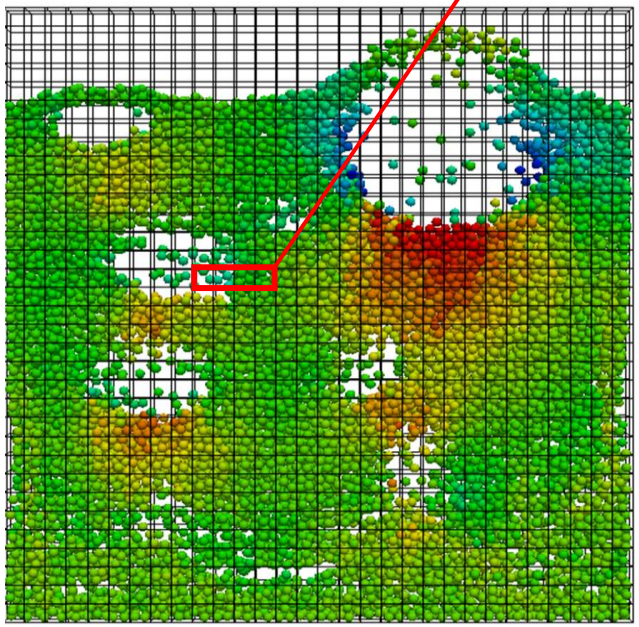
# *Coupling CFD-DEM with Microkinetic Modeling*

CELL *i*



PARTICLE *p*

**Snapshot of the fluidized bed reactor at time *t***



$t_n + \Delta t$

**Update of the solid phase**

Discrete Element Method (**DEM**)

GAS PHASE FROZEN

$t_n$

**SOLUTION ALGORITHM**

**Update the gas phase**

CFD

SOLID PHASE FROZEN AND UPDATED

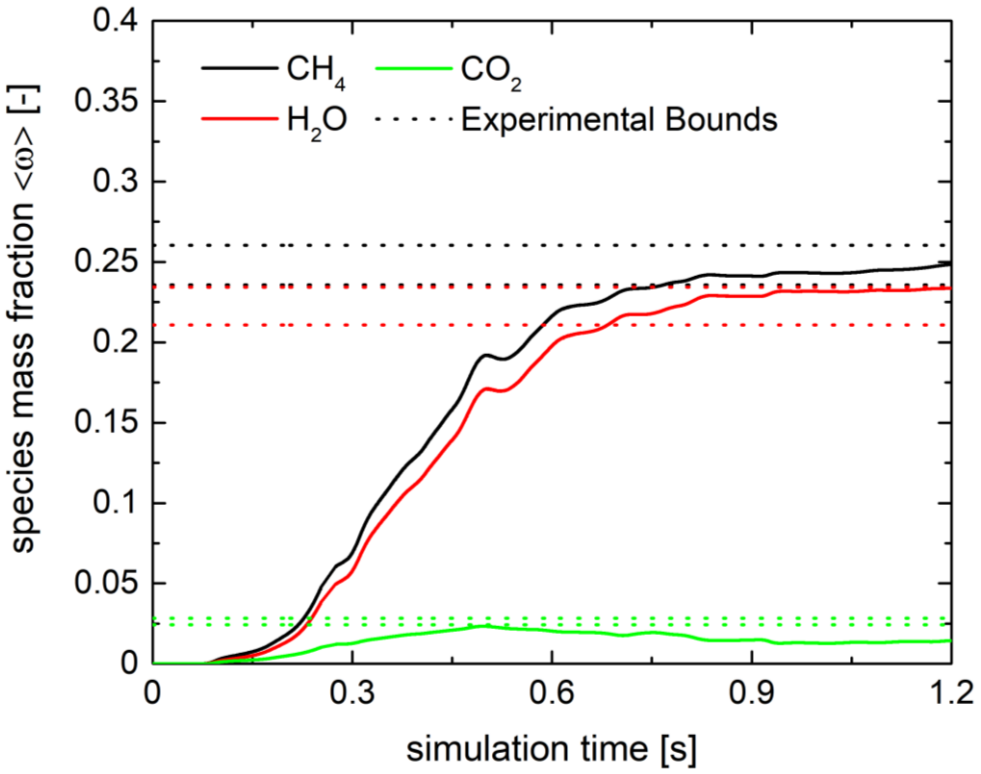
**Update the Gas-Solid Transfer**

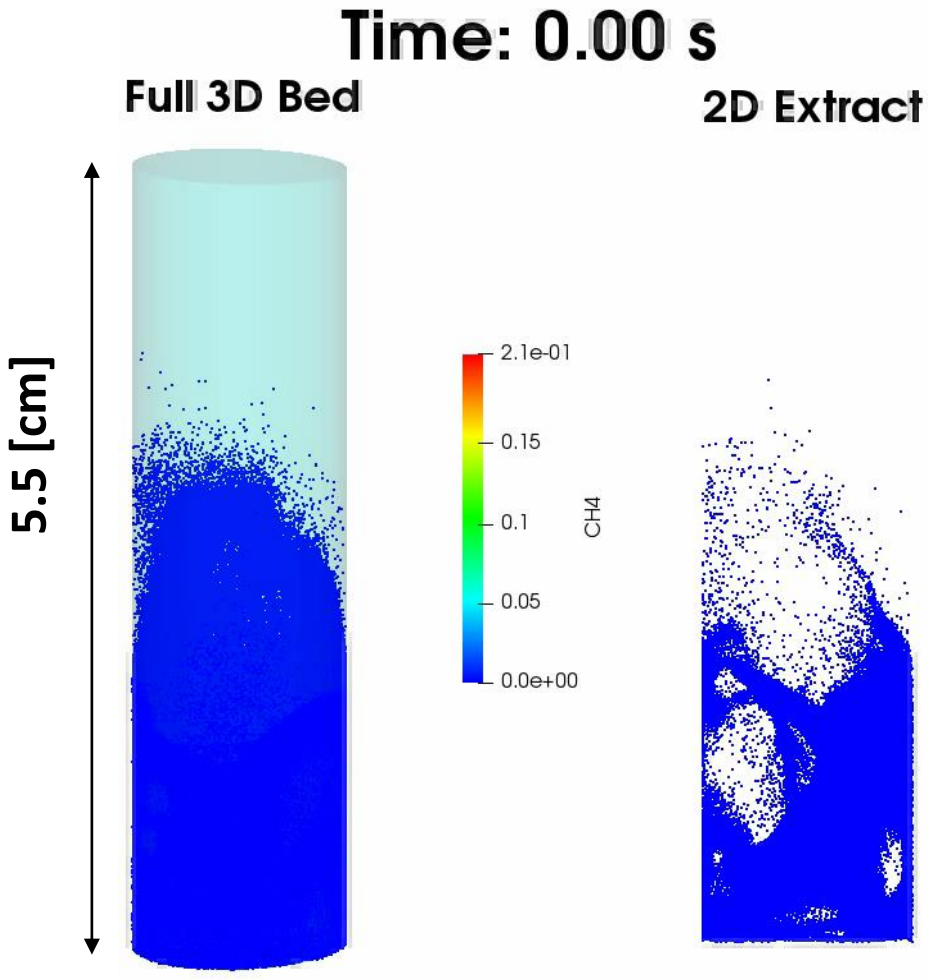
Compute the momentum, heat and mass transfer between gas in cell *i* and each particle p in cell *i*


R. Uglietti, M. Bracconi, M. Maestri, Reac. Chem. Eng., 3 (2018) 527

R. Uglietti, M. Bracconi, M. Maestri, Reac. Chem. Eng., 5 (2020) 278

# *CFD-DEM: example.*

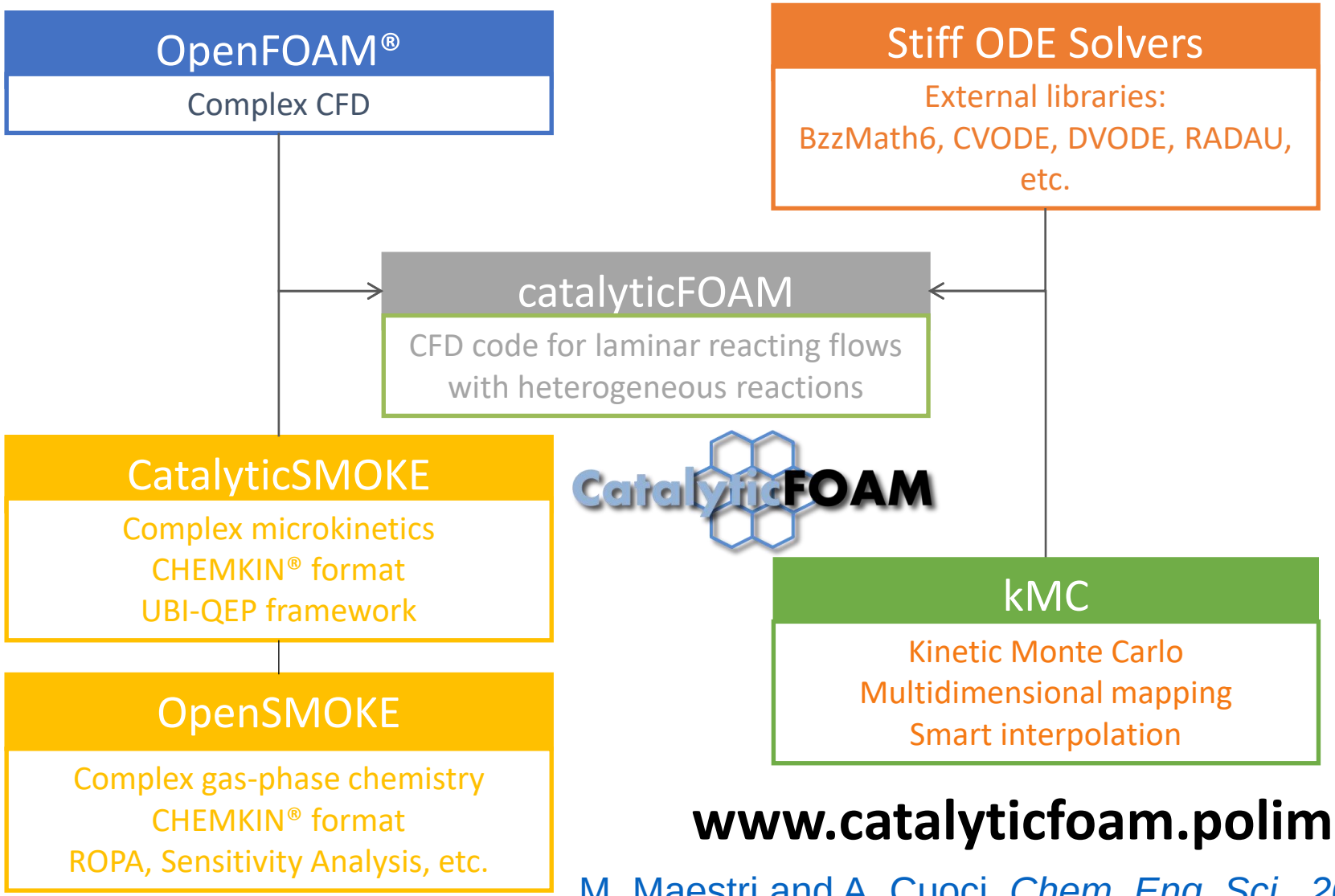




R. Uglietti, M. Bracconi, M. Maestri, Reac. Chem. Eng., 3 (2018) 527
R. Uglietti, M. Bracconi, M. Maestri, Reac. Chem. Eng., 5 (2020) 278

# *catalyticFOAM structure*

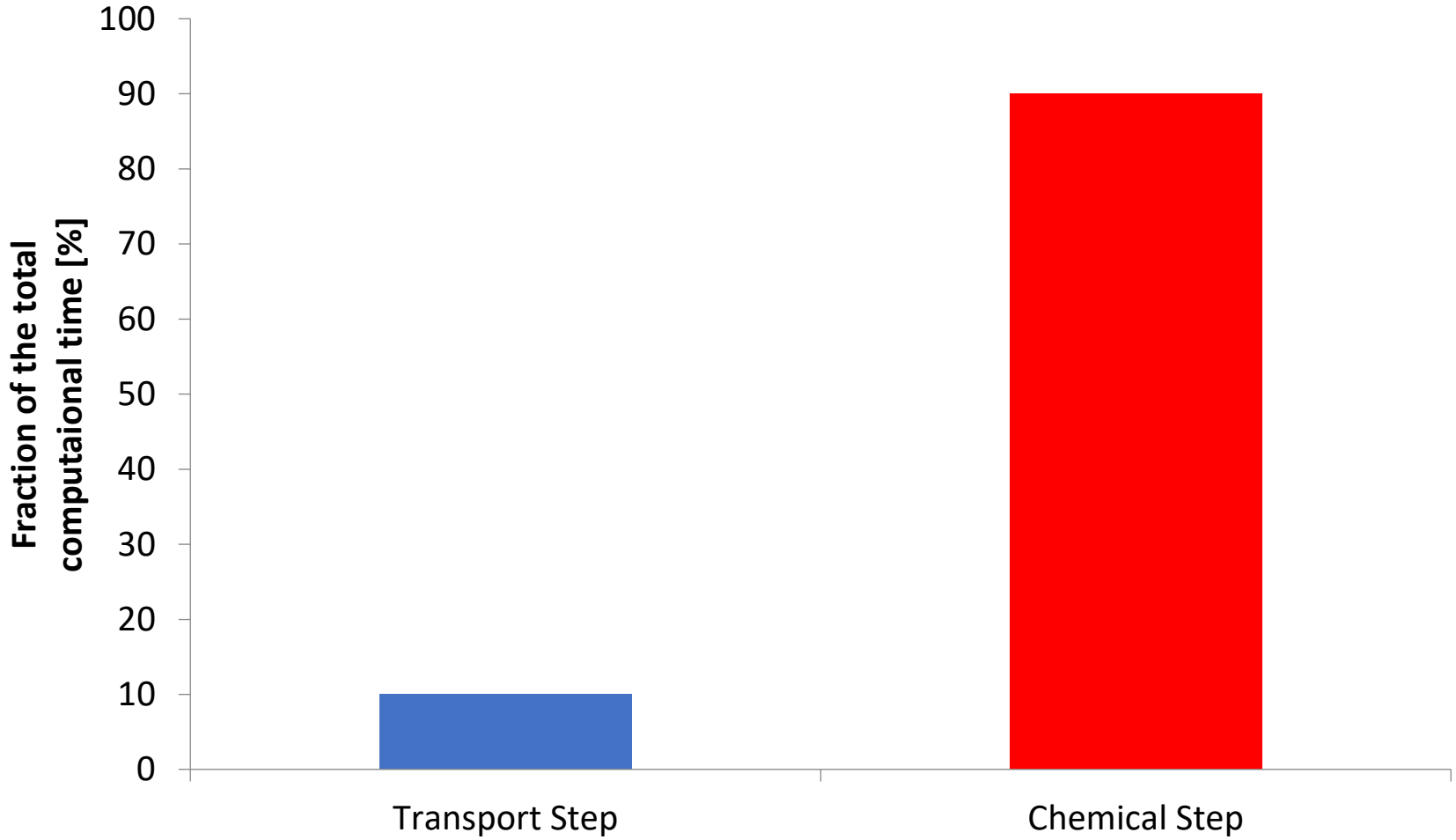



**www.catalyticfoam.polimi.it**

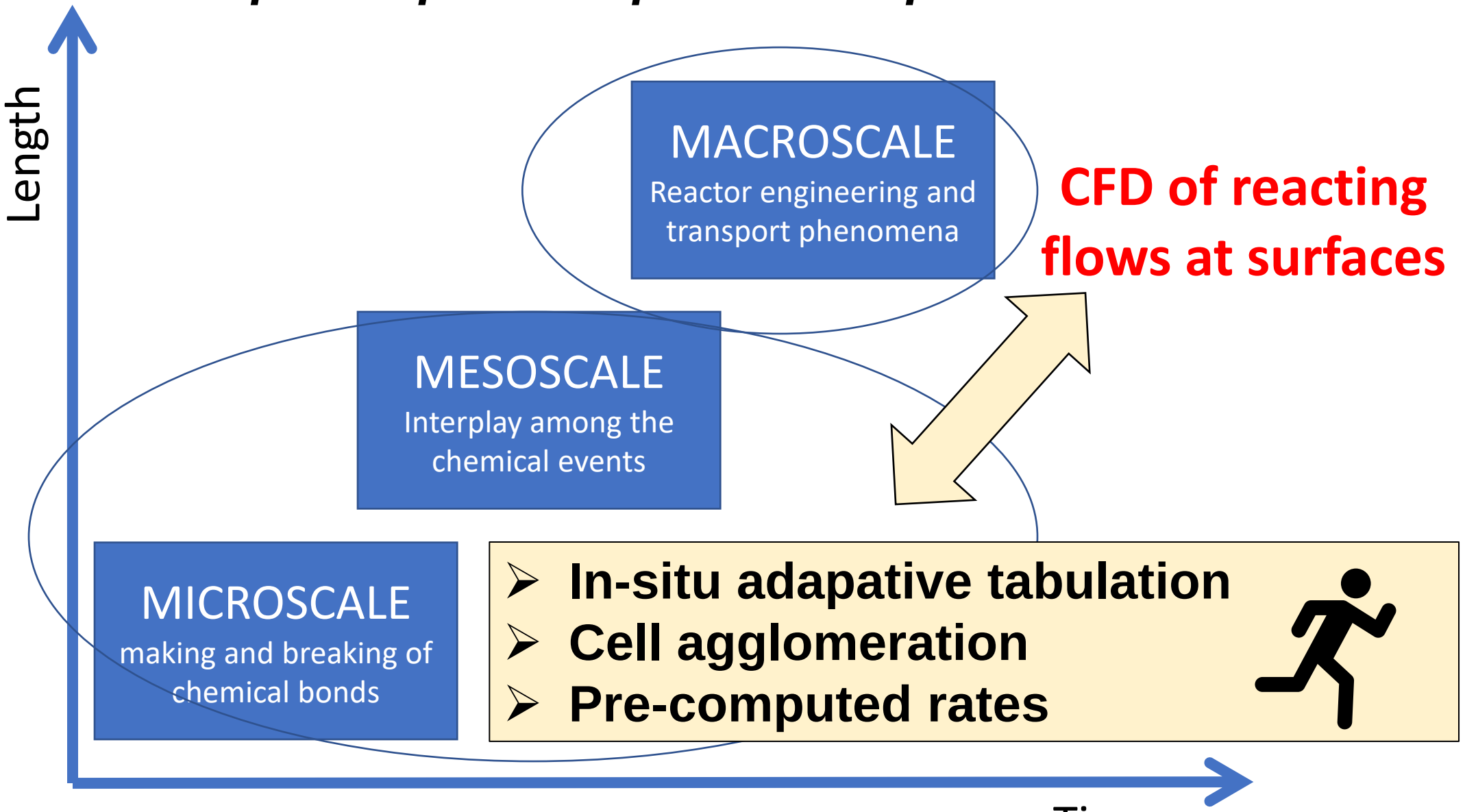
M. Maestri and A. Cuoci, *Chem. Eng. Sci., 2013*

T. Maffei, et al., *Chem. Eng. J., 2016*

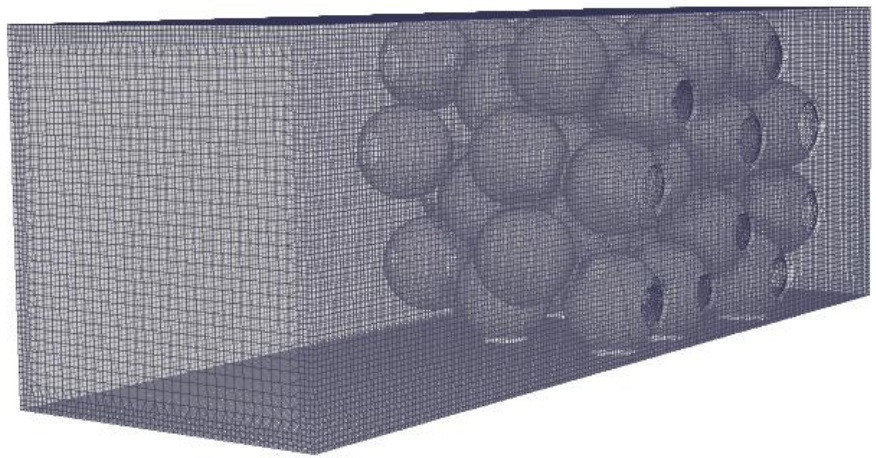
# *The computational time of the chemical step is the bottleneck*

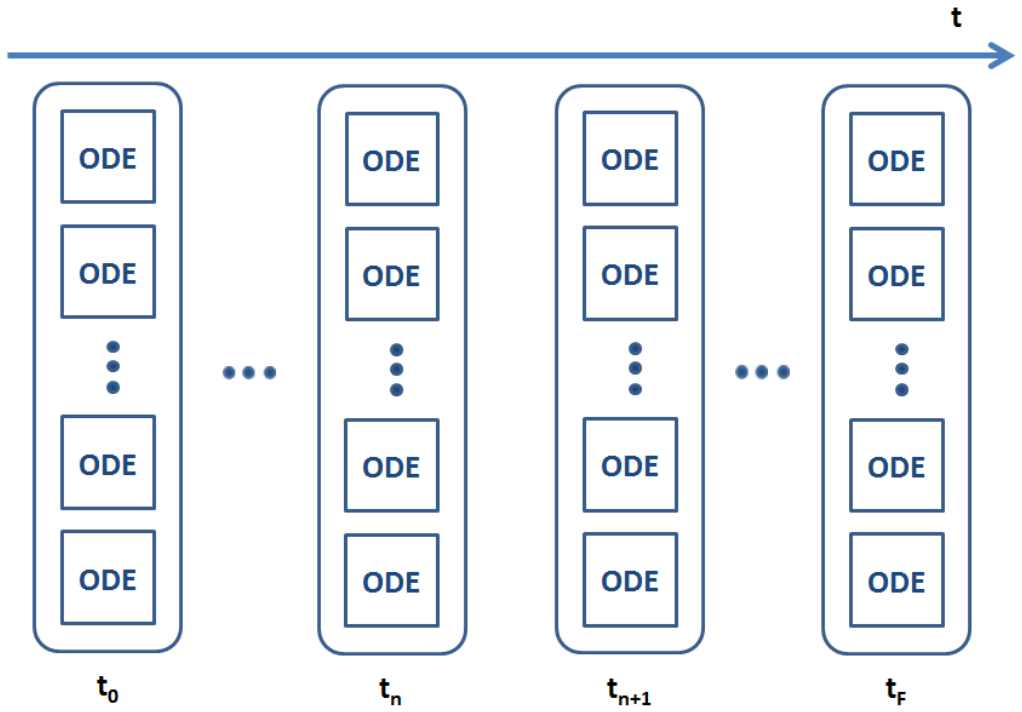


# *Speed-up techniques are required*



# *In situ Adaptive Tabulation - ISAT*



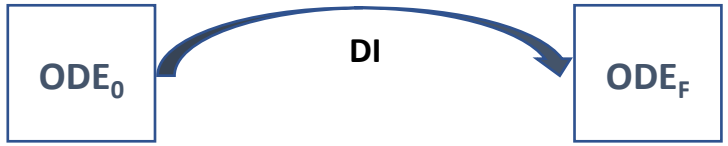



M. Bracconi, A. Cuoci, M. Maestri, AIChE Journal, 63 (2017) 95

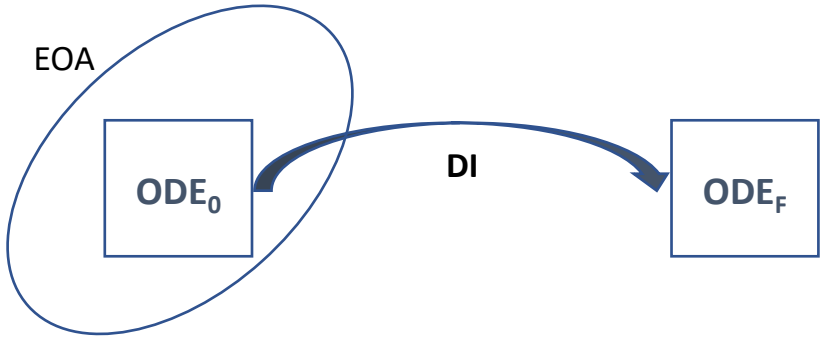
# *ISAT: How does it work?*

$ODE_0$

# *ISAT: How does it work?*



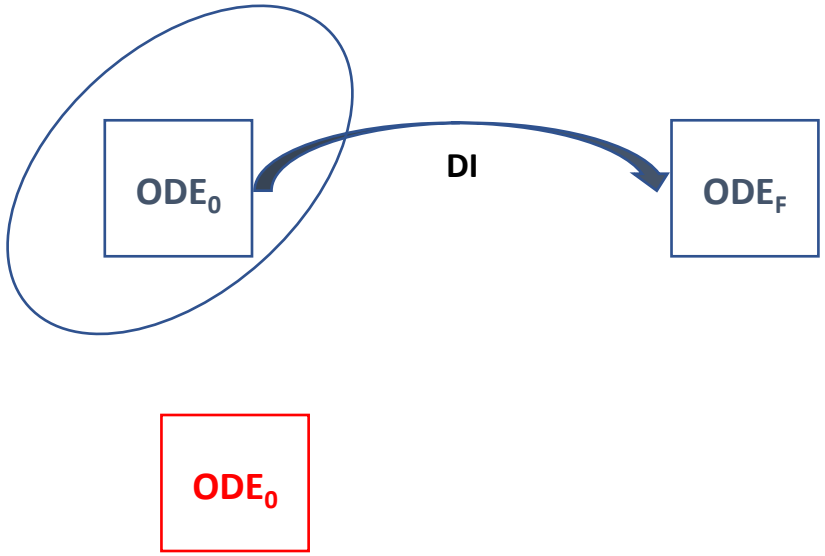
# *ISAT: How does it work?*



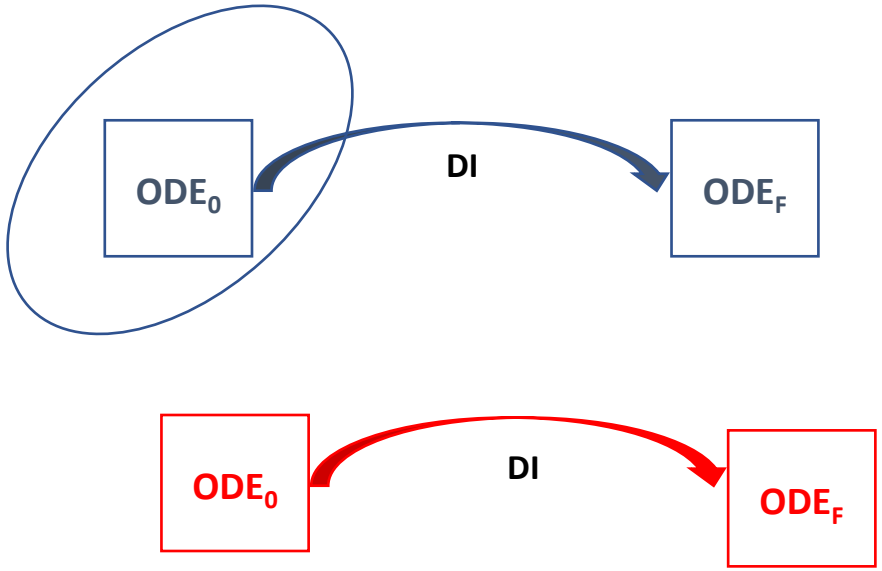

addition

The ODE systems whose initial condition falls inside the EOA could be solved by extrapolating the results of this one with an error at most of $\boldsymbol{\varepsilon_{tol}}$

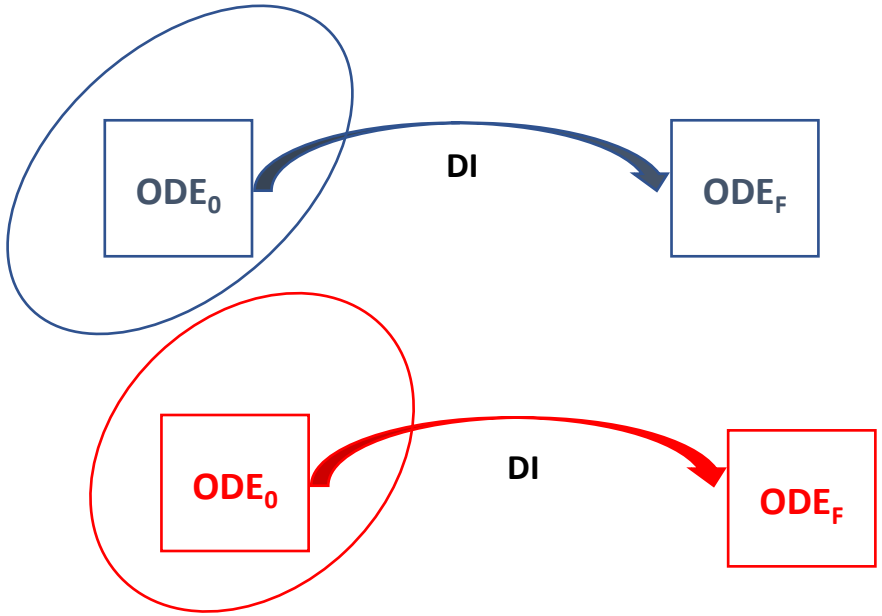
# *ISAT: How does it work?*

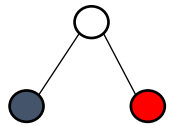


# *ISAT: How does it work?*



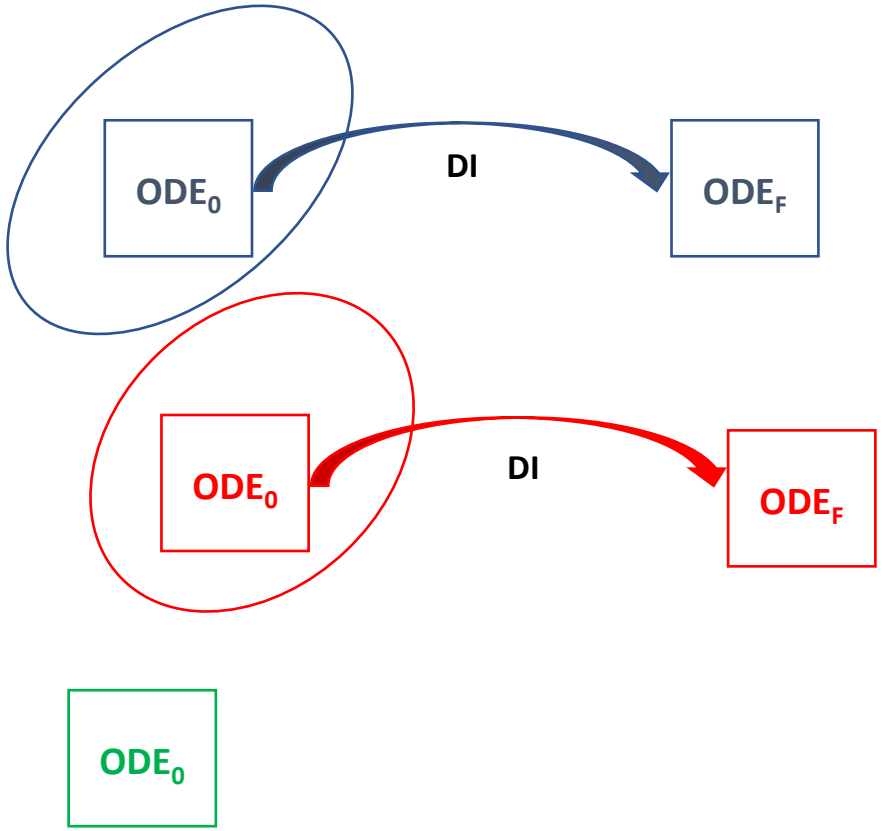
# *ISAT: How does it work?*

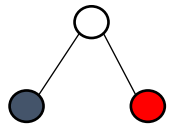


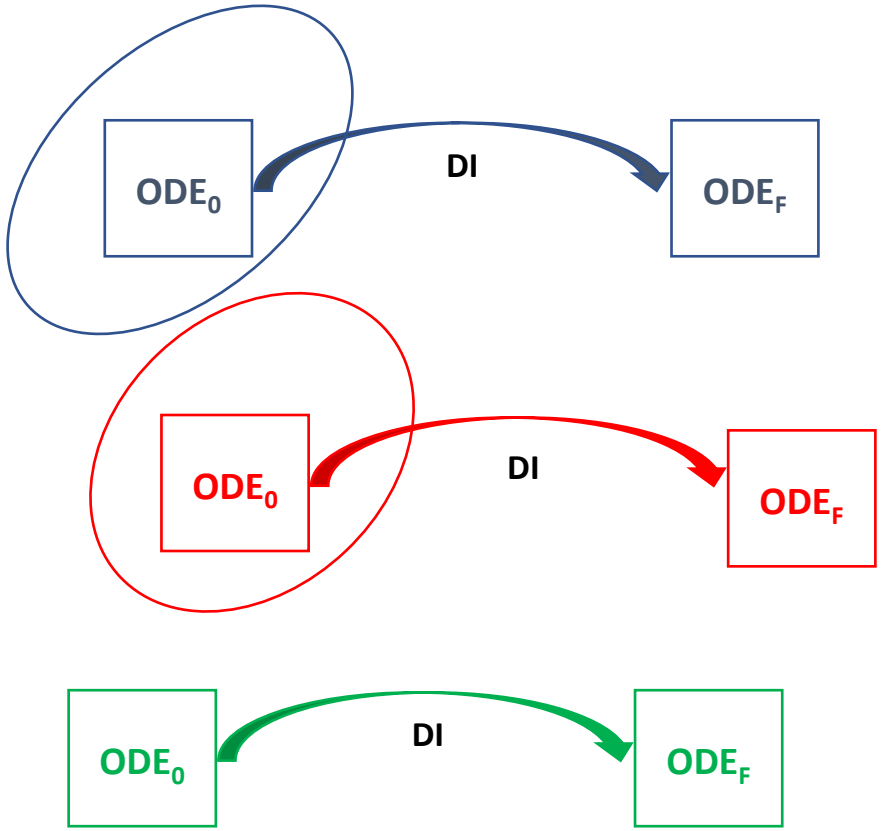
Storage table in the form of a binary tree

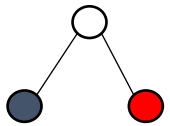
# *ISAT: How does it work?*



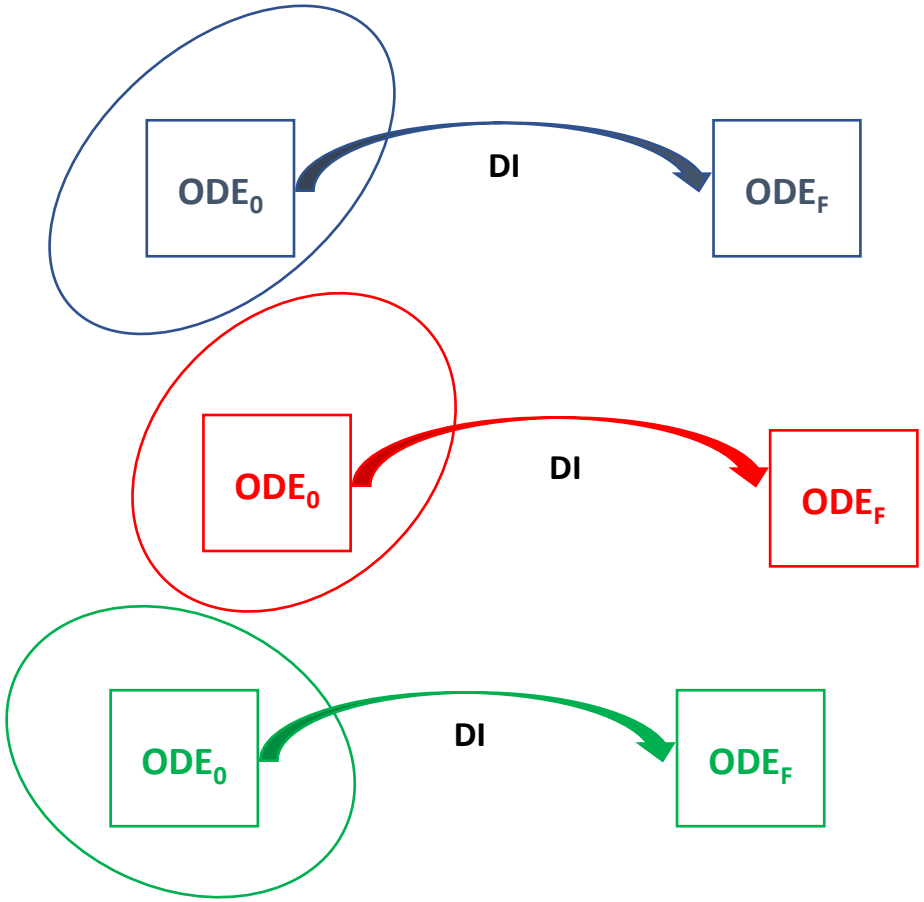



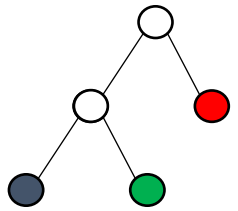
# *ISAT: How does it work?*



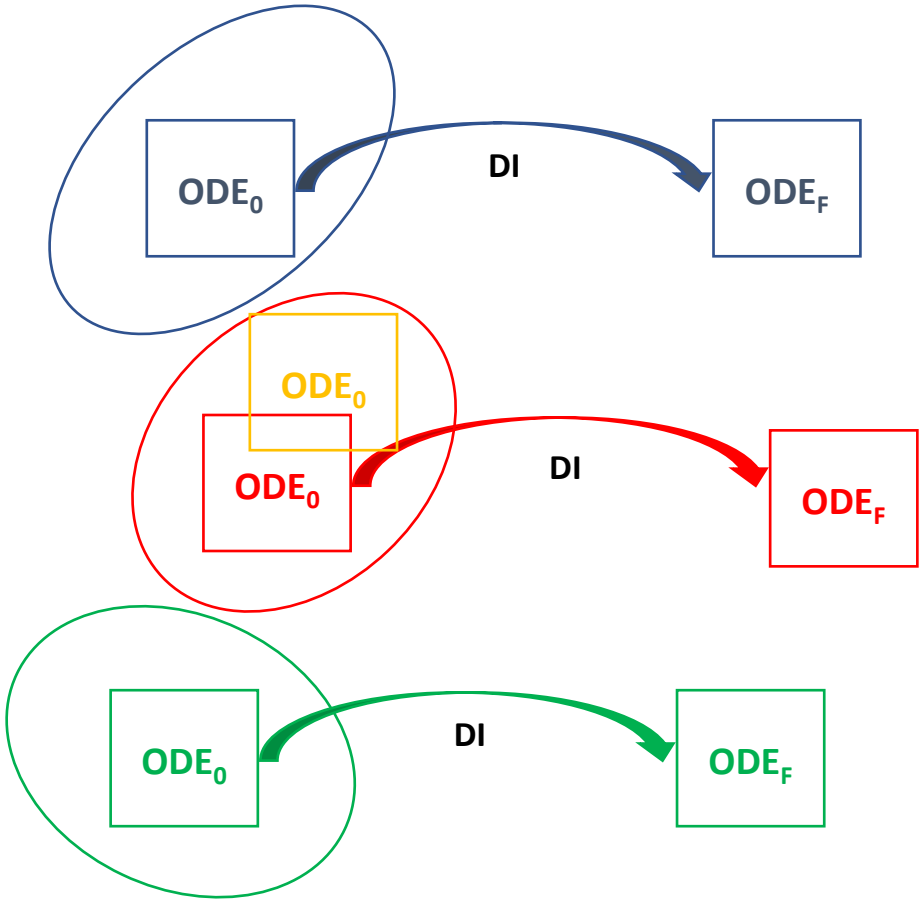
# *ISAT: How does it work?*

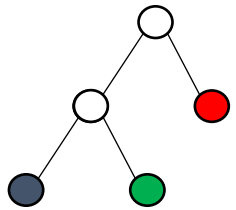


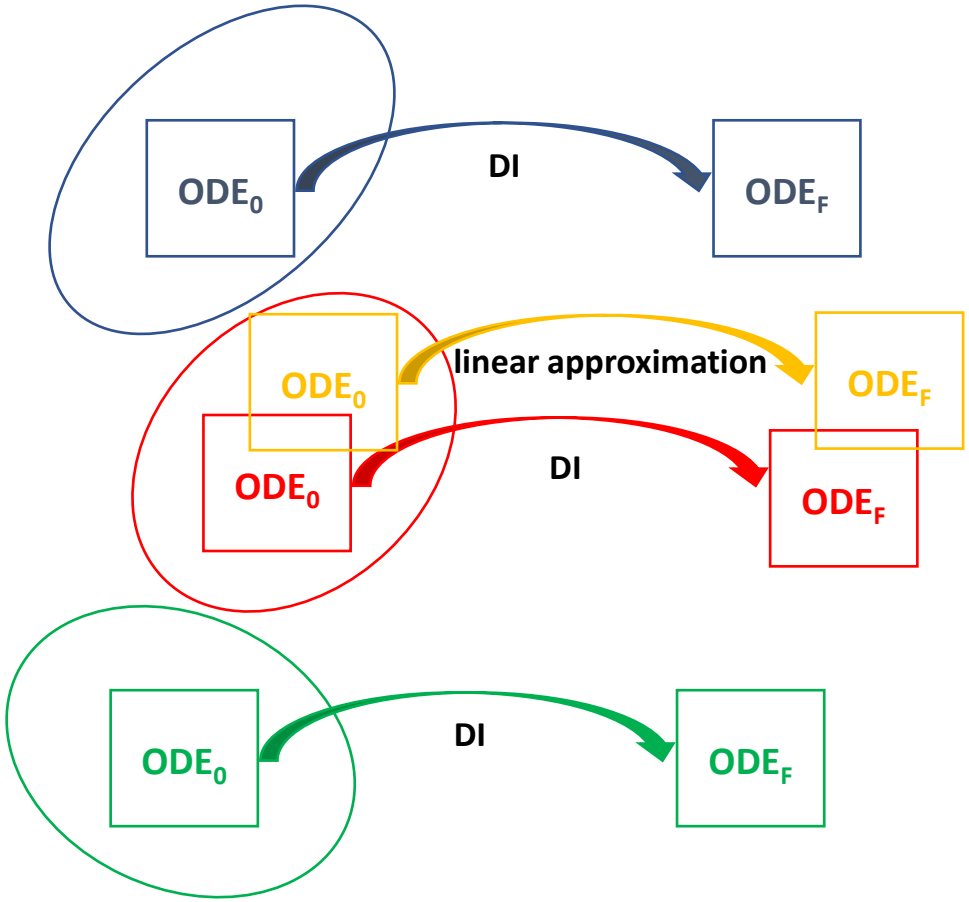
ISAT table is built during the simulation (“in situ”)

# *ISAT: How does it work?*

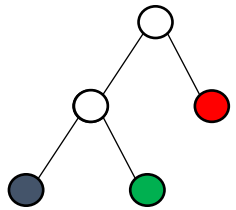


# *ISAT: How does it work?*

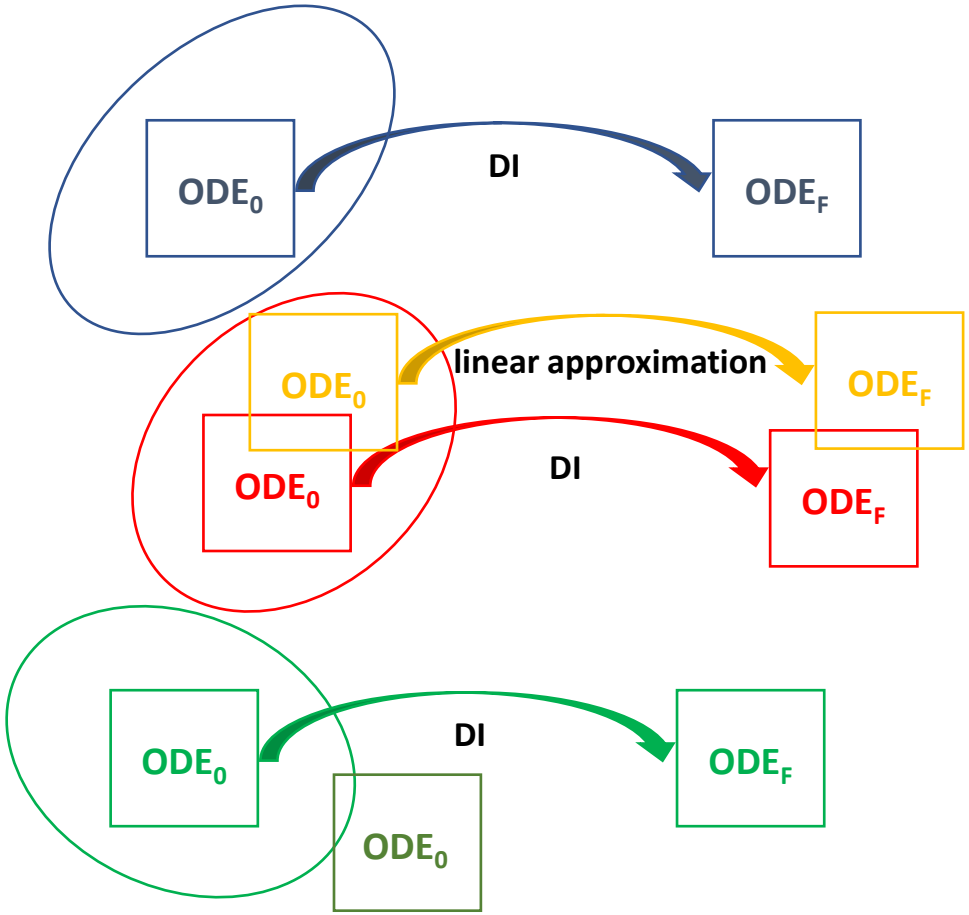


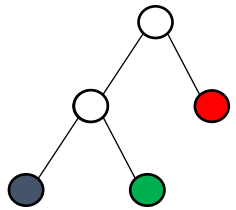

retrieval



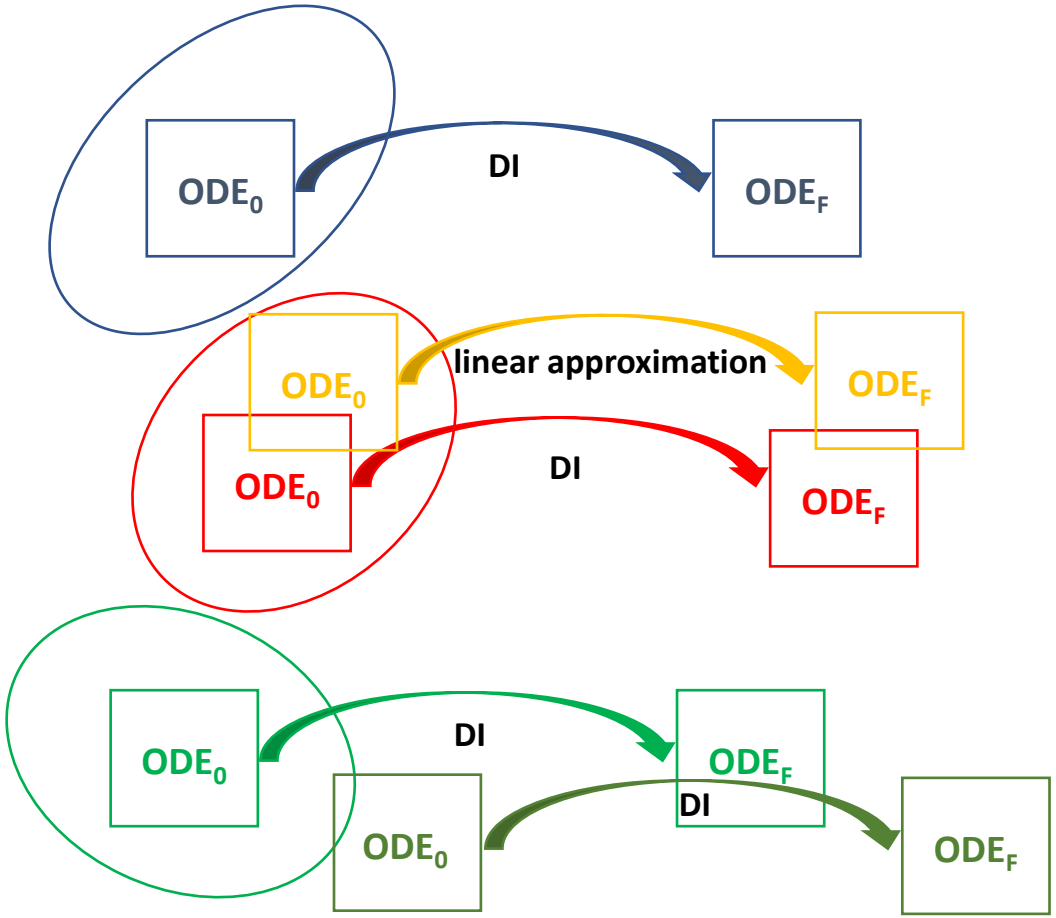
The information recovered during previous integration are exploited to approximate the ODE solution

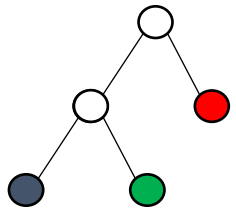
# *ISAT: How does it work?*



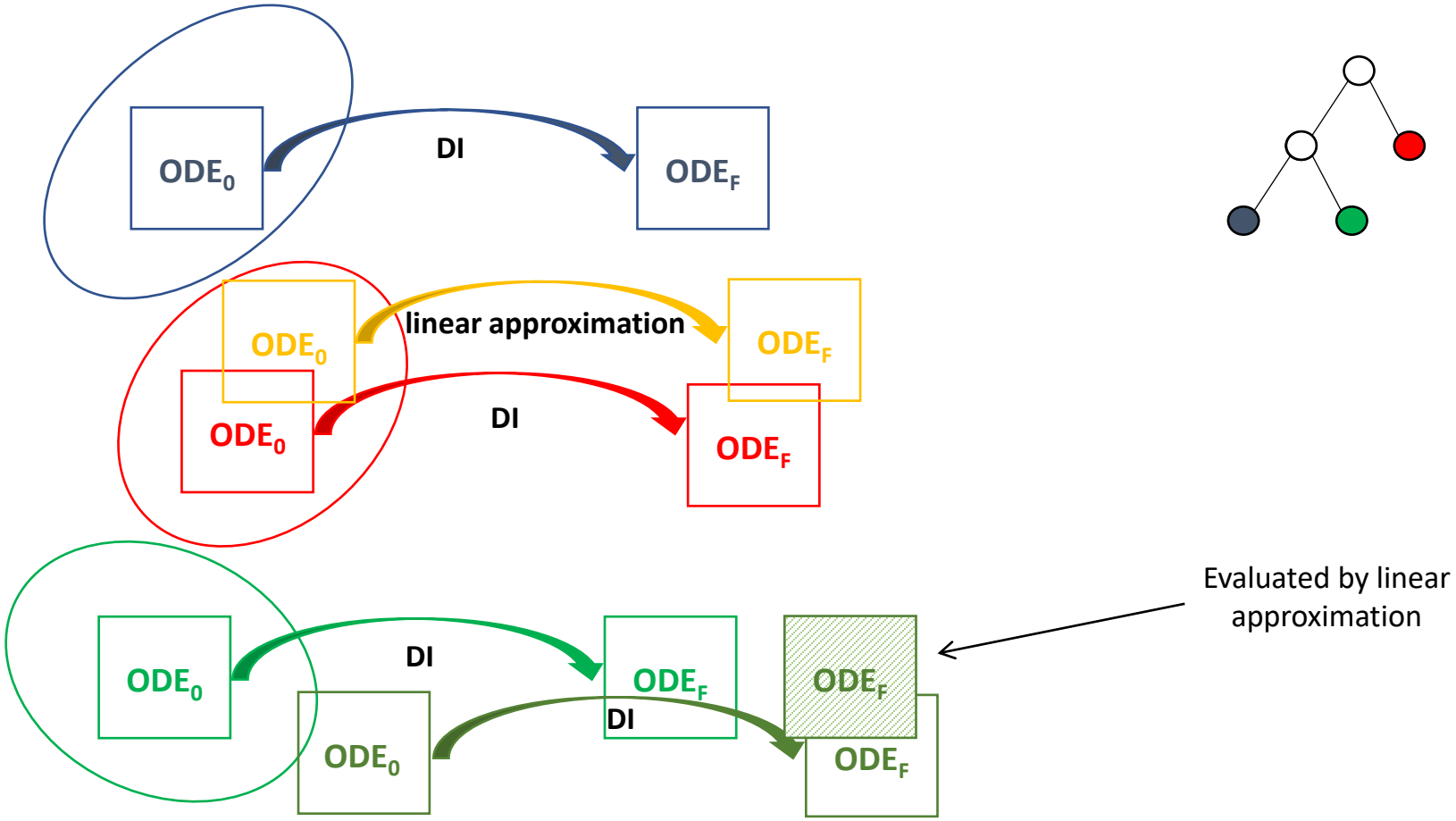



# *ISAT: How does it work?*

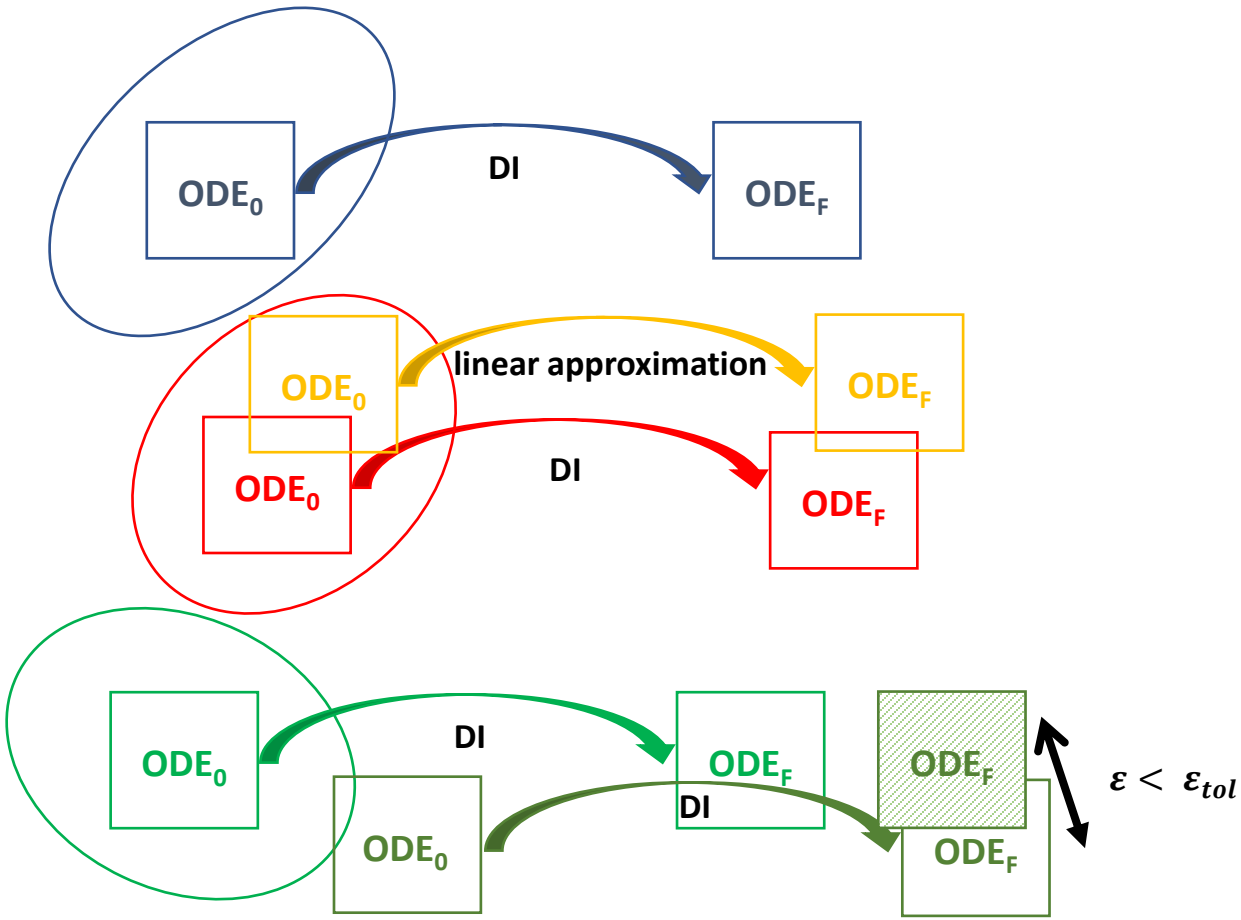


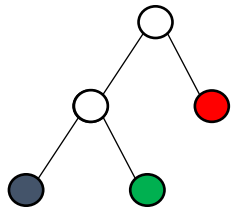



# *ISAT: How does it work?*



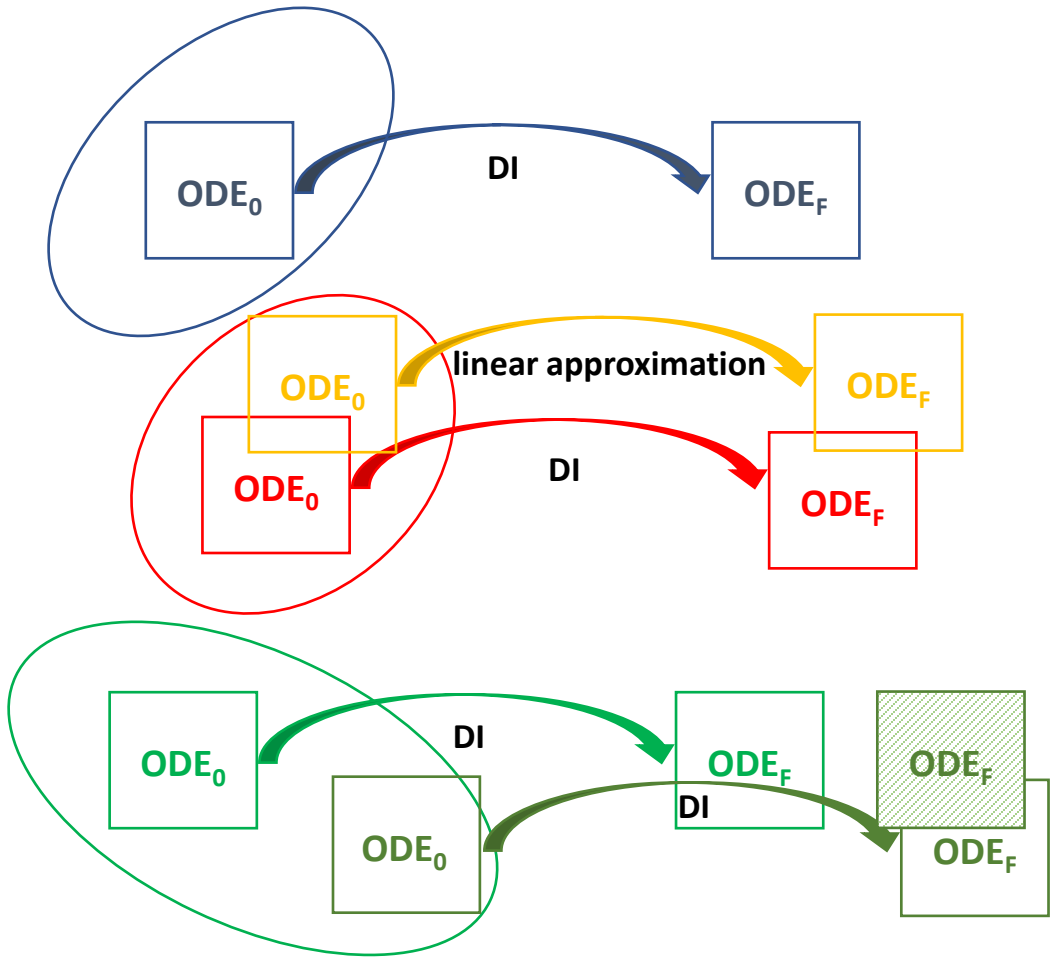
# *ISAT: How does it work?*

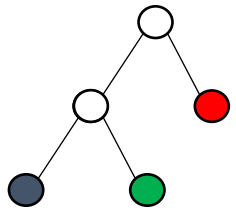


The approximation error is evaluated and if lower than $\varepsilon_{tol}$ the EOA grows to cover the query

# *ISAT: How does it work?*



growth

# *Packed bed reactor simulation: speed-up*

**Computational domain**

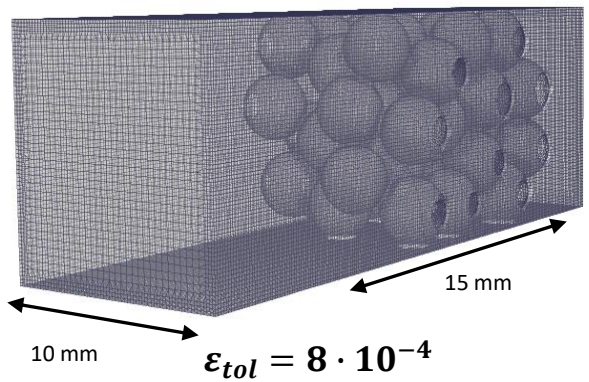
- 25 particles
- 3D domain
- 500k mesh cells

**Operative conditions**

- Isothermal
- Feed velocity: 0.35 m/s
- $H_2O$/CH4: 3.5 : 1

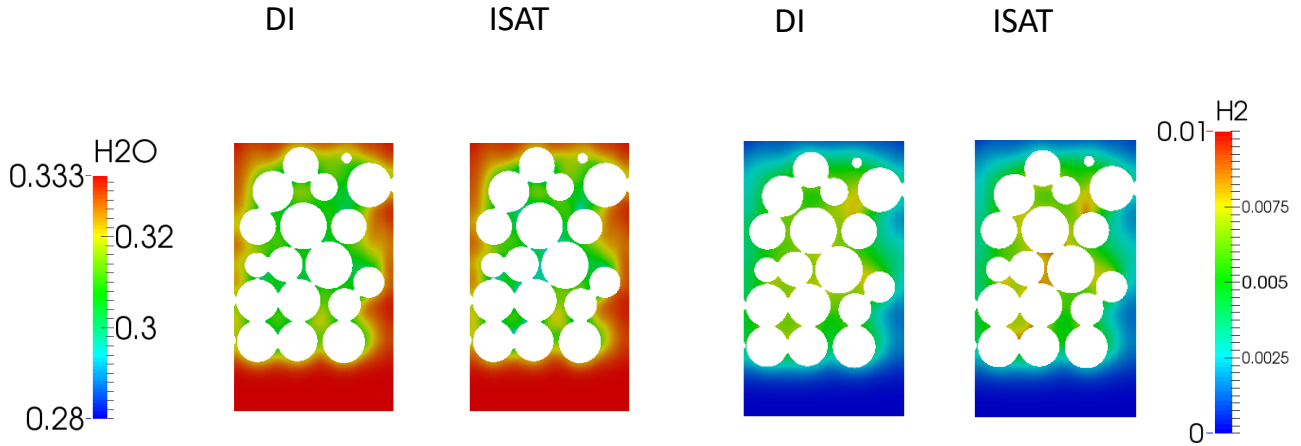
**Microkinetic model**

- 21 gas species
- 13 adsorbed species
- 80 surface reactions




$\varepsilon_{tol} = 8 \cdot 10^{-4}$

Time = 0.0015 s



# *Packed bed reactor simulation: speed-up*

**Computational domain**

- 25 particles
- 3D domain
- 500k mesh cells

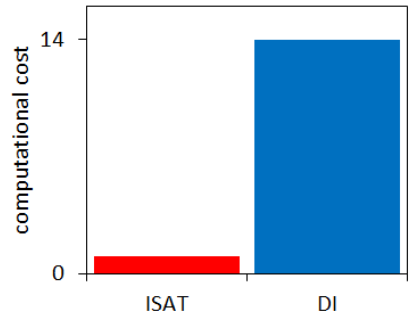
**Operative conditions**

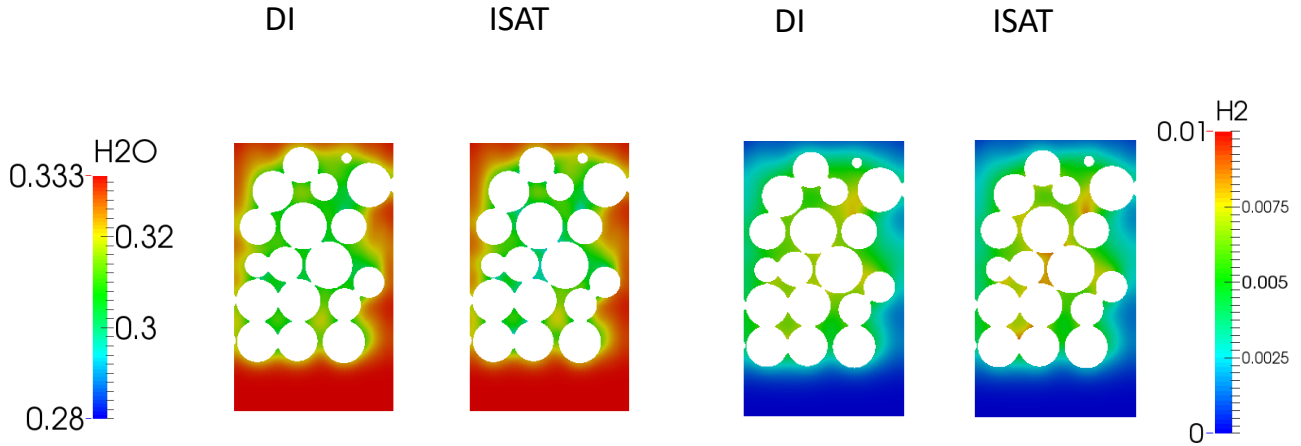
- Isothermal
- Feed velocity: 0.35 m/s
- $H_2O$/CH4: 3.5 : 1

**Microkinetic model**

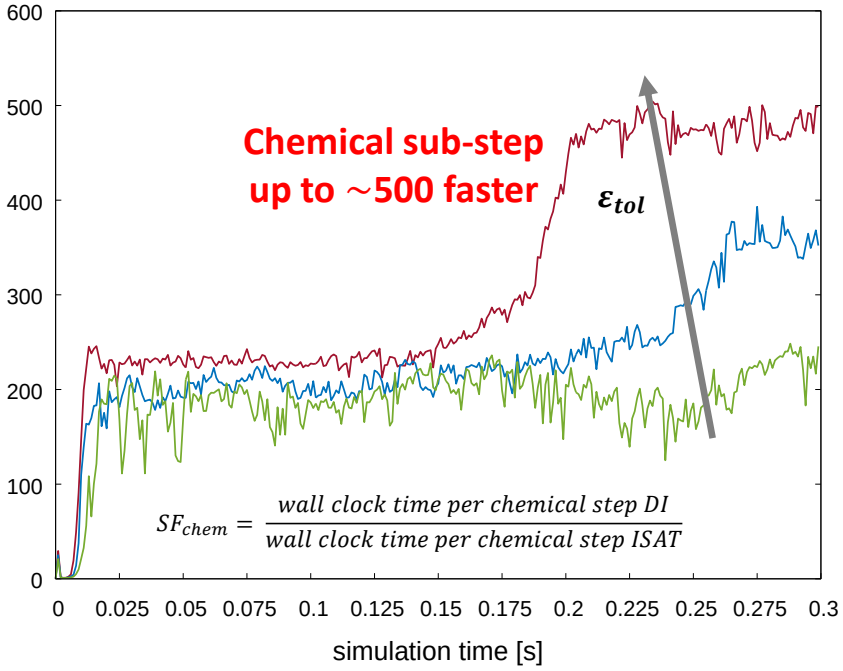
- 21 gas species
- 13 adsorbed species
- 80 surface reactions

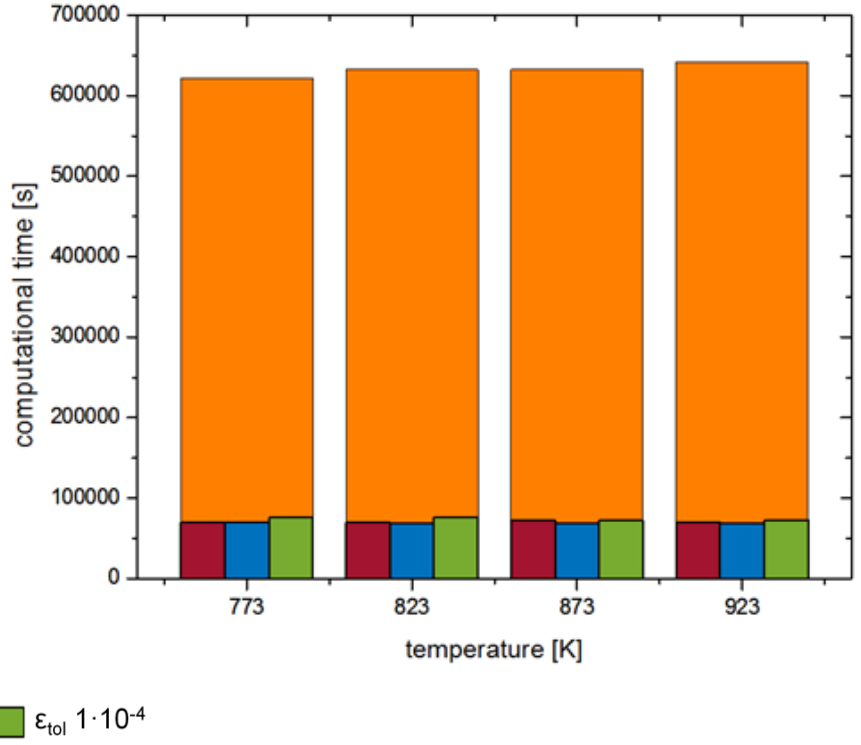
Simulation speed-up
15 times



# ISAT: Computational efficiency



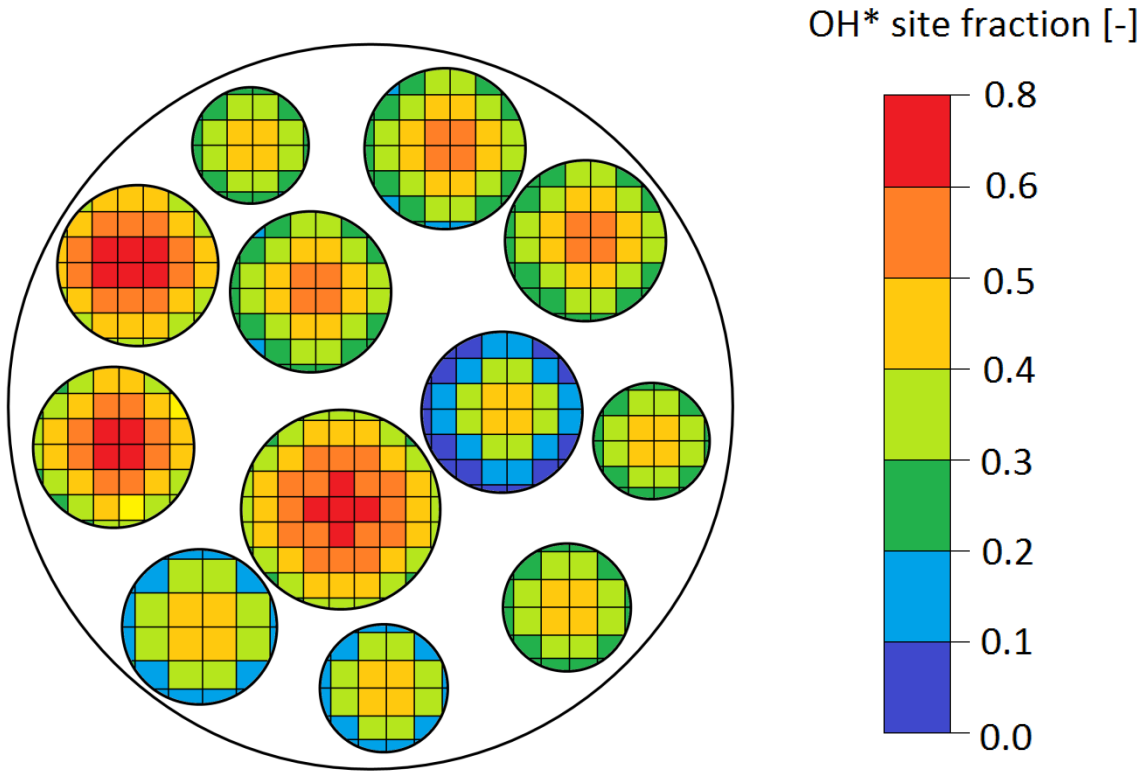

M. Bracconi, A. Cuoci, M. Maestri, AIChE Journal, 63 (2017) 95

**Overall computational time reduced up to ~5-15 times**

# *Cell Agglomeration algorithm*

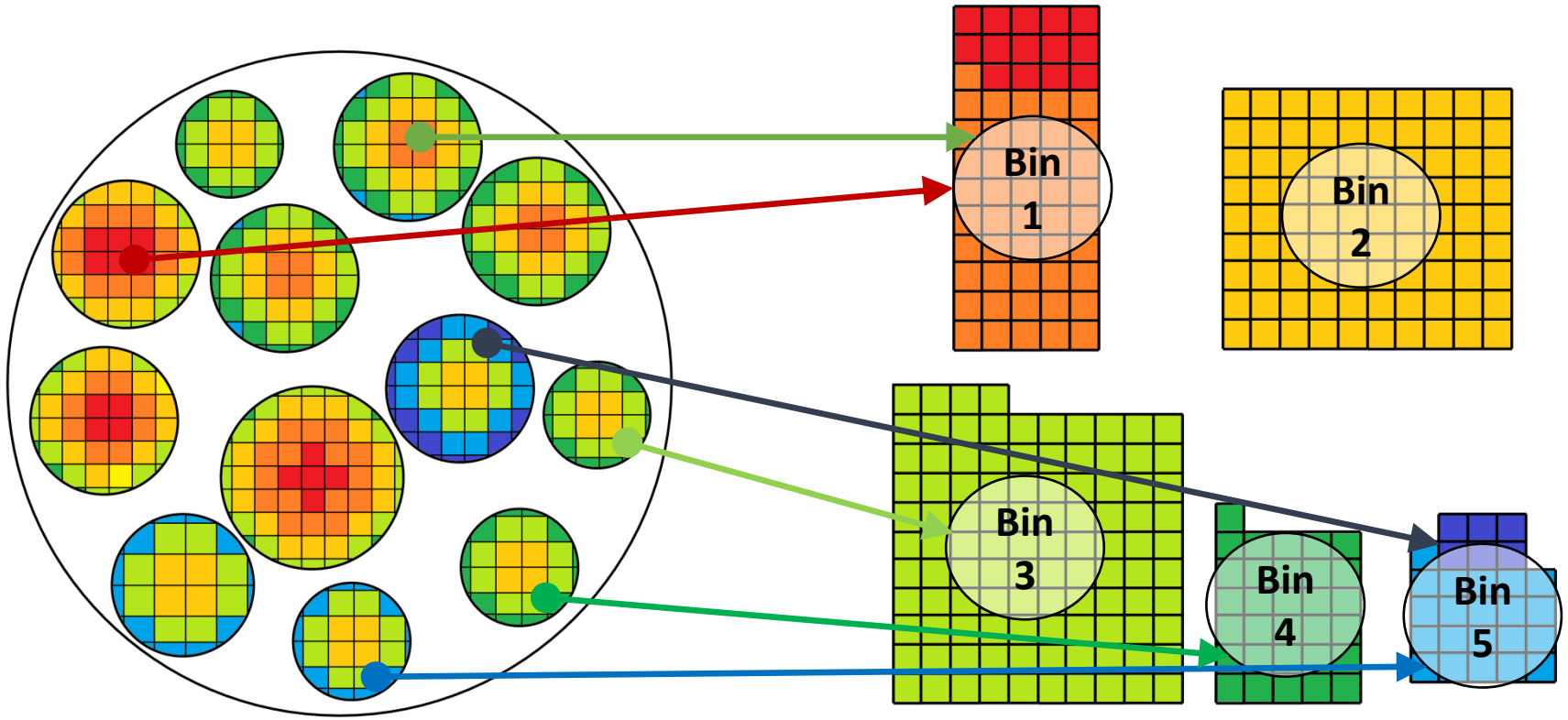
Rebughini, S., Cuoci, A., Dixon, A. G., Maestri, M., Comput. Chem. Eng., 97 (2017) 175

# *Cell Agglomeration algorithm*



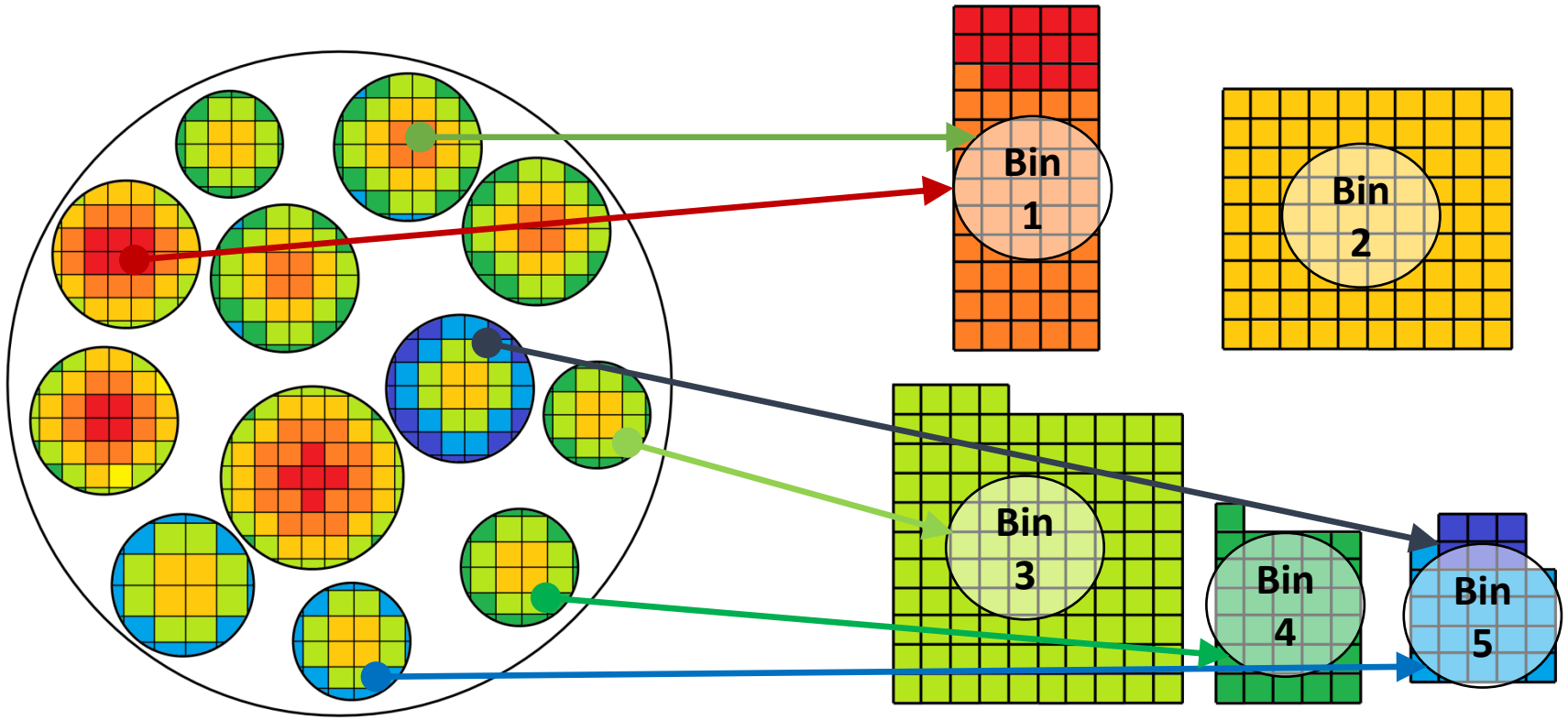

Rebughini, S., Cuoci, A., Dixon, A. G., Maestri, M., Comput. Chem. Eng., 97 (2017) 175

# *Cell Agglomeration algorithm*




Rebughini, S., Cuoci, A., Dixon, A. G., Maestri, M., Comput. Chem. Eng., 97 (2017) 175

# *Cell Agglomeration algorithm*



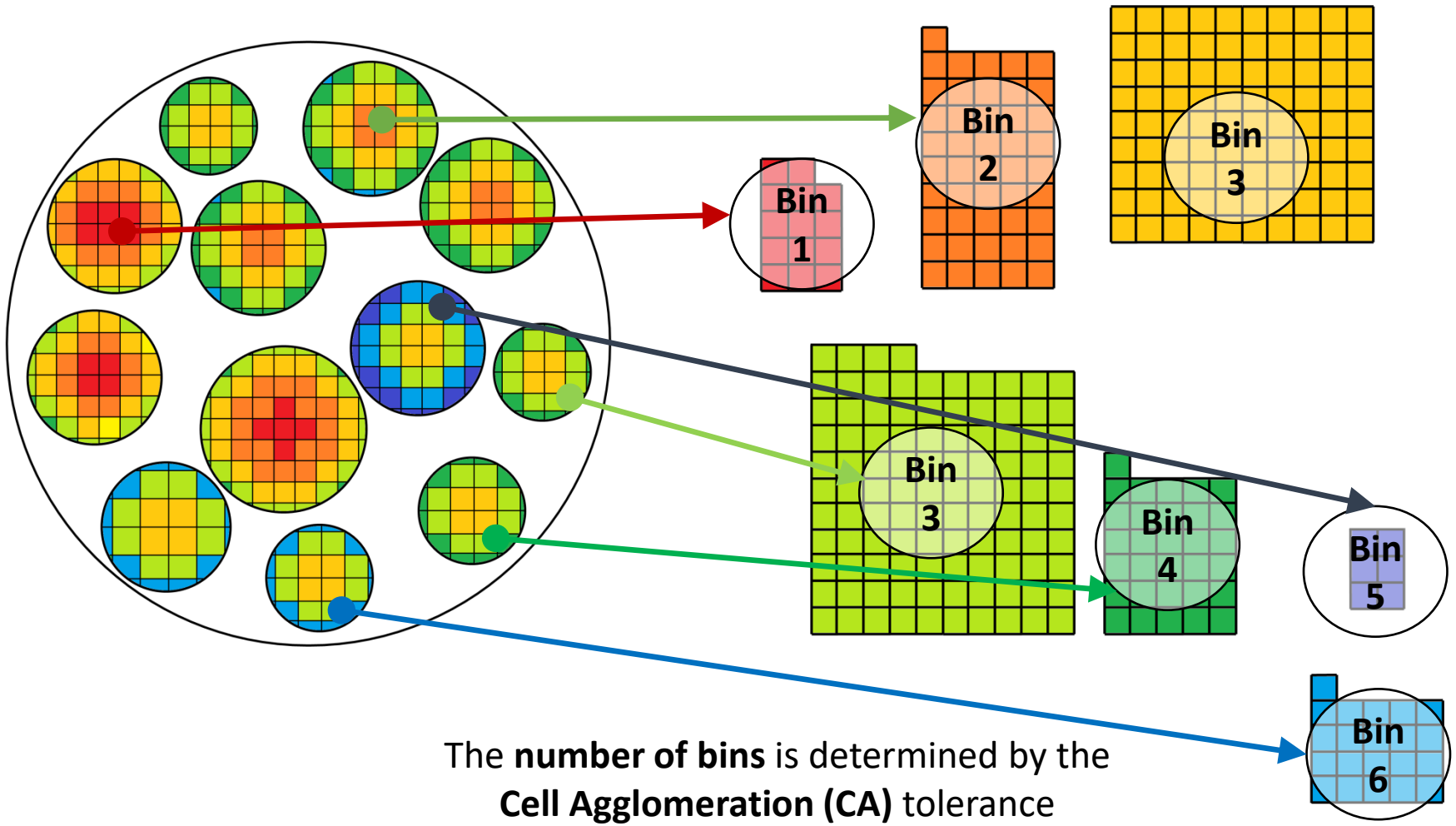

Number of calculated coverage:

Number of calculated coverage:

**NUMBER OF CELLS** **>>** **NUMBER OF BINS**

Rebughini, S., Cuoci, A., Dixon, A. G., Maestri, M., Comput. Chem. Eng., 97 (2017) 175

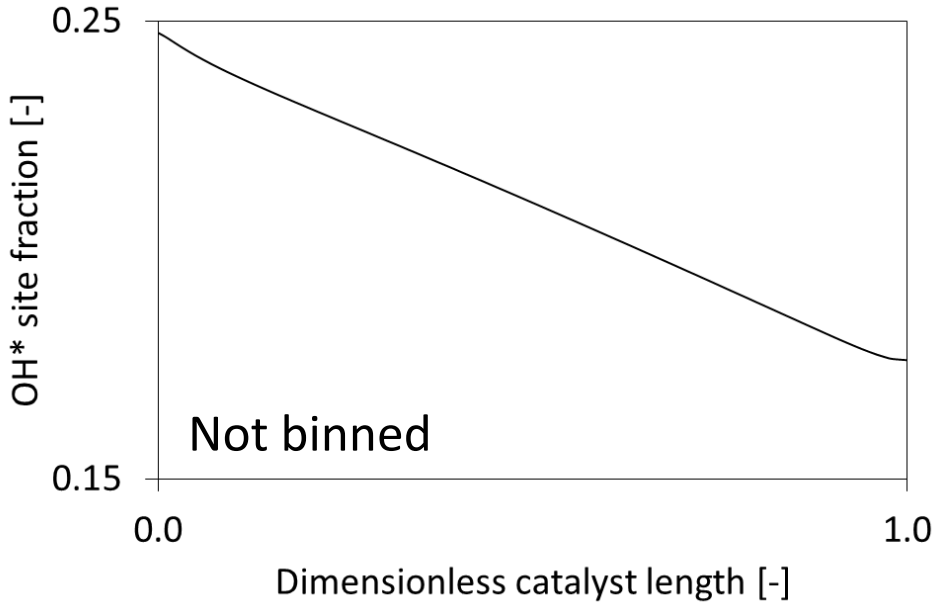
# *Cell Agglomeration algorithm*




The **number of bins** is determined by the **Cell Agglomeration (CA)** tolerance

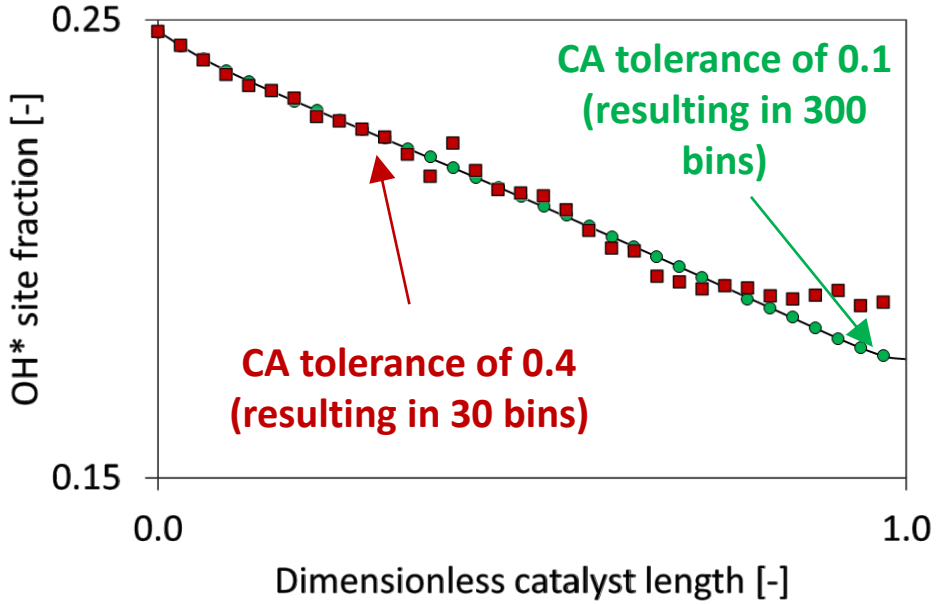
Rebughini, S., Cuoci, A., Dixon, A. G., Maestri, M., Comput. Chem. Eng., 97 (2017) 175

# *Cell Agglomeration: Binning effect on axial profiles*



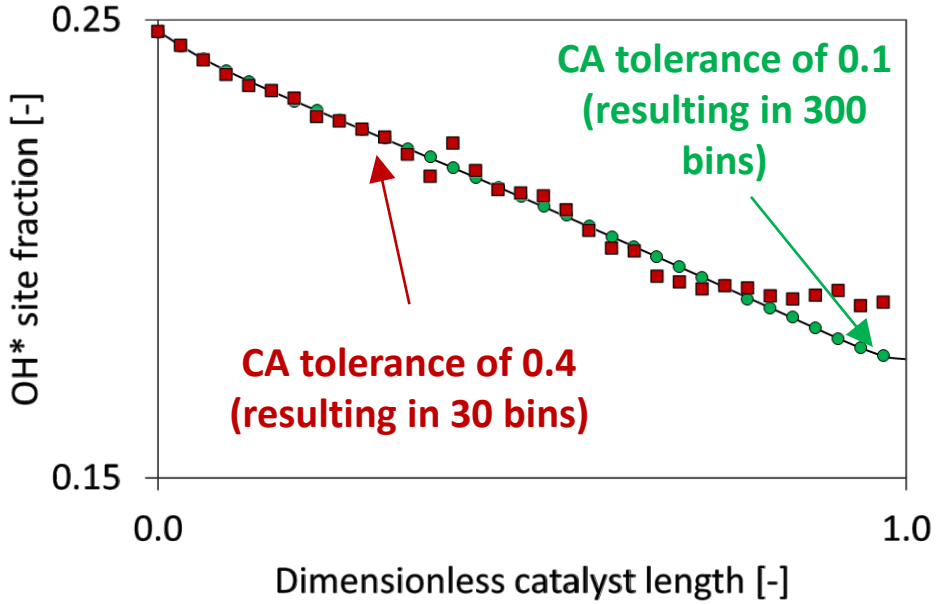

Rebughini, S., Cuoci, A., Dixon, A. G., Maestri, M., Comput. Chem. Eng., 97 (2017) 175

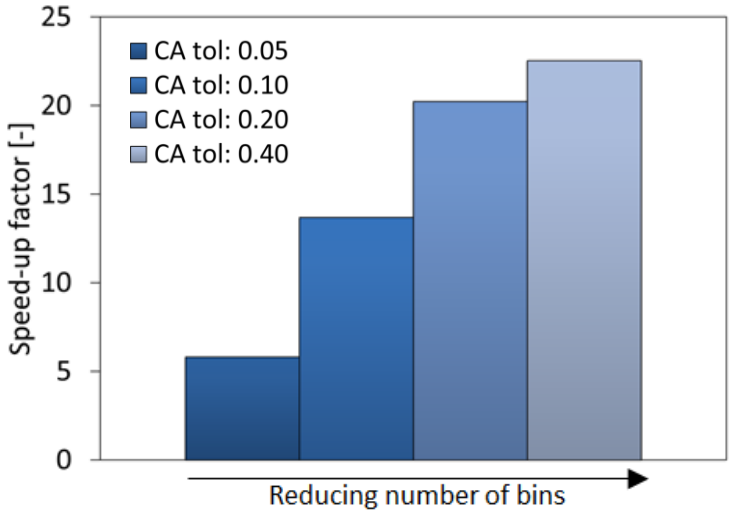
# *Cell Agglomeration: Binning effect on axial profiles*




Rebughini, S., Cuoci, A., Dixon, A. G., Maestri, M., Comput. Chem. Eng., 97 (2017) 175

# *Cell Agglomeration: Binning effect on axial profiles*

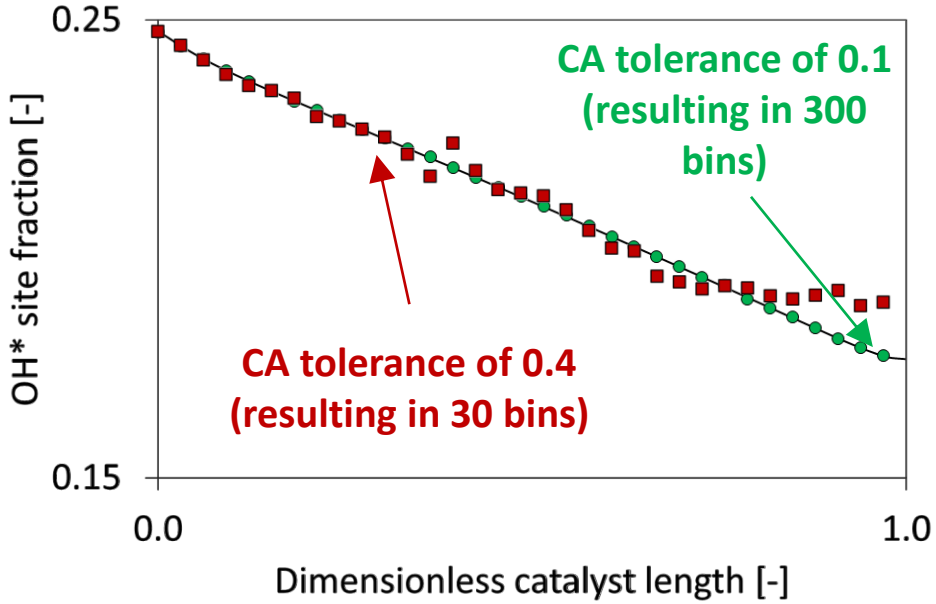


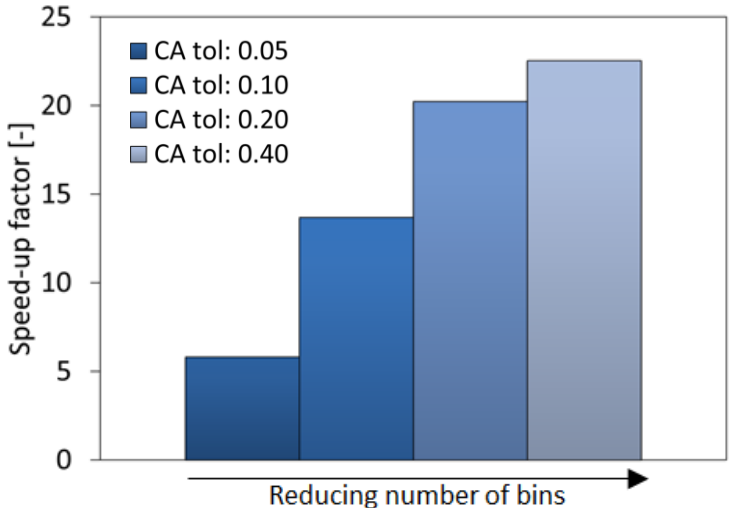

**Reducing the number of bins reduces the computational cost**

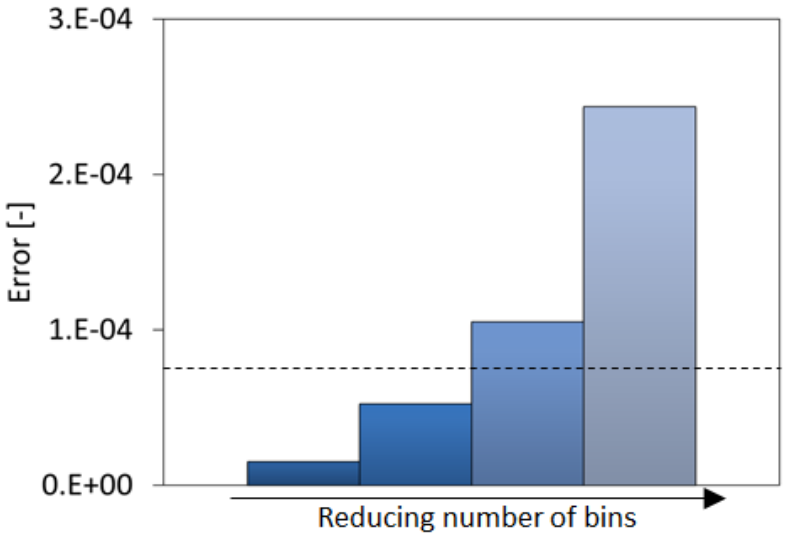
Rebughini, S., Cuoci, A., Dixon, A. G., Maestri, M., Comput. Chem. Eng., 97 (2017) 175

# *Cell Agglomeration: Binning effect on axial profiles*




**Reducing the number of bins reduces the computational cost,**
but **increases the error**



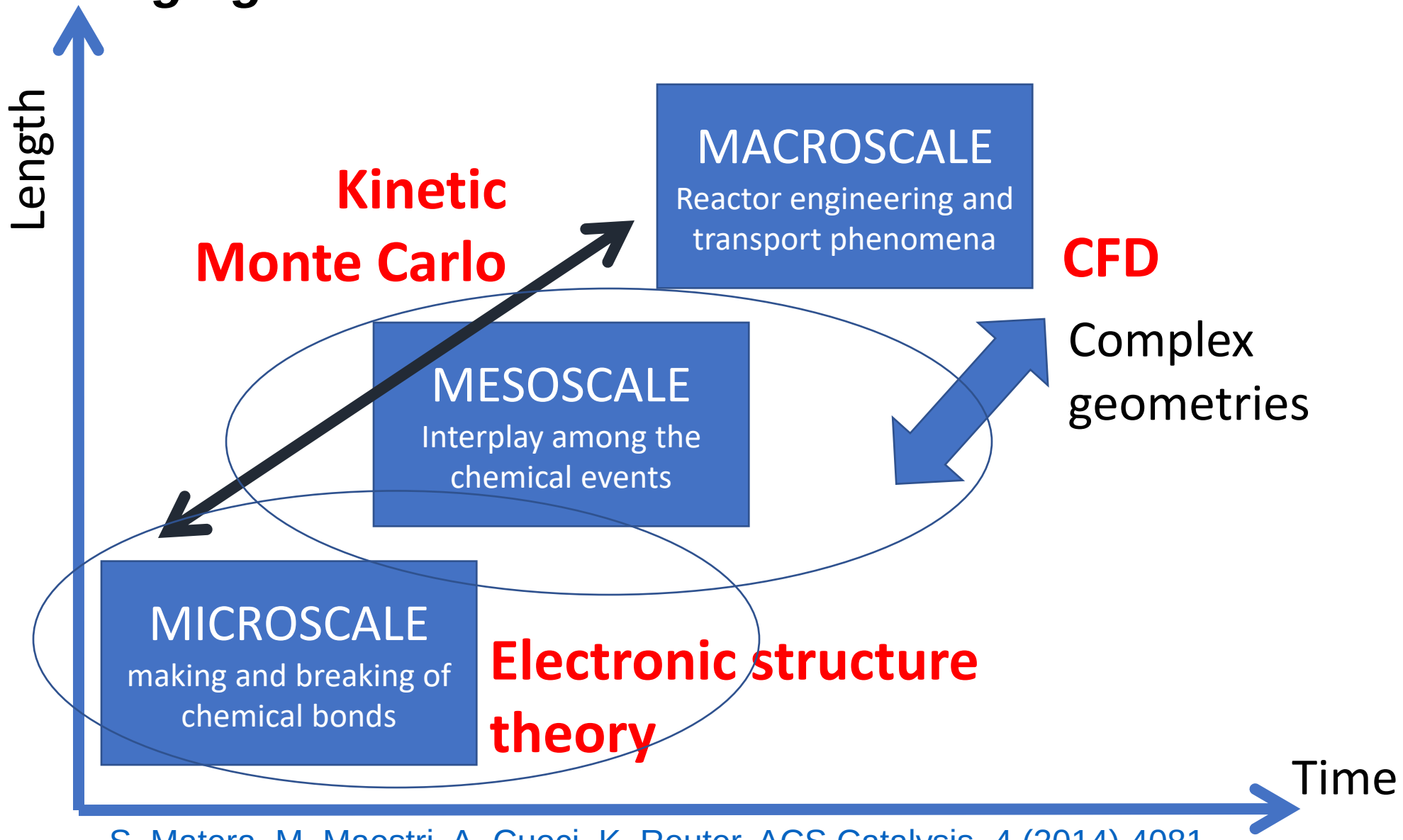
# *Pre-computing the chemistry*

Pre-computed chemistry is tabulated in order to easily accessible during simulation reducing computational effort:

- ✓ **Spline interpolation**
- ✓ **Random forest algorithm**

The knowledge of the «region of interest» is required

# *Bridging the scales: from micro- to macro-scale.*




S. Matera, M. Maestri, A. Cuoci, K. Reuter, ACS Catalysis, 4 (2014) 4081

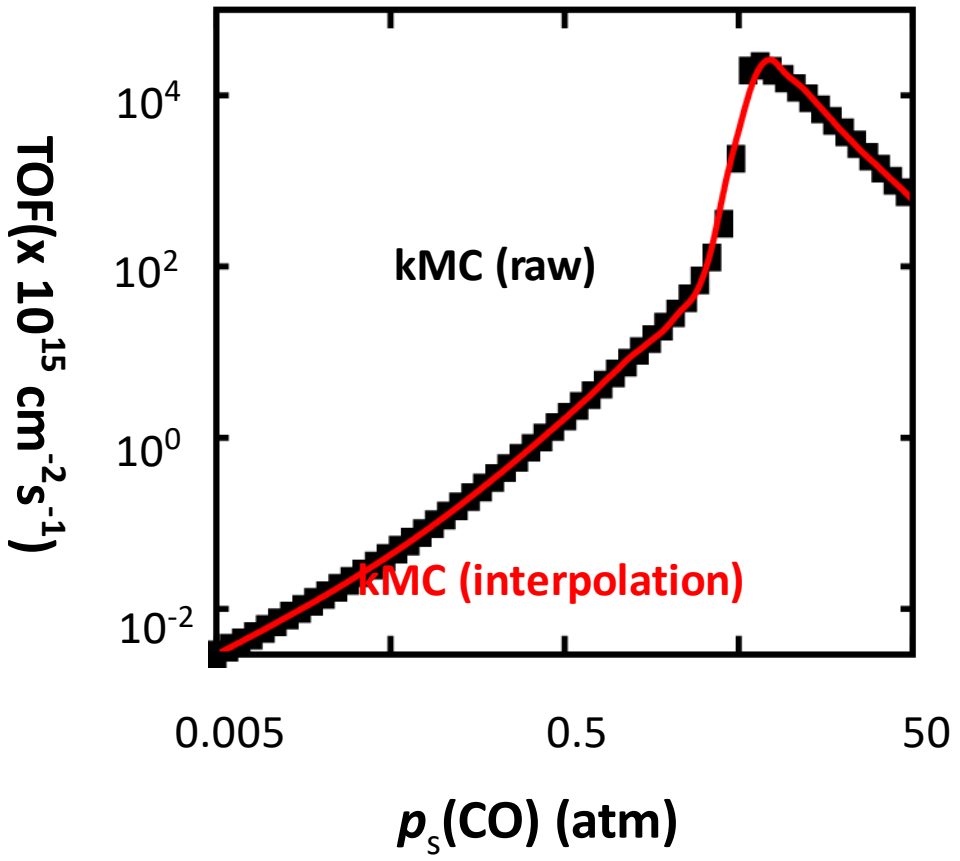
# *Effective decoupling of the scales*

- ✓ Continuum equations need boundary conditions for the mass fluxes $j^{\alpha}$ at the surface:

$$j_n^{\alpha} = \nu^{\alpha} M^{\alpha} \mathbf{TOF}$$

- ✓ **Coupled problem**: to determine the TOF with 1p-kMC the pressures at the surface are needed, but the pressure field depends on the TOF
- ✓ **kMC too expensive** for direct coupling to the flow solver
- ✓ **Run kMC beforehand and interpolate (Modified Shepard)**
- ✓ **The interpolated function will be then used during CFD**

**$T_s$ = 600K, $p_s(O_2)$= 1atm**

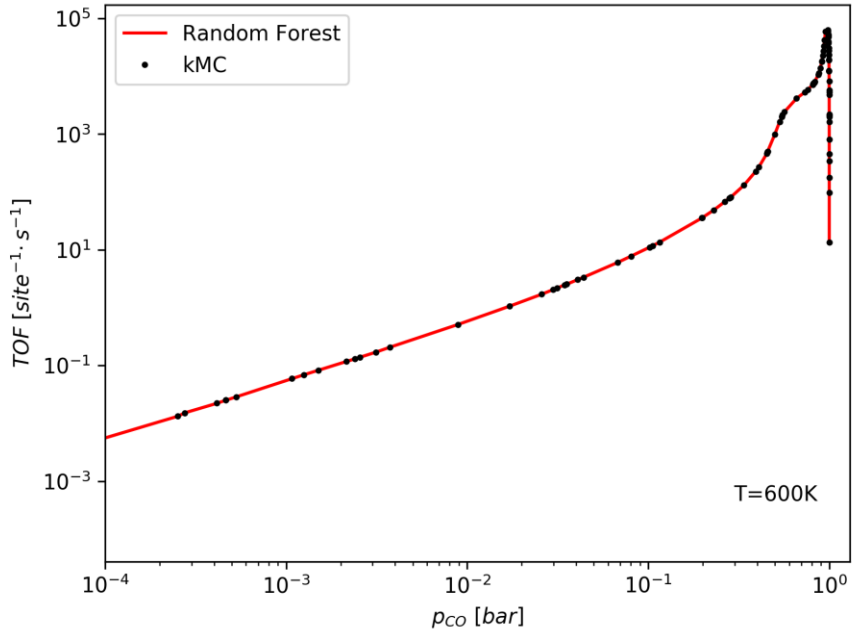


# *Effective decoupling of the scales*

- ✓ Continuum equations need boundary conditions for the mass fluxes $j^{\alpha}$ at the surface:

$$j_n^{\alpha} = \nu^{\alpha} M^{\alpha} \mathbf{TOF}$$

- ✓ **Coupled problem**: to determine the TOF with 1p-kMC the pressures at the surface are needed, but the pressure field depends on the TOF
- ✓ **kMC too expensive** for direct coupling to the flow solver
- ✓ **Run kMC beforehand and interpolate (Modified Shepard)**
- ✓ **The interpolated function will be then used during CFD**




**Tabulation and interpolation techniques**
**Machine learning**

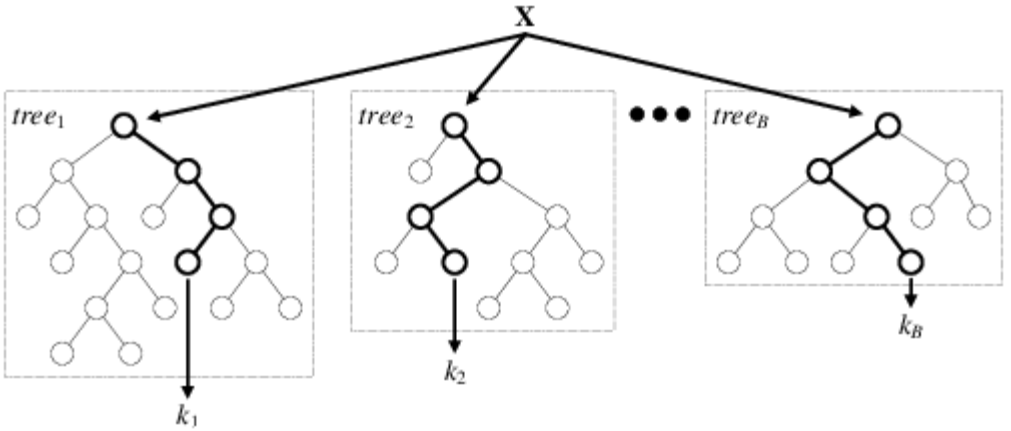
M. Bracconi and M. Maestri, Chemical Engineering Journal, 400 (2020) 125469

# Machine learning techniques to overcome current limitations

**Machine-learning techniques enable to tabulate pre-computed TOFs to obtain an efficient prediction [1]**

**A possible choice is Random Forest (RF) [2]**

- Ensemble of **regression trees**
- Single tree generated by **bagging**
- Capability in dealing with **high dimension dataset and big data**
- Provide **efficient tabulation and retrieval capability**
- Provide quantitative information of **variable importance**



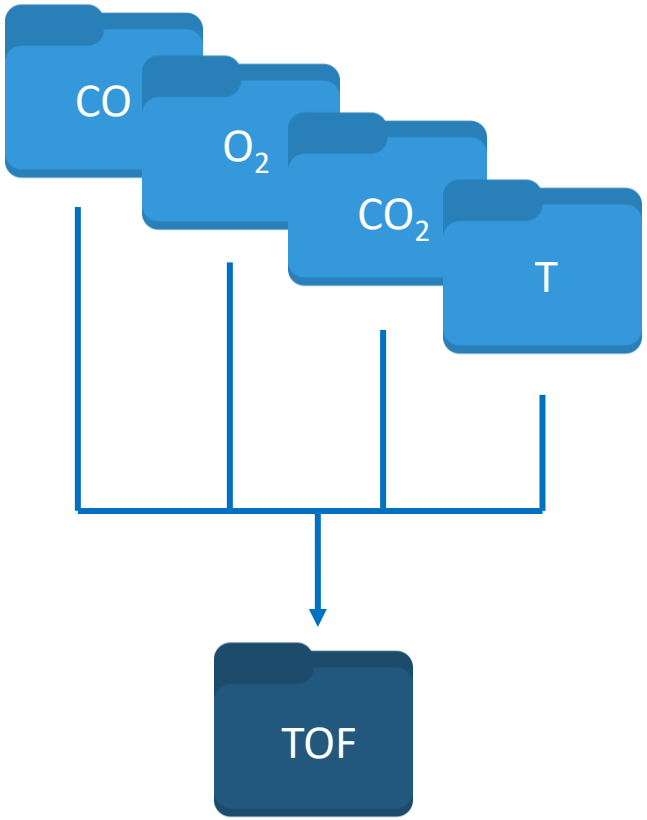

$$\hat{k} = \frac{1}{B}\sum_{i=1}^{B} k_i(\mathbf{X})$$

**Random Forest is a very good candidate for enabling the coupling with reactor models**


[1] Partoupur et al., Computers & Chemical Engineering (2018)
[2] Brieman, et al., "Cart: Classification and regression trees." (1995)

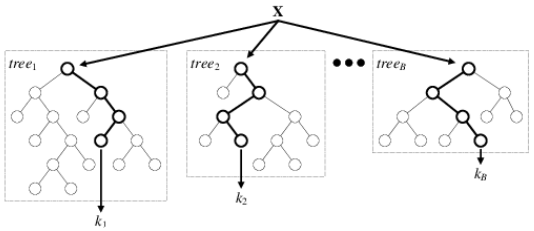
# Effective coupling between kMC and reactor models

## Raw data

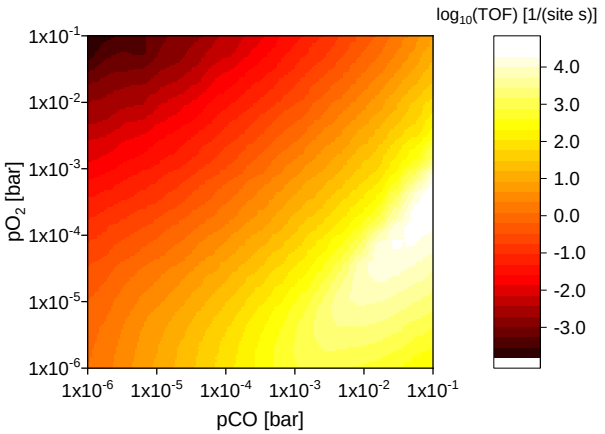



## Random Forest

Ensemble of decision trees trained with the raw data

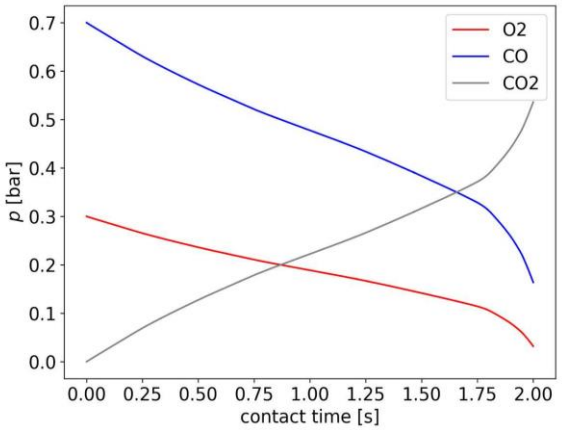



Continuous representation of TOF

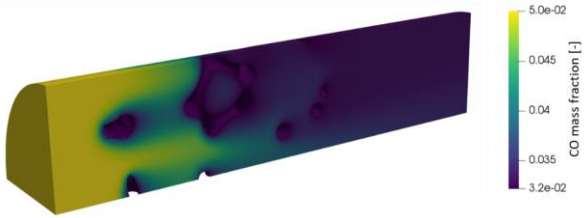



## Reactor models (e.g. PFR or CFD)

Species profiles along the reactor




CFD concentration maps



# Example: CO oxidation on $RuO_2$

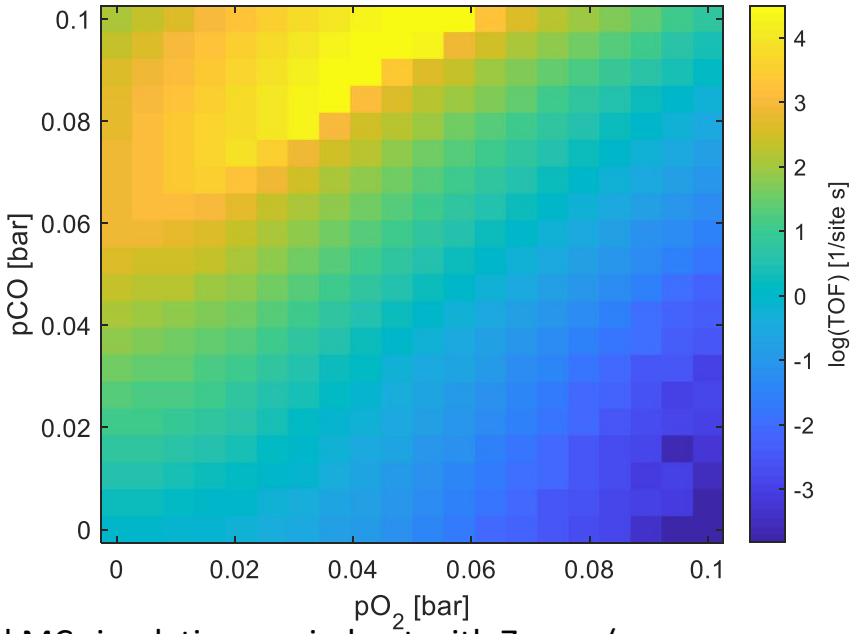
[1] Matera et al., ACS Catalysis (2014)

**kMC CO oxidation on $RuO_2$(110) [1]:**

- **Two species (2 directions)**
- **P = 1 bar – T = 600 K**
- **$CO_2$ net production rate is the target TOF**

| Operating conditions: | Range (mol/mol) |
|---|---|
| CO | $1{\cdot}10^{-6}$ - $1{\cdot}10^{-1}$ |
| $O_2$ | $1{\cdot}10^{-6}$ - $1{\cdot}10^{-1}$ |

**Evenly-spaced grid (log) with 20 points for each direction (400 training data)**




kMC simulation carried out with Zacros (www.zacros.org)
Stamatakis et al., Journal of Chemical Physics 134(21)

[1] Matera et al., ACS Catalysis (2014)

# Example: CO oxidation on $RuO_2$

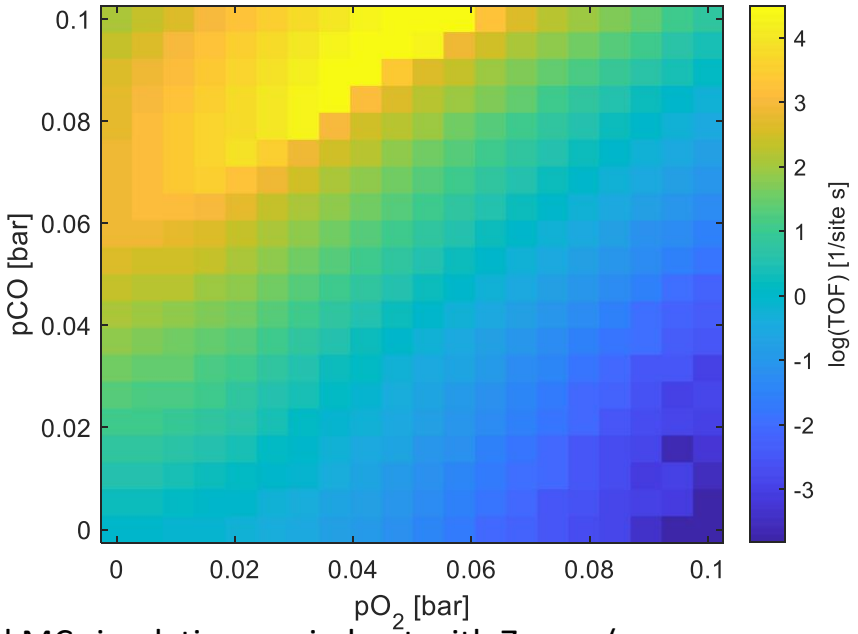
[1] Matera et al., ACS Catalysis (2014)

**kMC CO oxidation on $RuO_2$(110) [1]:**

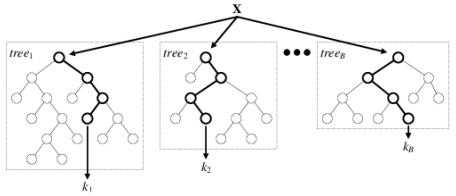
- **Two species (2 directions)**
- **P = 1 bar – T = 600 K**
- **$CO_2$ net production rate is the target TOF**

| Operating conditions: | Range (mol/mol) |
|---|---|
| CO | $1\cdot10^{-6}$ - $1\cdot10^{-1}$ |
| $O_2$ | $1\cdot10^{-6}$ - $1\cdot10^{-1}$ |

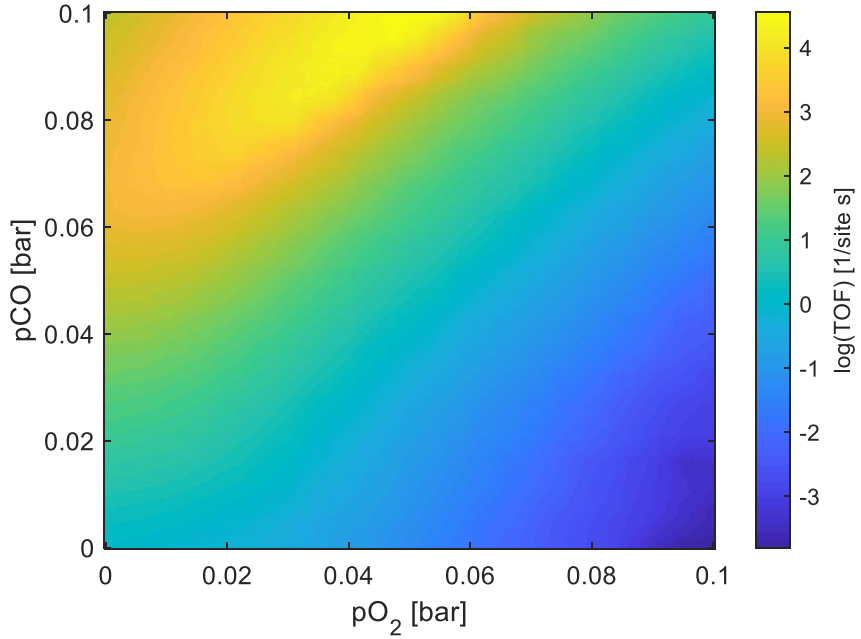
**Evenly-spaced grid (log) with 20 points for each direction (400 training data)**




**Raw datapoints train a Random Forest**




**Continuous representation of the TOF**



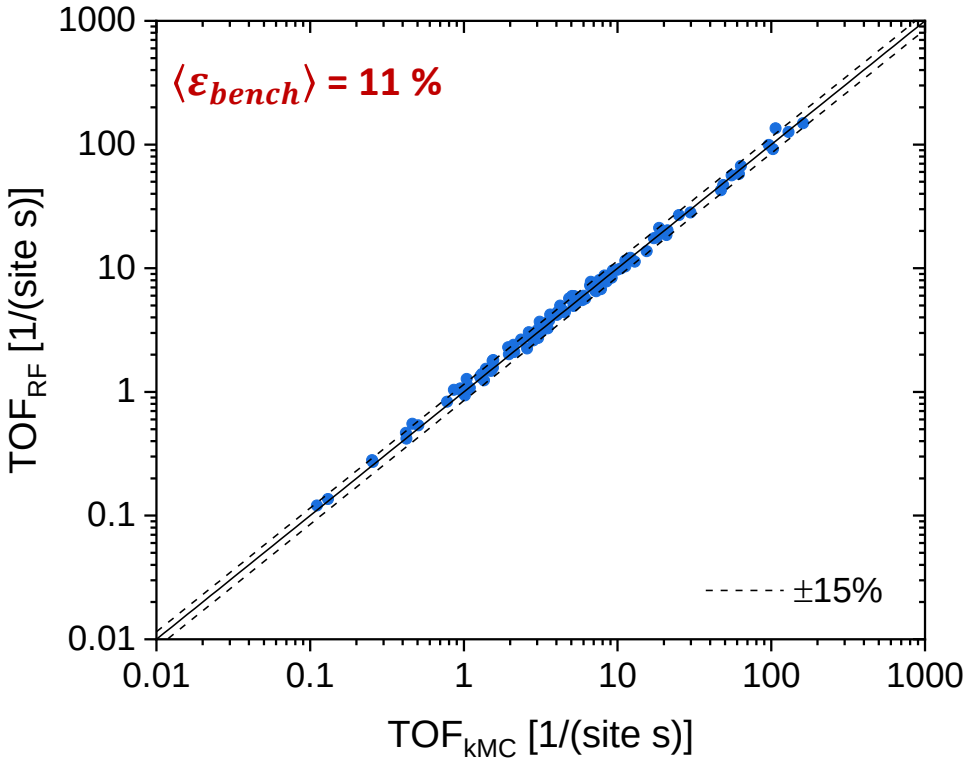

kMC simulation carried out with Zacros (www.zacros.org)
Stamatakis et al., Journal of Chemical Physics 134(21)

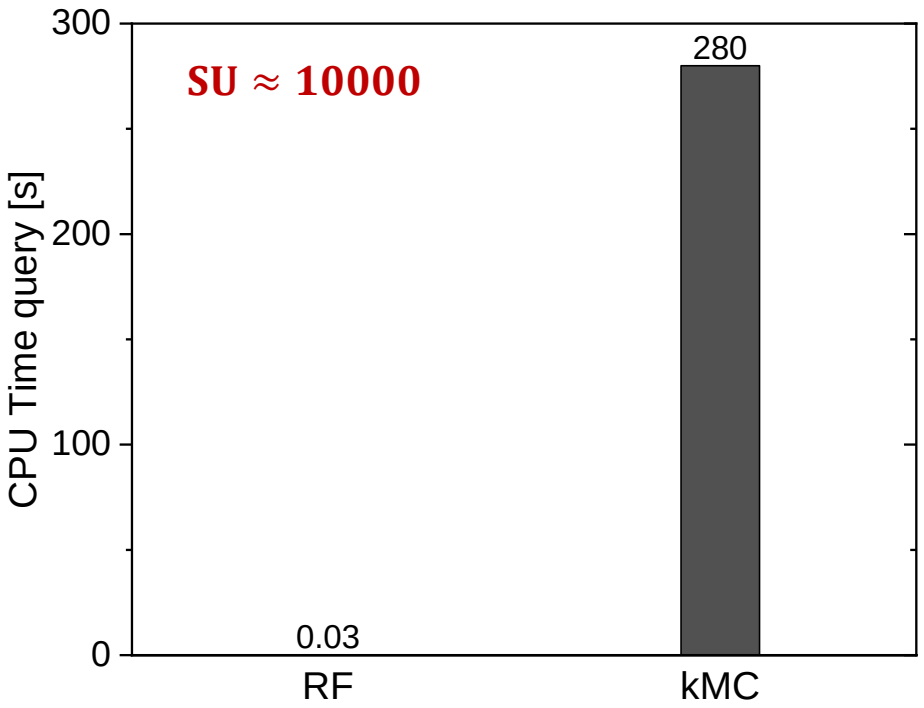
[1] Matera et al., ACS Catalysis (2014)

# Example: CO oxidation on $RuO_2$

**kMC CO oxidation on $RuO_2$(110) [1]:**

- **Accuracy tested by comparing the RF prediction against kMC computed values**
- **Speed-up evaluated comparing the CPU employed for computing a TOF with kMC and RF**



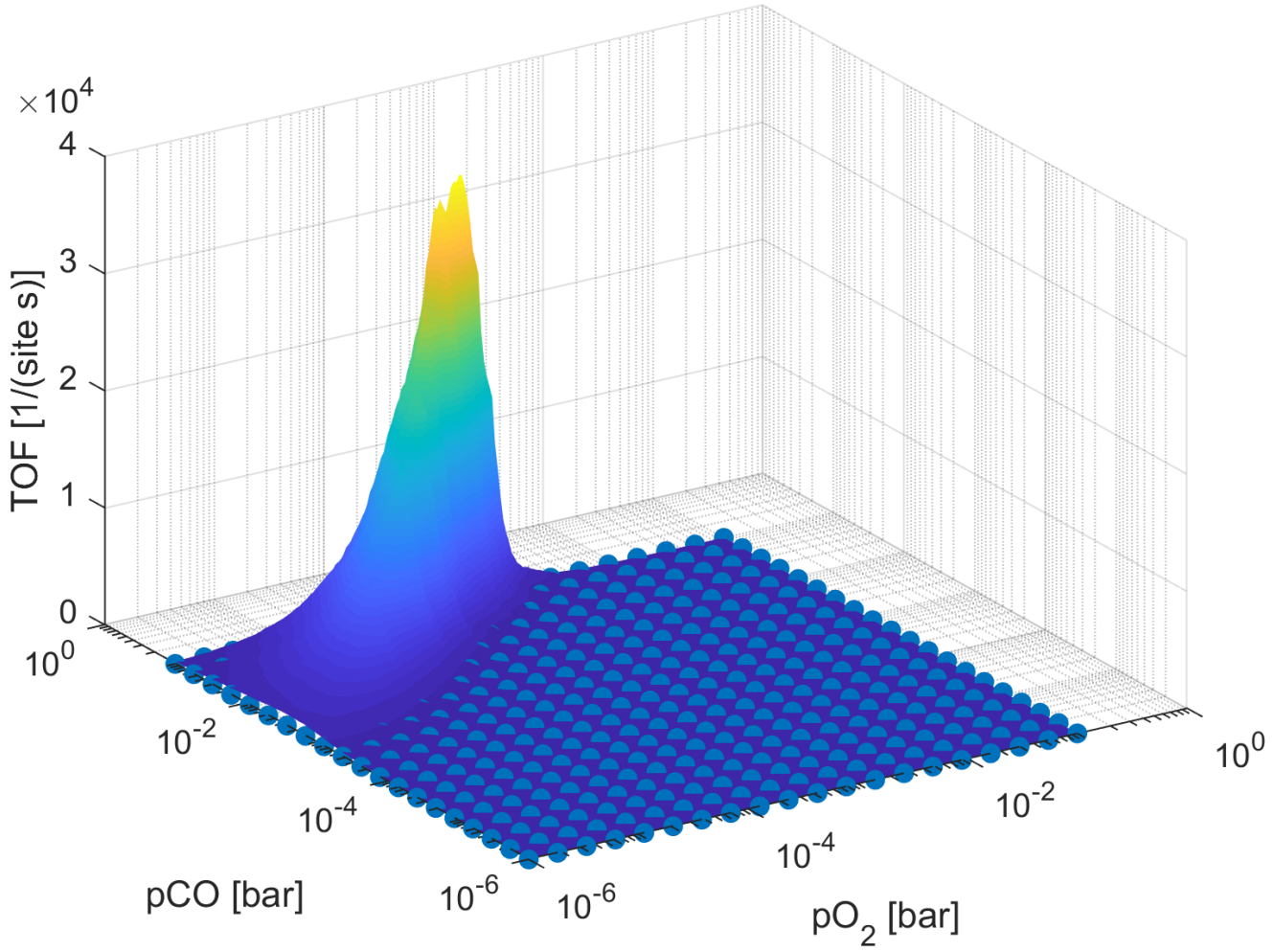




$$\langle \varepsilon_{bench} \rangle = \frac{1}{N_Q} \sum_{i=0}^{N_Q} \left| \frac{TOF_{kMC}(\boldsymbol{x}_i) - TOF_{RF}(\boldsymbol{x}_i)}{TOF_{kMC}(\boldsymbol{x}_i)} \right|$$

$$\mathrm{SU} = \frac{1}{N_Q} \sum_{i=0}^{N_Q} \frac{t_{kMC\,i}}{t_{RF\,i}}$$

[1] Matera et al., ACS Catalysis (2014)

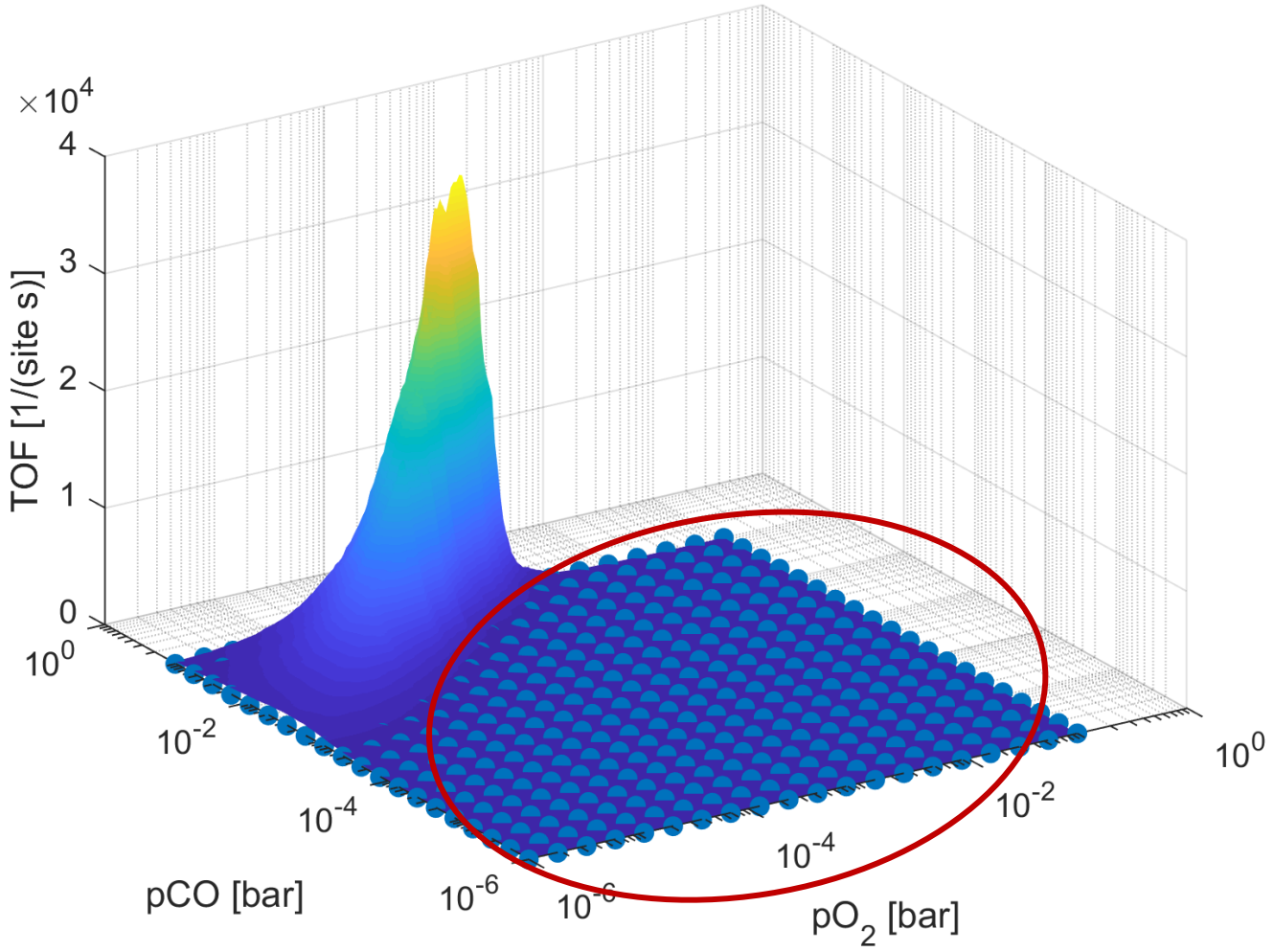
# Are we properly defining the training set?




- **Most of the training points are placed in region of the composition space where they are not necessary to properly describe the function**
- **Evaluation of the training points in those regions is a waste of computational time**
- **Systems characterized by large dimensionality require a huge amount of training points**

M. Bracconi and M. Maestri, Chemical Engineering Journal, 400 (2020) 125469

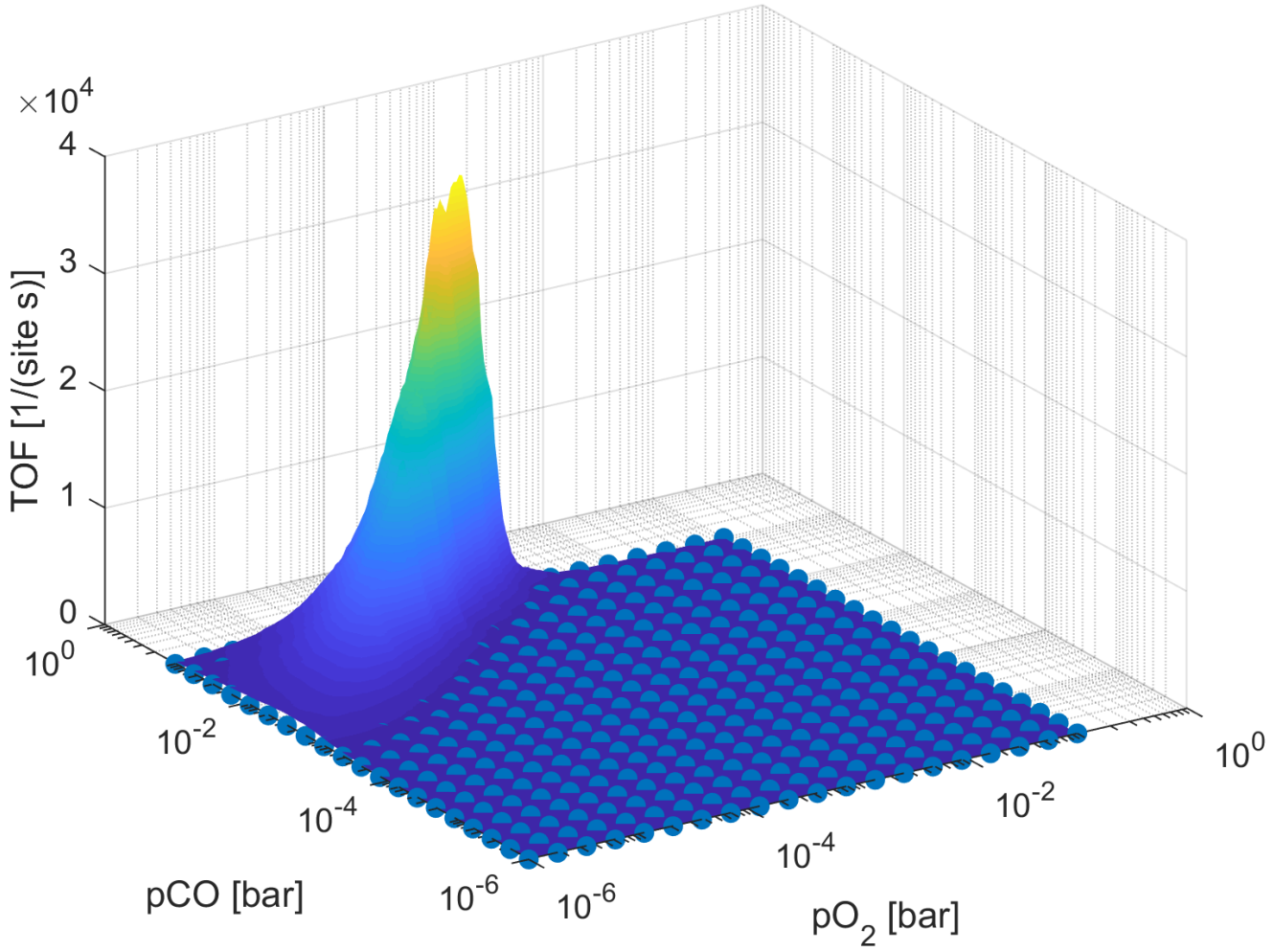
# Are we properly defining the training set?




- **Most of the training points are placed in region of the composition space where they are not necessary to properly describe the function**
- **Evaluation of the training points in those regions is a waste of computational time**
- **Systems characterized by large dimensionality require a huge amount of training points**

M. Bracconi and M. Maestri, Chemical Engineering Journal, 400 (2020) 125469

# Are we properly defining the training set?



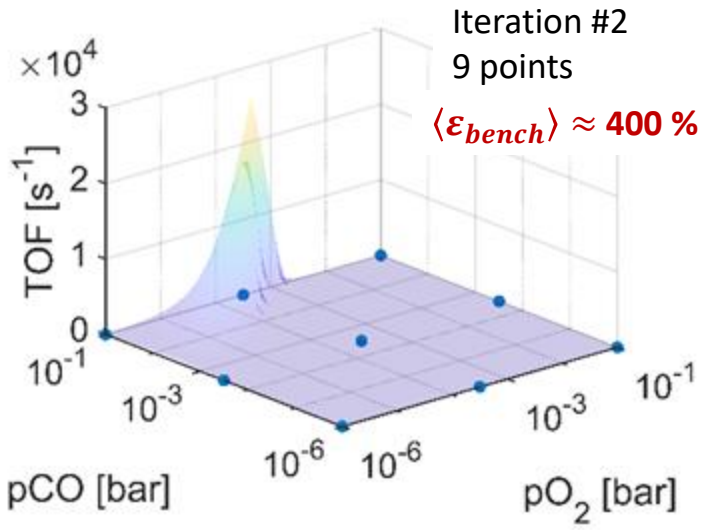

- **Most of the training points are placed in region of the composition space where they are not necessary to properly describe the function**
- **Evaluation of the training points in those regions is a waste of computational time**
- **Systems characterized by large dimensionality require a huge amount of training points**

**Proper definition of the training points is crucial to improve accuracy and efficiency of the approach**

M. Bracconi and M. Maestri, Chemical Engineering Journal, 400 (2020) 125469

# Advanced design of the training points: CO oxidation on $RuO_2$

- **Sequential addition of training points where required to improve the accuracy of the approximation and to minimize the number of computationally intensive evaluation of the kMC model**




| Number of points | |
|---|---|
| CO | O2 |
| 3 | 3 |

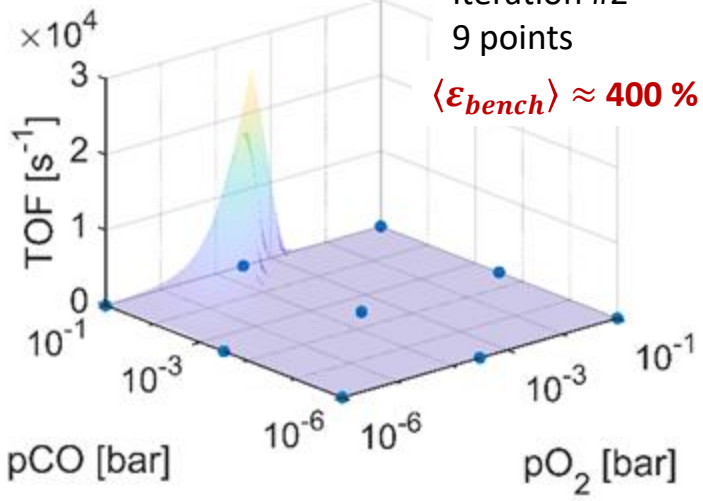
M. Bracconi and M. Maestri, Chemical Engineering Journal, 400 (2020) 125469

# Advanced design of the training points: CO oxidation on $RuO_2$

- **Sequential addition of training points where required to improve the accuracy of the approximation and to minimize the number of computationally intensive evaluation of the kMC model**

Iteration #2
9 points

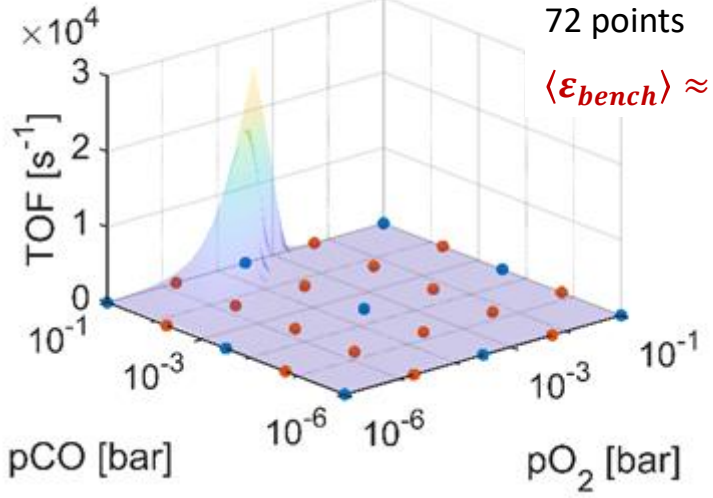
$\langle \varepsilon_{bench} \rangle \approx 400\ \%$




| Number of points | |
|---|---|
| CO | O2 |
| 3 | 3 |

Iteration #2
72 points

$\langle \varepsilon_{bench} \rangle \approx 40\ \%$

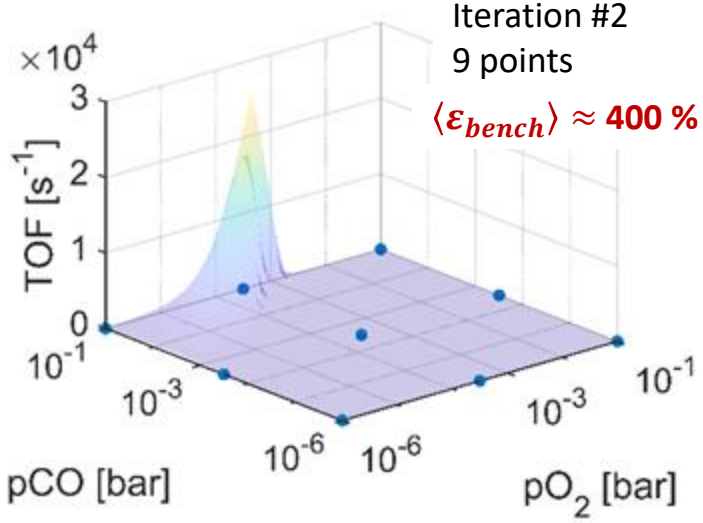



| Number of points | |
|---|---|
| CO | O2 |
| 3 | 5 |

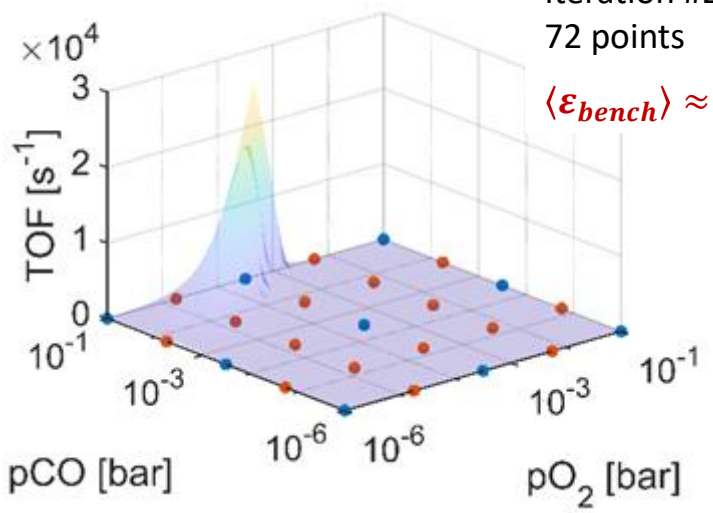
M. Bracconi and M. Maestri, Chemical Engineering Journal, 400 (2020) 125469

# Advanced design of the training points: CO oxidation on $RuO_2$

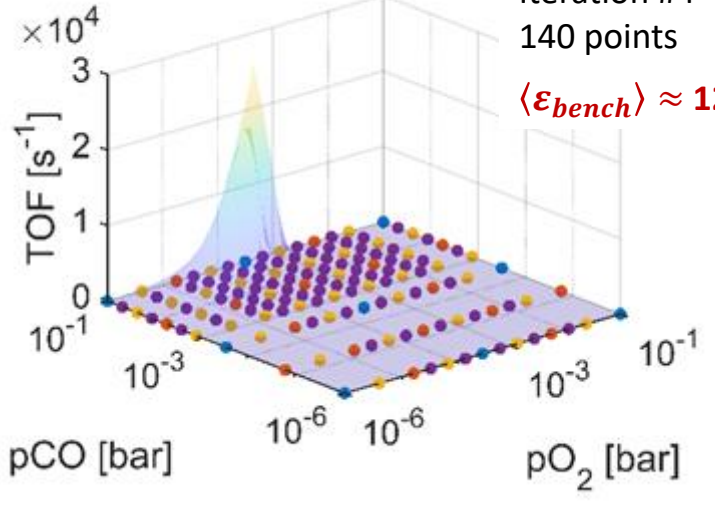
- **Sequential addition of training points where required to improve the accuracy of the approximation and to minimize the number of computationally intensive evaluation of the kMC model**




| Number of points | |
|---|---|
| CO | O2 |
| 3 | 3 |




| Number of points | |
|---|---|
| CO | O2 |
| 3 | 5 |

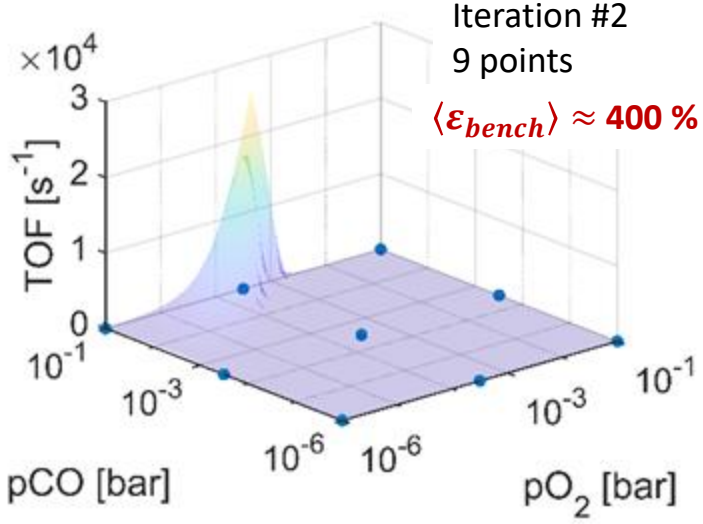



| Number of points | |
|---|---|
| CO | O2 |
| 10 | 14 |

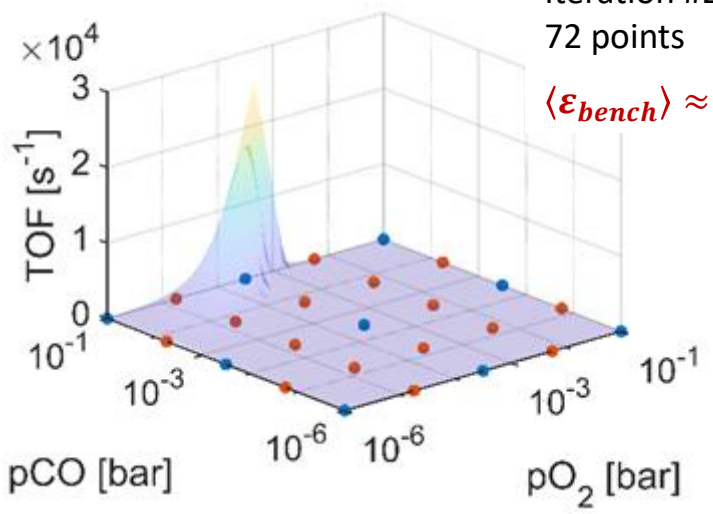
M. Bracconi and M. Maestri, Chemical Engineering Journal, 400 (2020) 125469

# Advanced design of the training points: CO oxidation on $RuO_2$

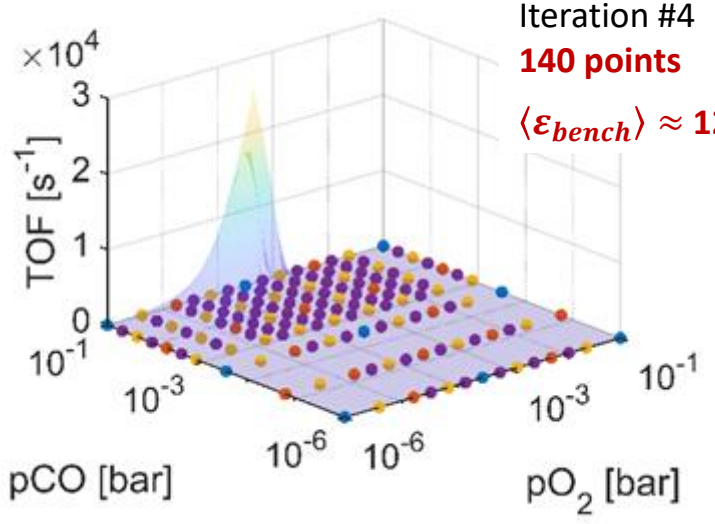
- **Sequential addition of training points where required to improve the accuracy of the approximation and to minimize the number of computationally intensive evaluation of the kMC model**




| Number of points | |
|---|---|
| CO | O2 |
| 3 | 3 |




| Number of points | |
|---|---|
| CO | O2 |
| 3 | 5 |




| Number of points | |
|---|---|
| CO | O2 |
| 10 | 14 |

**Analogous accuracy by using much less points**

# Advanced design of the training points: WGS on Rh

**Water gas shift reaction network [1]:**

- **Four species plus temperature (5 directions)**
- **$CO_2$ net production rate is the tabulated TOF**
- **Target accuracy 2.5% (benchmark error)**

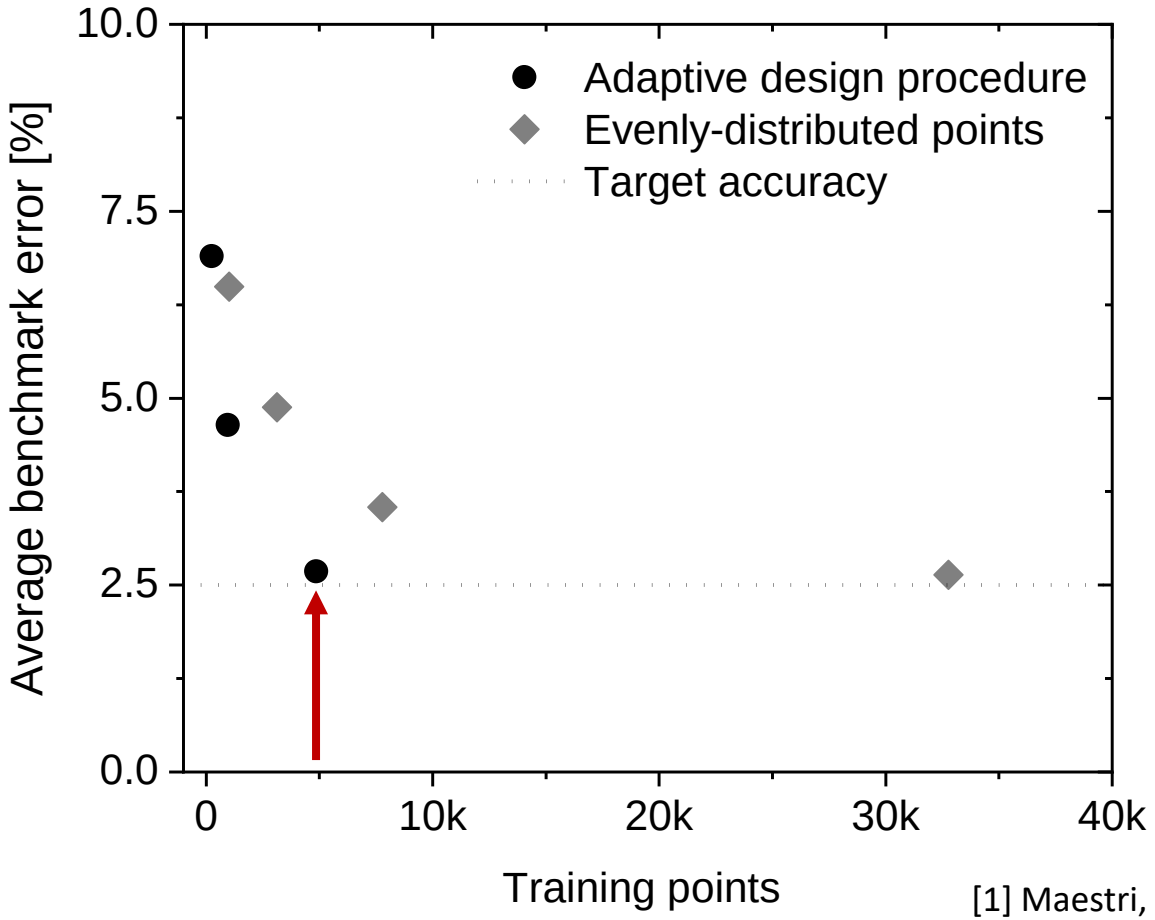
| Operating conditions: | Range (mol/mol) | | Range (mol/mol) |
|---|---|---|---|
| CO | $1\cdot10^{-2}$ - $1\cdot10^{-1}$ | H2 | $1\cdot10^{-6}$ - $1\cdot10^{-3}$ |
| H2O | $1\cdot10^{-2}$ - $1\cdot10^{-1}$ | CO2 | $1\cdot10^{-6}$ - $1\cdot10^{-3}$ |
| | | T | 650 – 900 K |

[1] Maestri, et al. AIChE Journal 55 (2009)

# Advanced design of the training points: WGS on Rh

**Water gas shift reaction network [1]:**

- **Four species plus temperature (5 directions)**
- **$CO_2$ net production rate is the tabulated TOF**
- **Target accuracy 2.5% (benchmark error)**




| Operating conditions: | Range (mol/mol) | | Range (mol/mol) |
|---|---|---|---|
| CO | $1{\cdot}10^{-2}$ - $1{\cdot}10^{-1}$ | H2 | $1{\cdot}10^{-6}$ - $1{\cdot}10^{-3}$ |
| H2O | $1{\cdot}10^{-2}$ - $1{\cdot}10^{-1}$ | CO2 | $1{\cdot}10^{-6}$ - $1{\cdot}10^{-3}$ |
| | | T | 650 – 900 K |

- **Dataset is iteratively refined along the direction of higher importance**

| Final number of points | | | | |
|---|---|---|---|---|
| CO | $H_2O$ | H2 | $CO_2$ | T |
| 6 | 6 | 5 | 3 | 9 |

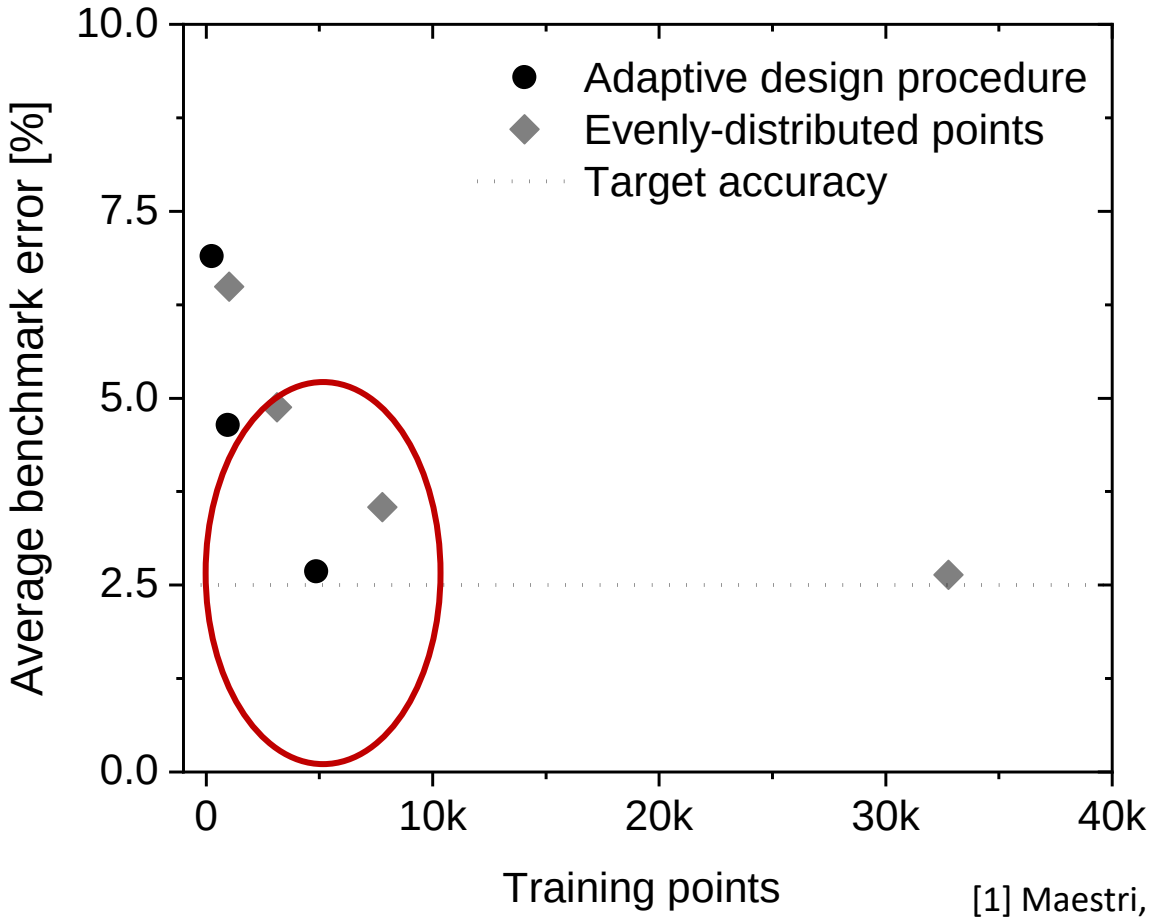
- **Target accuracy is reached by using 4860 points**

[1] Maestri, et al. AIChE Journal 55 (2009)

# Advanced design of the training points: WGS on Rh

**Water gas shift reaction network [1]:**

- **Four species plus temperature (5 directions)**
- **$CO_2$ net production rate is the tabulated TOF**
- **Target accuracy 2.5% (benchmark error)**




| Operating conditions: | Range (mol/mol) | | Range (mol/mol) |
|---|---|---|---|
| CO | $1{\cdot}10^{-2}$ - $1{\cdot}10^{-1}$ | H2 | $1{\cdot}10^{-6}$ - $1{\cdot}10^{-3}$ |
| H2O | $1{\cdot}10^{-2}$ - $1{\cdot}10^{-1}$ | CO2 | $1{\cdot}10^{-6}$ - $1{\cdot}10^{-3}$ |
| | | T | 650 – 900 K |

- **Dataset is iteratively refined along the direction of higher importance**

| Final number of points | | | | |
|---|---|---|---|---|
| CO | $H_2O$ | H2 | $CO_2$ | T |
| 6 | 6 | 5 | 3 | 9 |

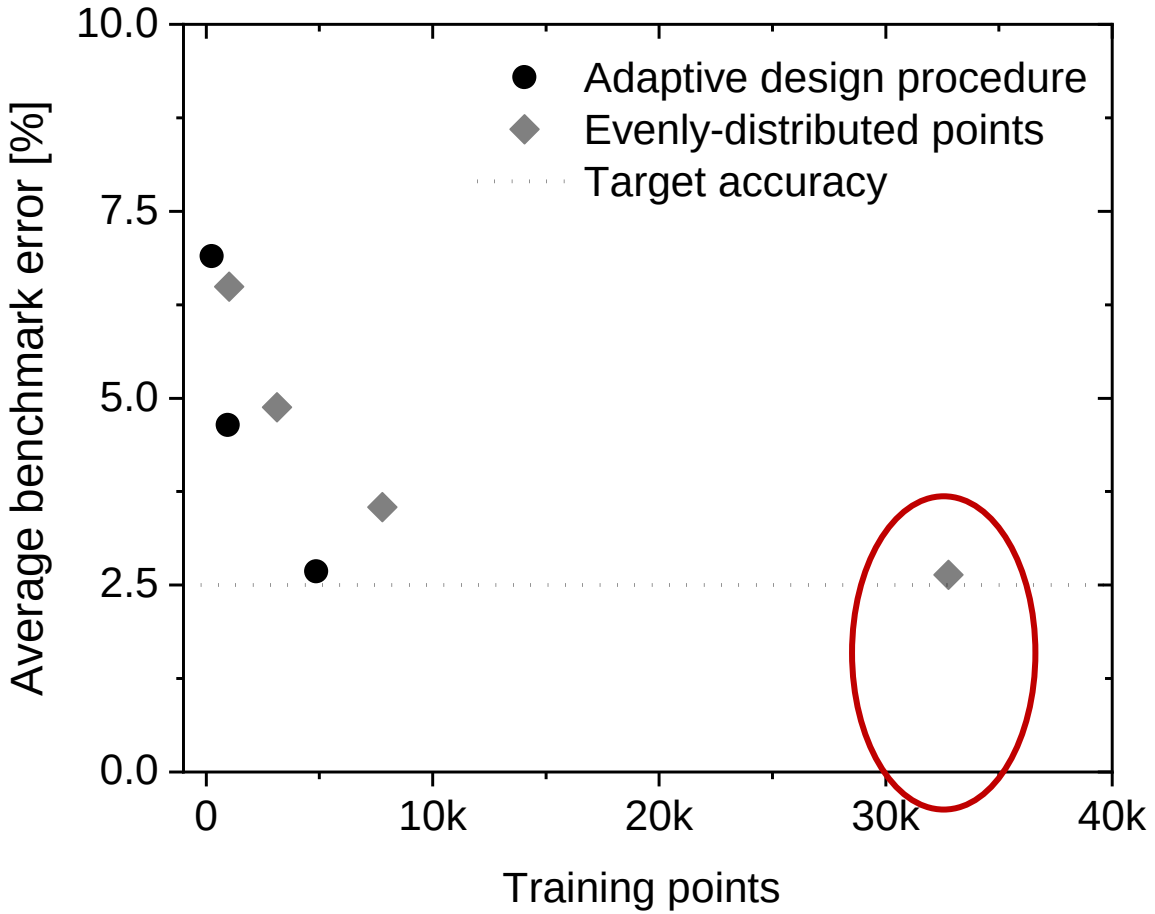
- **Target accuracy is reached by using ~60k points**
- **An evenly-distributed grid with the same number of points provide circa twice the benchmark error**

[1] Maestri, et al. AIChE Journal 55 (2009)

# Advanced design of the training points: WGS on Rh

**Water gas shift reaction network:**

- **Four species plus temperature (5 directions)**
- **$CO_2$ net production rate is the tabulated TOF**
- **Target accuracy 2.5% (benchmark error)**



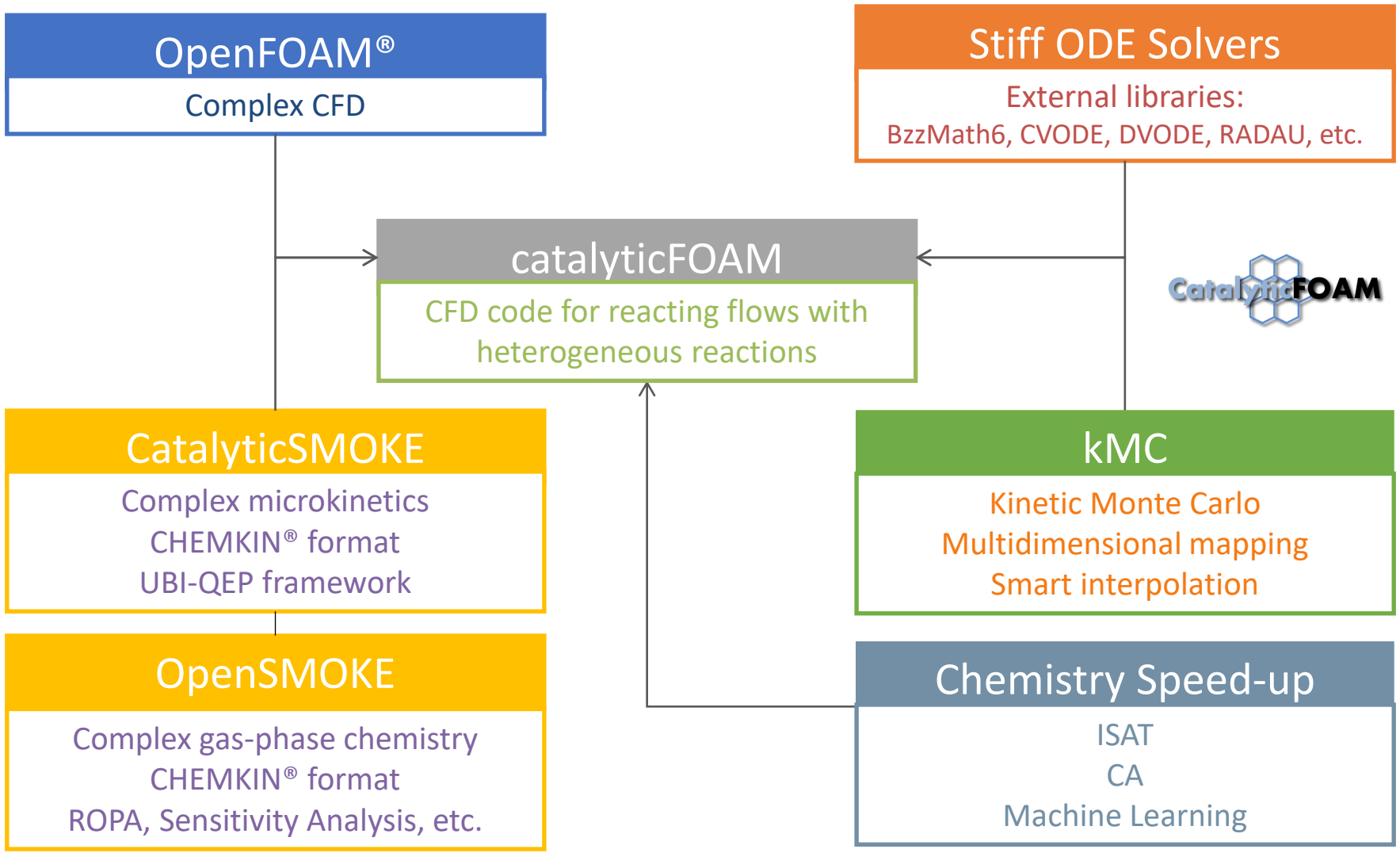

| Operating conditions: | Range (mol/mol) | | Range (mol/mol) |
|---|---|---|---|
| CO | $1\cdot10^{-2}$ - $1\cdot10^{-1}$ | H2 | $1\cdot10^{-6}$ - $1\cdot10^{-3}$ |
| H2O | $1\cdot10^{-2}$ - $1\cdot10^{-1}$ | CO2 | $1\cdot10^{-6}$ - $1\cdot10^{-3}$ |
| | | T | 650 – 900 K |

- **Dataset is iteratively refined along the direction of higher importance**

| Final number of points | | | | |
|---|---|---|---|---|
| CO | $H_2O$ | H2 | $CO_2$ | T |
| 6 | 6 | 5 | 3 | 9 |

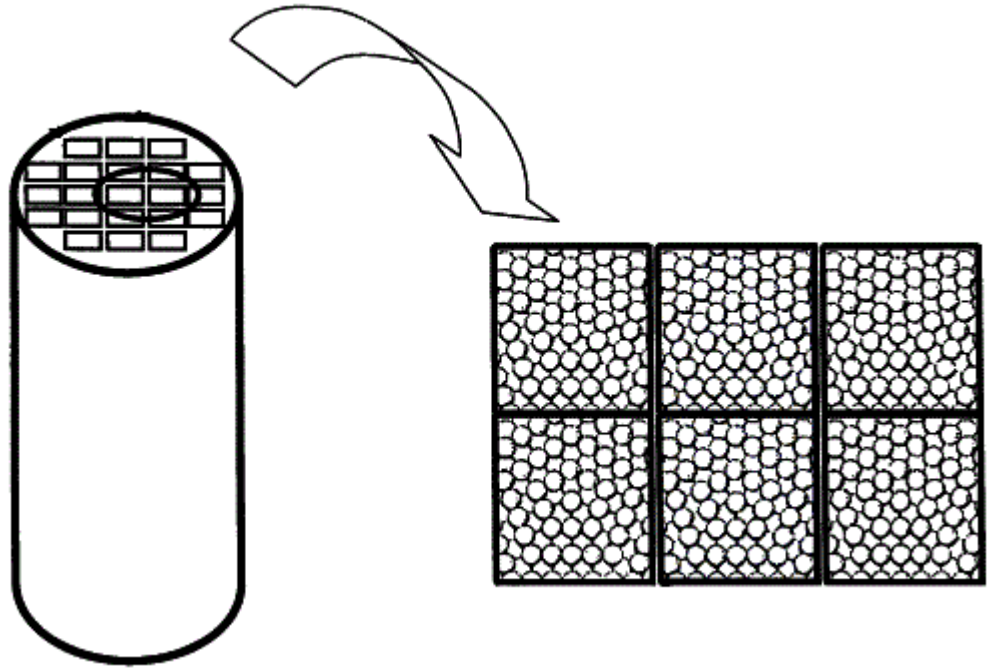
- **Target accuracy is reached by using ~60k points**
- **An evenly-distributed grid with the same number of points provide circa twice the benchmark error**
- **An evenly-distributed grid with more than four times the number of points (~32k points) is required for reaching the same error**

# *catalyticFOAM structure*




**www.catalyticfoam.polimi.it**

# *Micro packed bed reactors*



**Packed bed**

High catalytic area per unit volume

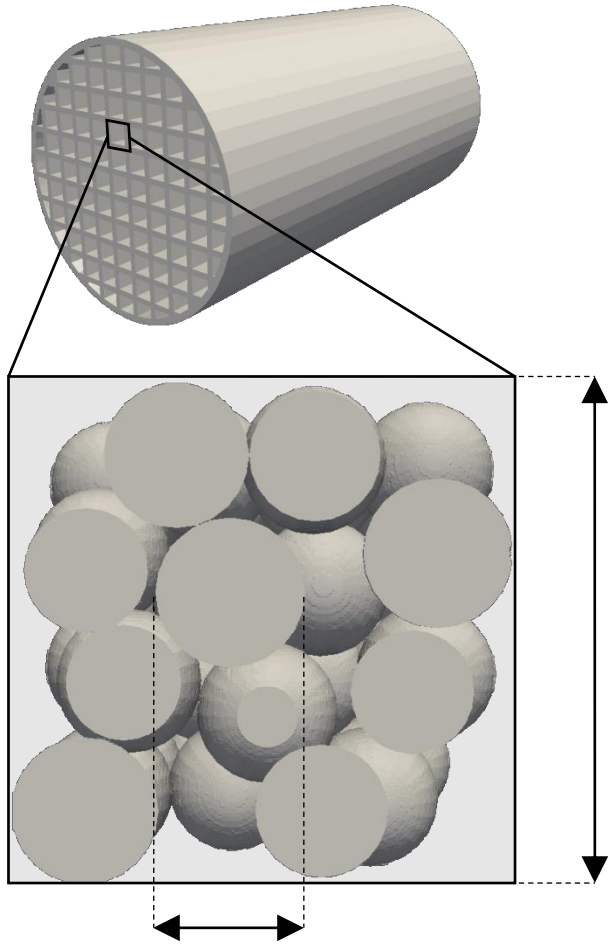
**Honeycomb**

High radial heat transfer within the honeycomb matrix

Vervloet, D., Kapteijn, F., Nijenhuis, J., van Ommen, J.R., 2013. Catal. Today 216, 111-116.

M. Iovane, R. Zennaro, P. Forzatti, G. Groppi, L. Lietti, E. Tronconi, C. G. Visconti, S. Rossini, and E. Mignone, ed: US Patent 20,120,184,631, 2012.

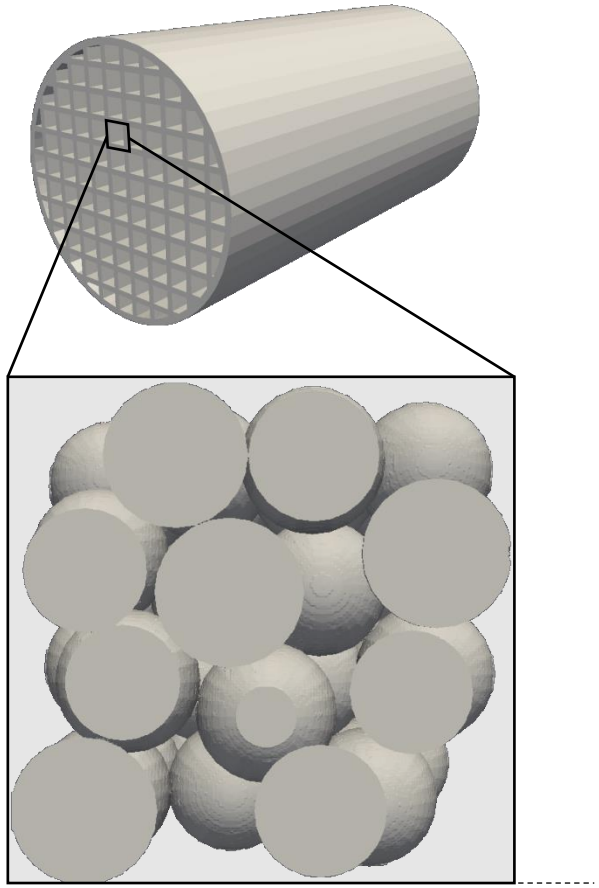
# *Micro packed bed reactors*




**Unconventional dimension**

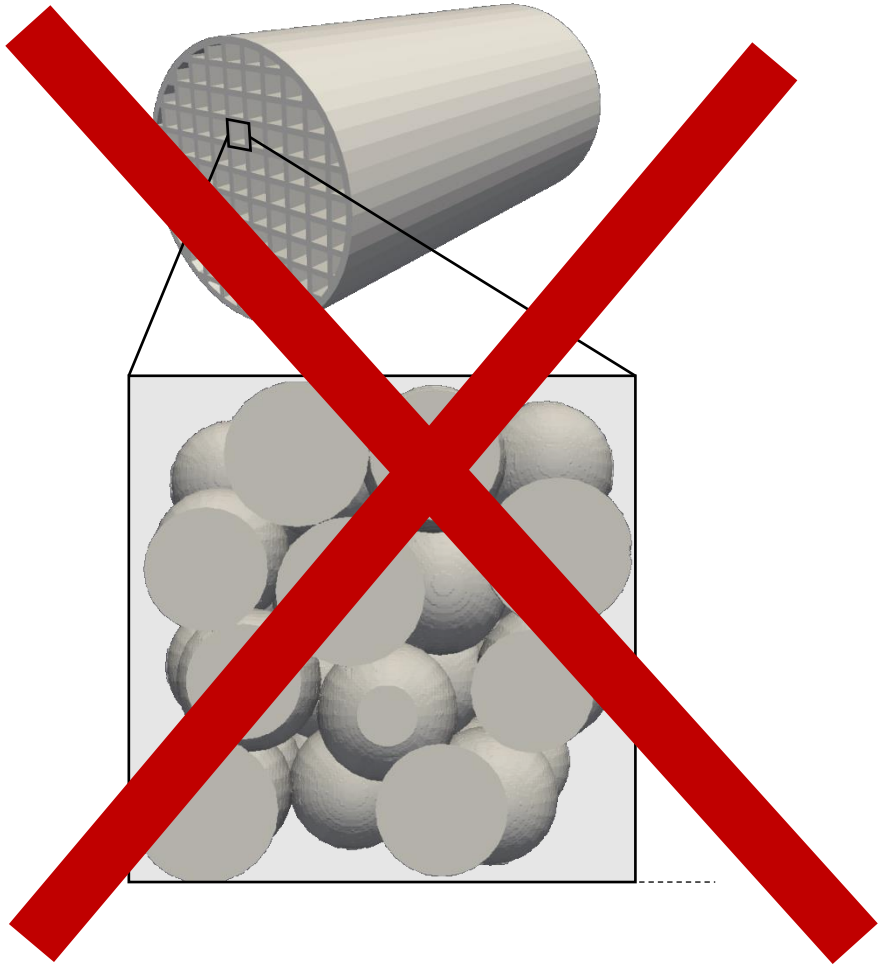
- Low Reynolds number
- Low tube-to-particle diameter ratio
- Square section (not common in packed bed reactors)

# *"Full scale" simulation is impractical*



| | |
|---|---|
| Number of channels | 25-100 |
| Number of spheres per channel | 300-1300 |
| Total number of spheres | **7500-13000** |
| Number of cells | **$10^8$-$10^9$** |

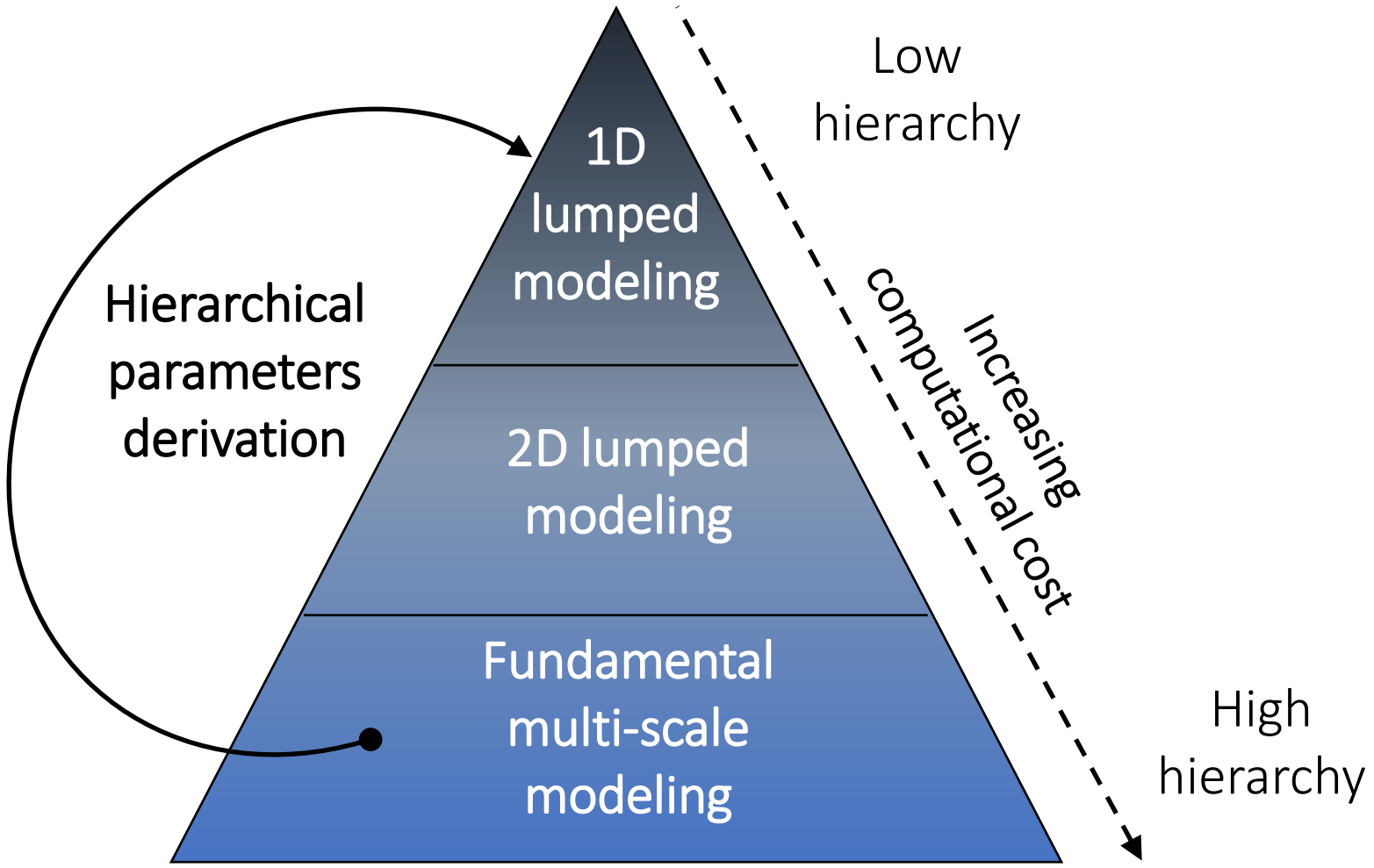
# *"Full scale" simulation is impractical*



| | |
|---|---|
| Number of channels | 25-100 |
| Number of spheres per channel | 300-1300 |
| Total number of spheres | **7500-13000** |
| Number of cells | **$10^8$-$10^9$** |

Detailed simulation of the **whole micro packed bed reactor** is impractical in terms of CPU hours

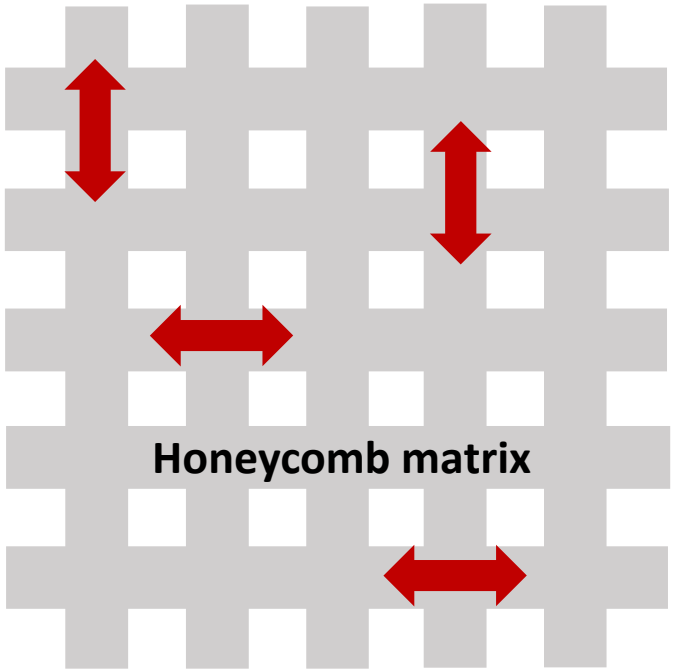
# *Hierarchical modeling* for reactor analysis



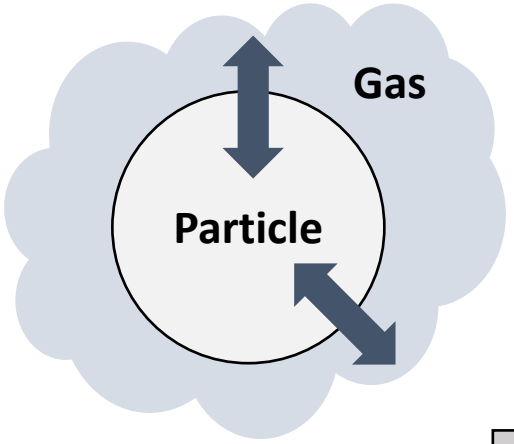

S. Rebughini, A. Cuoci, M. Maestri, Chem. Eng. J., 2016

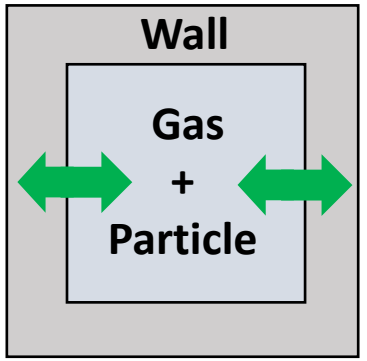
# *Hierarchical analysis of micro packed beds*

**Heat transfer in the honeycomb matrix**




**Gas-to-particle heat and mass transfer**

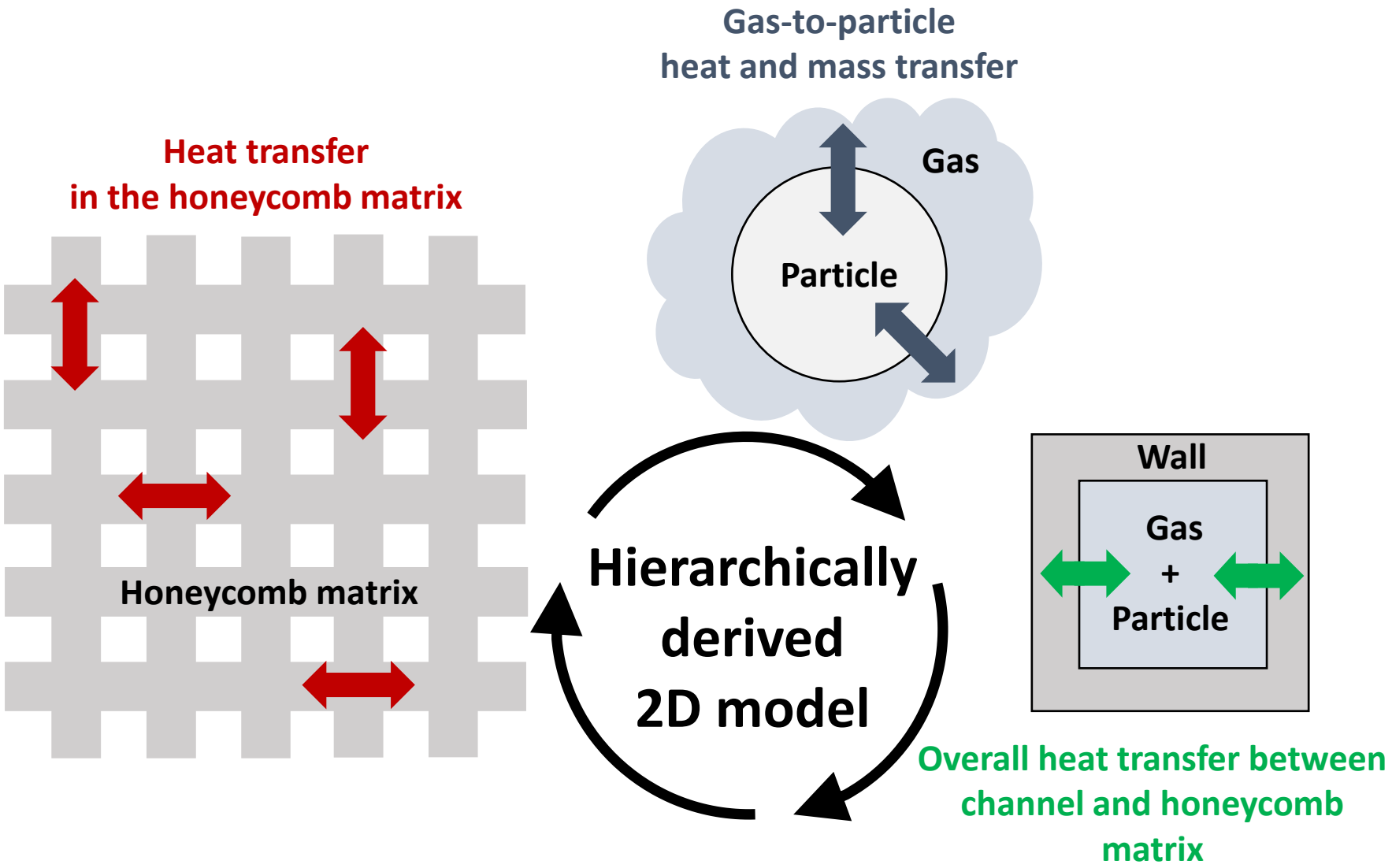



**Overall heat transfer between channel and honeycomb matrix**

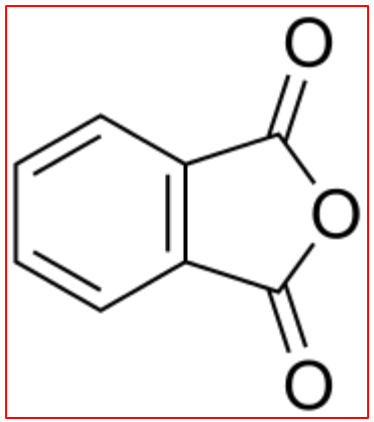
S. Rebughini, A. Cuoci, M. Maestri, Chem. Eng. J., 2016

# *Hierarchical analysis of micro packed beds*

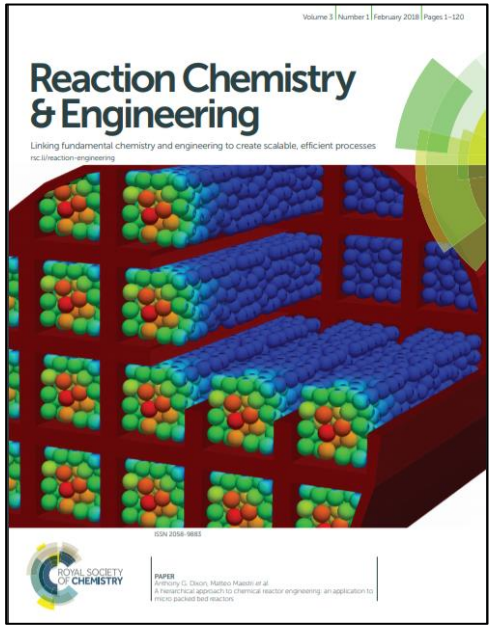



S. Rebughini, A. Cuoci, M. Maestri, Chem. Eng. J., 2016

# *Comparison of micro packed bed and packed bed*

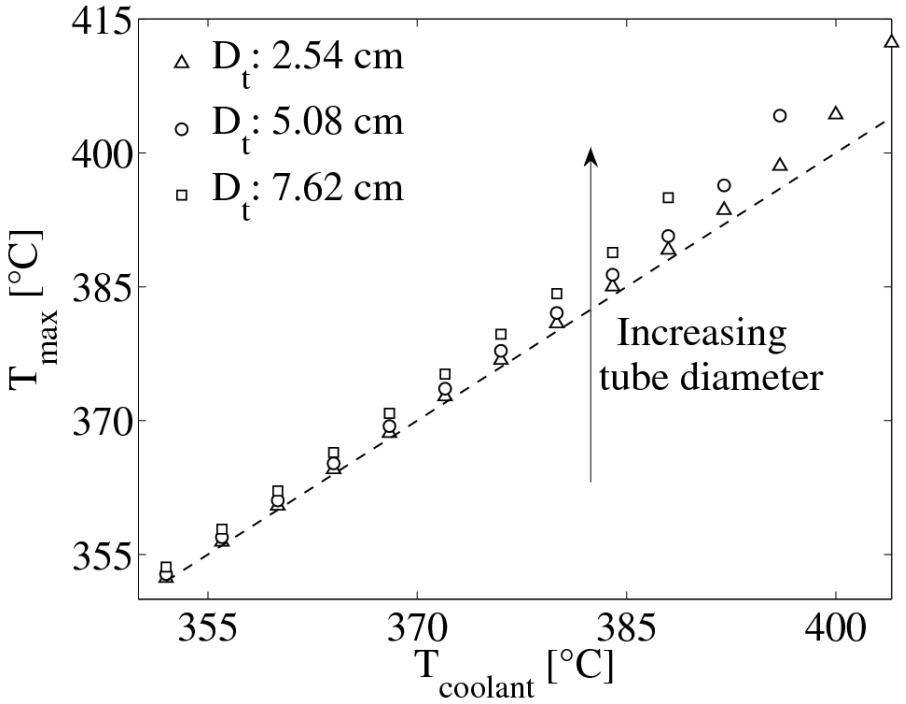


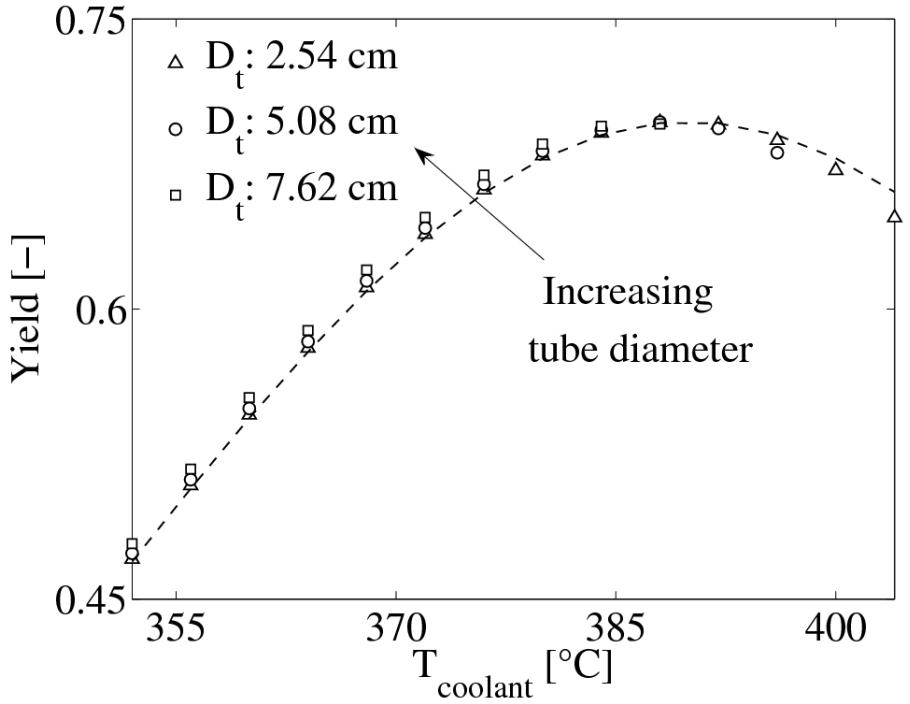


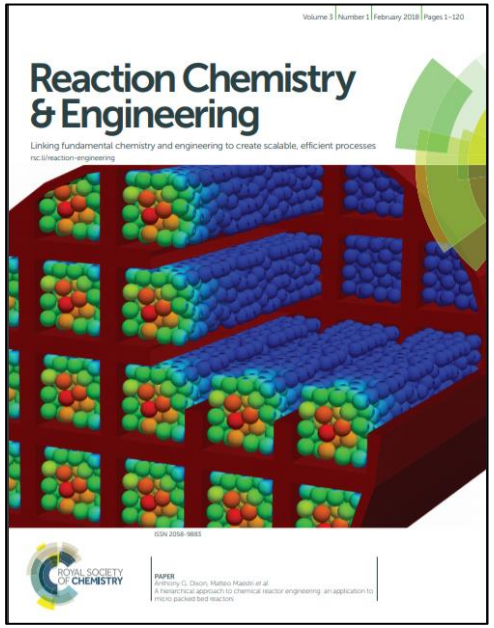

S. Rebughini, M. Bracconi, A.G. Dixon, M. Maestri, Reac. Chem. Eng., 2018

# *Comparison of micro packed bed and packed bed*



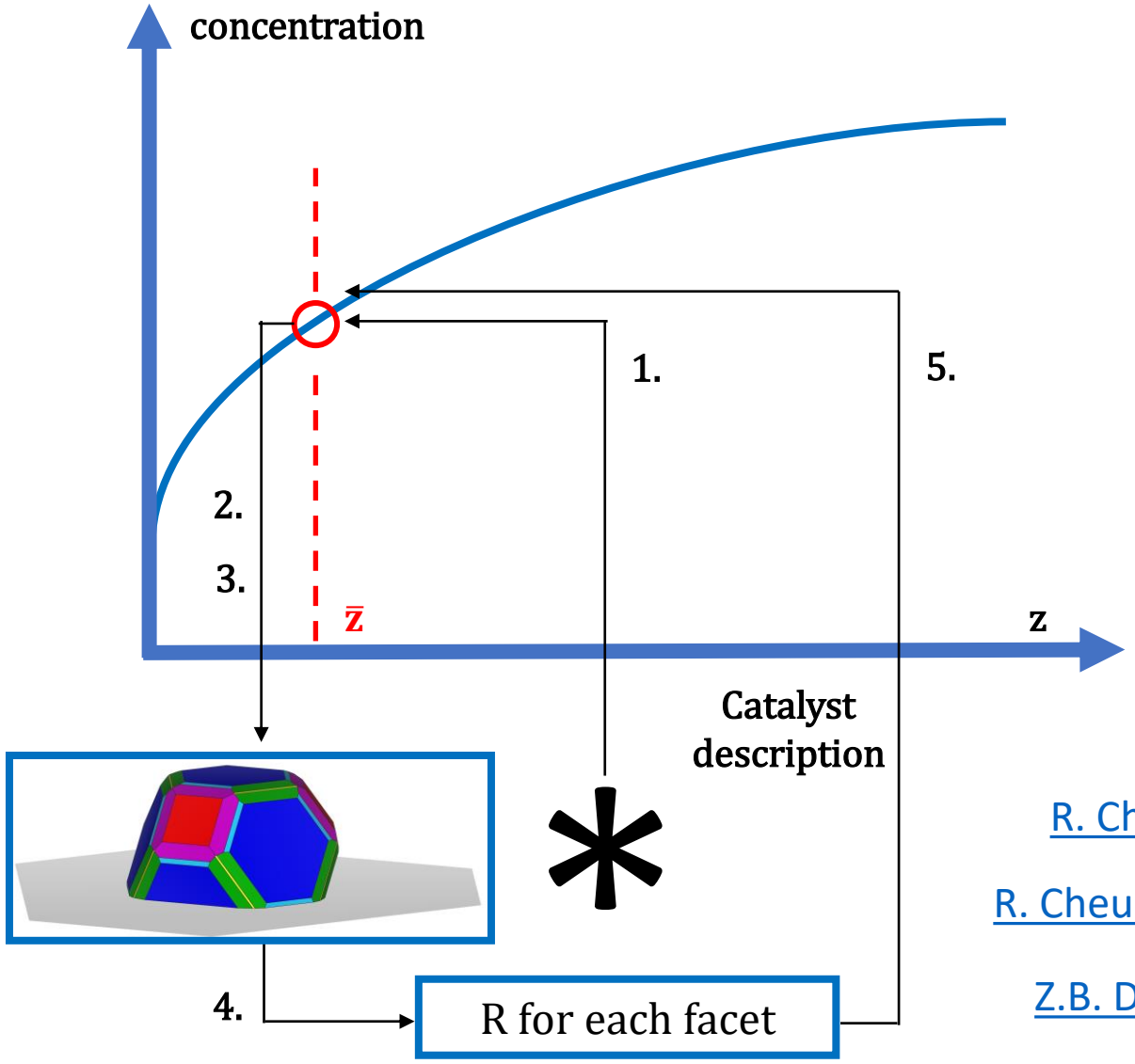



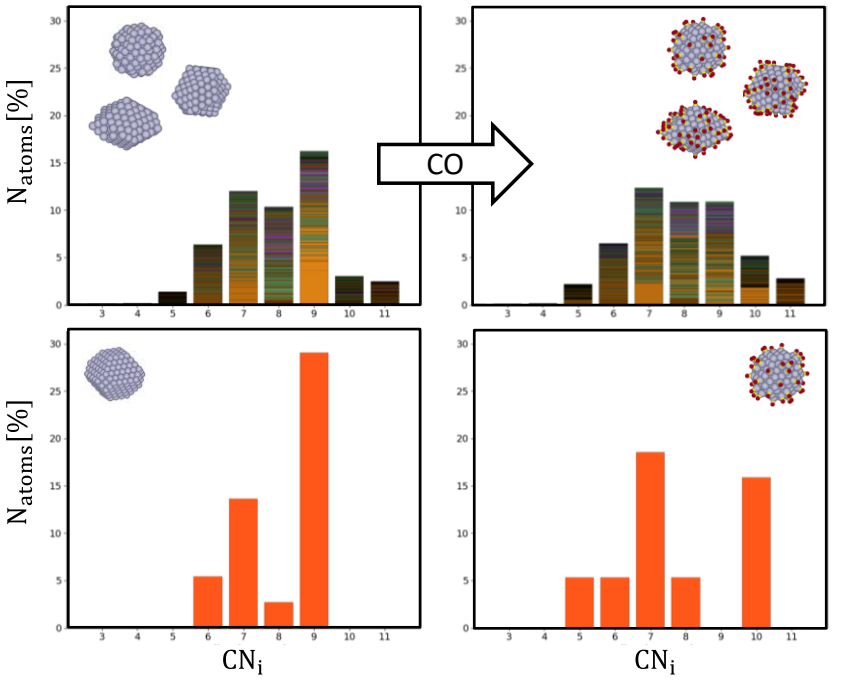

Effect of the tube diameter of the reactor performances in terms of yield and maximum temperature inside the reactor




S. Rebughini, M. Bracconi, A.G. Dixon, M. Maestri, Reac. Chem. Eng., 2018

# *Extension to structure-dependent microkinetic modeling*




ensemble of shapes (773 K)

ground state shape


R. Cheula, A. Soon and M. Maestri, Cat. Sci. Tech., 8 (2018) 3493

R. Cheula, M. Maestri, G. Mpourmpakis, ACS Catal., 10 (2020) 6149

Z.B. Ding and M. Maestri, Ind. & Eng. Chem. Res., 58 (2019) 9864

# *Conclusions*

- ✓ The design of the observed functionality of the catalyst requires reaction engineering

- ✓ Methods for coupling CFD and surface reactivity:
  - operator splitting
  - multiregion approaches
  - CFD DEM

- ✓ The calculation of the reaction terms is the bottleneck of the simulation:
  - ISAT
  - Cell agglomeration
  - Pre-computing of the reactivity (…a must for kMC coupling)

- ✓ Hierarchical approaches to tackle problems of scale relevant to practical applications

# *Tools*

https://github.com/multiscale-catalysis-polimi

**catalyticFoam:**
https://github.com/multiscale-catalysis-polimi/catalyticFoam

**Tools for structure-dependent microkinetic modeling:**
https://github.com/multiscale-catalysis-polimi/nanoparticles_ensembles

**Machine Learning adaptive design procedure:**
https://github.com/multiscale-catalysis-polimi/adaptiveDesignProcedure

**Polimi-reactor-modeling suite:**
macroscopic reactor models (pseudo-homogeneous/heterogeneous models, intraphase phenomena; 1D-2D models, …)
*** it will be released by the end of the summer ***

## *Acknowledgements*


Mauro Bracconi
Alberto Cuoci
Giancarlo Gentile
Tiziano Maffei
Stefano Rebughini
Riccardo Uglietti




677423

814416


**Thank you for your attention!**

# Coupling microkinetic modeling and kMC simulations with CFD in heterogeneous catalysis part 2

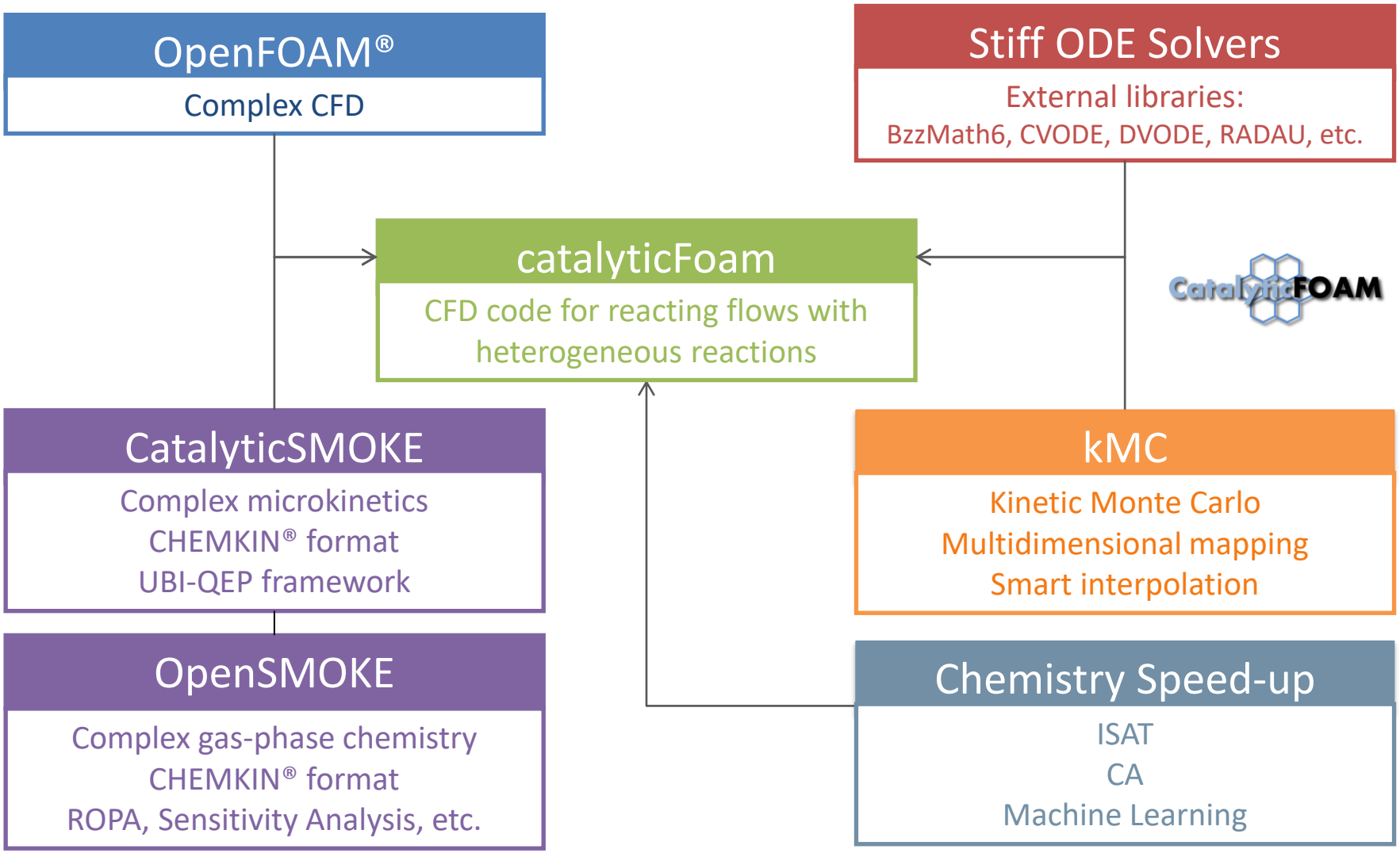
**Mauro Bracconi & Matteo Maestri**

Laboratory of Catalysis and Catalytic Processes LCCP

POLITECNICO MILANO 1863

Online workshop on «Theory, applications and tools for multiscale kinetic modeling»
Day 3 – July 15, 2020

# catalyticFoam background




**www.catalyticfoam.polimi.it** - https://github.com/multiscale-catalysis-polimi/catalyticFoam

Mauro Bracconi – Matteo Maestri

# Some technicalities …

- **catalyticFoam** is written in C++ while some additional features use Python language

- **catalyticFoam** is built upon the OpenFOAM framework and it is fully compatible with version **4.x** and **5.x**

- **catalyticFoam** requires the following external compulsory libraries:
  - **Eigen** (http://eigen.tuxfamily.org/index.php?title=Main_Page)
  - **RapidXML** (http://rapidxml.sourceforge.net/)
  - **Boost** C++ (http://www.boost.org/)
  - **OpenSMOKE++** (provided with catalyticFoam)

- **catalyticFoam** has been currently tested under several Linux distros (e.g., Ubuntu, CentOS). Windows compatibility is under assessment and development

- **catalyticFoam** is a text-based software without GUI

# How to install in Linux

1/ Open a bash terminal

2/ Clone the github repository:

**git clone https://github.com/multiscale-catalysis-polimi/catalyticFoam.git**

3/ Change directory

**cd catalyticFoam**

4/ Setup the installation enviroment by setting the path to the external compulsory library in **mybashrc** file

```
export OFVERSION='-DOFVERSION=50'

#Compulsory path
export EIGEN_LIBRARY_PATH=$HOME/NumericalLibraries/Eigen/Eigen-3.3.5
export BOOST_LIBRARY_PATH=$HOME/NumericalLibraries/Boost/boost_1_68_0/build
export RAPIDXML_LIBRARY_PATH=$HOME/NumericalLibraries/RapidXML/rapidxml-1.13
```

6/ Source the bash file:

**source mybashrc**

5/ Compile the software and libraries with:

**./Allwmake**

# Heterogeneous chemistry management

## Generation of catalyticFoam compatible kinetic mechanism

**catalyticFoam manages**:

- macroscopic reaction rates (e.g., LHHW models) – Day 1
- microkinetic mean field models (supplied in CHEMKIN format) – Day 1
- kinetic Monte Carlo schemes – Day 2

**Speeding-up chemical simulation**:

- **In-situ adaptive tabulation**
- **Cell agglomeration**
- **Machine Learning**

# Macroscopic reaction rates

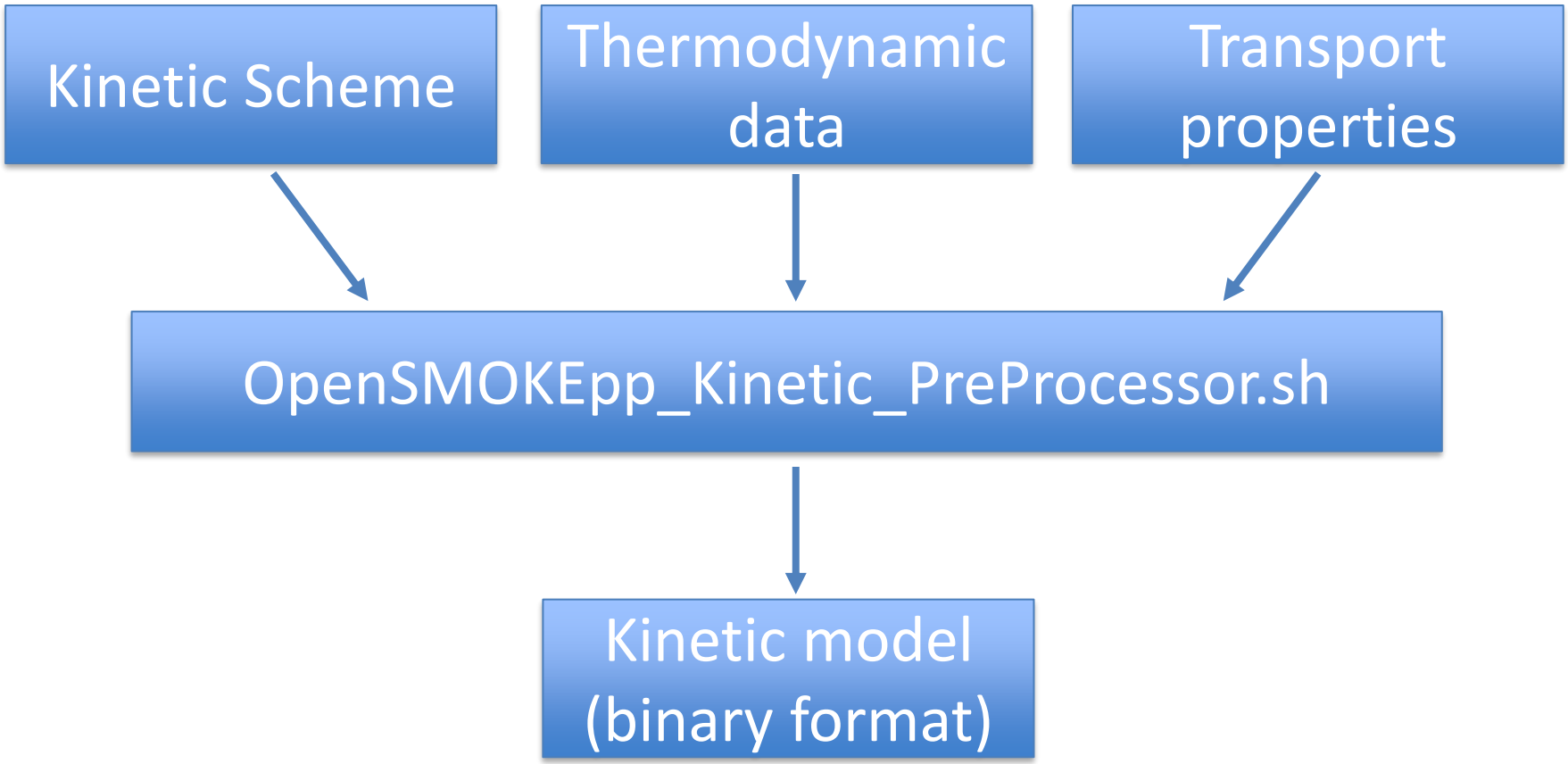
**Macroscopic reaction rates are directly introduced in the source code**

**A dummy kinetic scheme in CHEMKIN format is required to compute thermodynamic properties**

**You need to modify reactions/reactionRates.H, by introducing your specific rates:**

```
namespace OpenSMOKE
{
        template<typename map>
        class KineticsMap_Surface_CHEMKIN_Lumped : public KineticsMap_Surface_CHEMKIN<map>
        {
                public:

                KineticsMap_Surface_CHEMKIN_Lumped(ThermodynamicsMap_Surface_CHEMKIN<map>& thermo, rapidxml::xml_document<>& doc)
                : KineticsMap_Surface_CHEMKIN<map>(thermo, doc)
                {

                }


                virtual void UserDefinedReactionRates(const OpenSMOKEVectorDouble& c,
                                                                                        const OpenSMOKEVectorDouble& z,
                                                                                        const OpenSMOKEVectorDouble& a,
                                                                                        const OpenSMOKEVectorDouble& gamma)

                {


                        // Fill this with your kinetic model
                        // Example in github

                }

                private:
        };

}
```

# Microkinetic mechanism



# Microkinetic mechanism

| file .kin | file .sur | file .tdc | file .tra |
| --- | --- | --- | --- |
| For each homogeneous reaction define:<br>• stochiometry<br>• type of reaction<br>• kinetic parameters | For each heterogenous reaction define:<br>• stochiometry<br>• type of reaction<br>• kinetic parameters | The thermodynamic data are specified for all the species as a function of temperature | Contains the coefficients through which transport properties are computed |

# Microkinetic mechanism

## Homogeneous chemistry

List of all the elements

List of all the species

Equilibrium reaction

Only forward reaction

Third body reaction

Fall-Off reaction

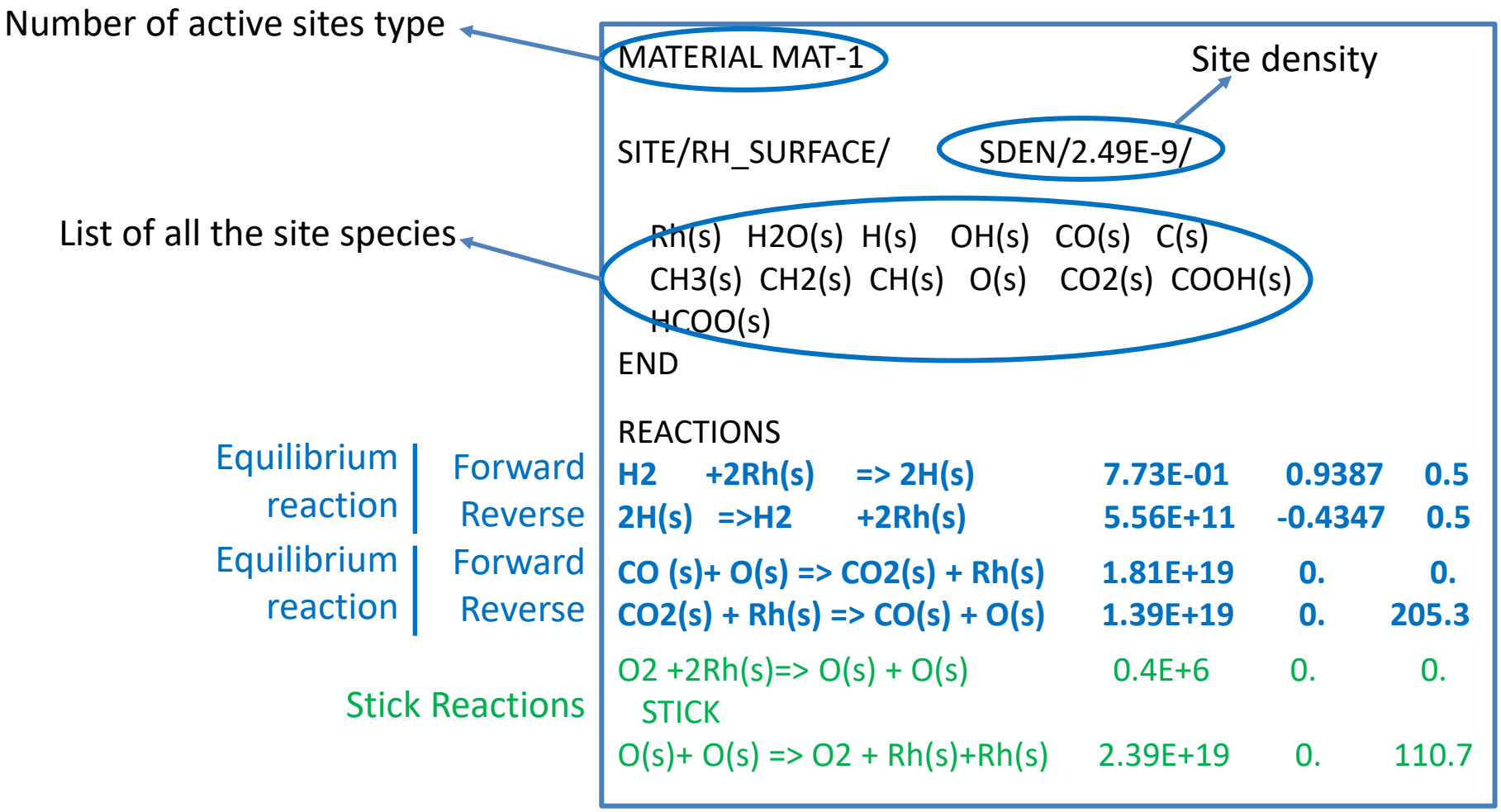
```
ELEMENTS
O  H  C  N  AR
END

SPECIES
H2   H   O   O2   OH   H2O   HO2   H2O2   C   CH   CH2
CH   CH3   CH4   CO   CO2
END

REACTIONS
CH2  +  O2 <=>  O  +  CH2O          2.40E+12   .00   1500.00
O  +  CH3  =>  H  +  H2  +  CO      3.37E+13   .00      .00
2O  +  M  =>  O2  +  M              1.20E+17  -1.00     .00
H2/2.4/H2O/15.40/CH4/2.00/CO/1.75/CO2/3.60
CH  +  H2  <=>  CH3                 1.97E+12   .43   -370.00
     LOW/    4.82E+25    -2.80    590.00/
    TROE/     0.578  122.0  2535.0  9365.0 /
END
```

kinetic parameters A α Ea

$$k = A \cdot T^{\alpha} \cdot \exp\left(-\frac{Ea}{R \cdot T}\right)$$

# Microkinetic mechanism

## Heterogeneous chemistry



# Microkinetic mechanism

## Thermodynamic properties

$$\frac{C_P}{R} = a_1 + a_2 \cdot T + a_3 \cdot T^2 + a_4 \cdot T^3 + a_5 \cdot T^4$$

$$\frac{H^0}{RT} = a_1 + \frac{a_2}{2} \cdot T + \frac{a_3}{3} \cdot T^2 + \frac{a_4}{4} \cdot T^3 + \frac{a_5}{5} \cdot T^4 + \frac{a_6}{6} \cdot T^5$$

$$\frac{S^0}{R} = a_1 lnT + a_2 \cdot T + \frac{a_3}{2} \cdot T^2 + \frac{a_4}{3} \cdot T^3 + \frac{a_5}{4} \cdot T^4 + a_T$$

Name of the species in the scheme

Elementary composition

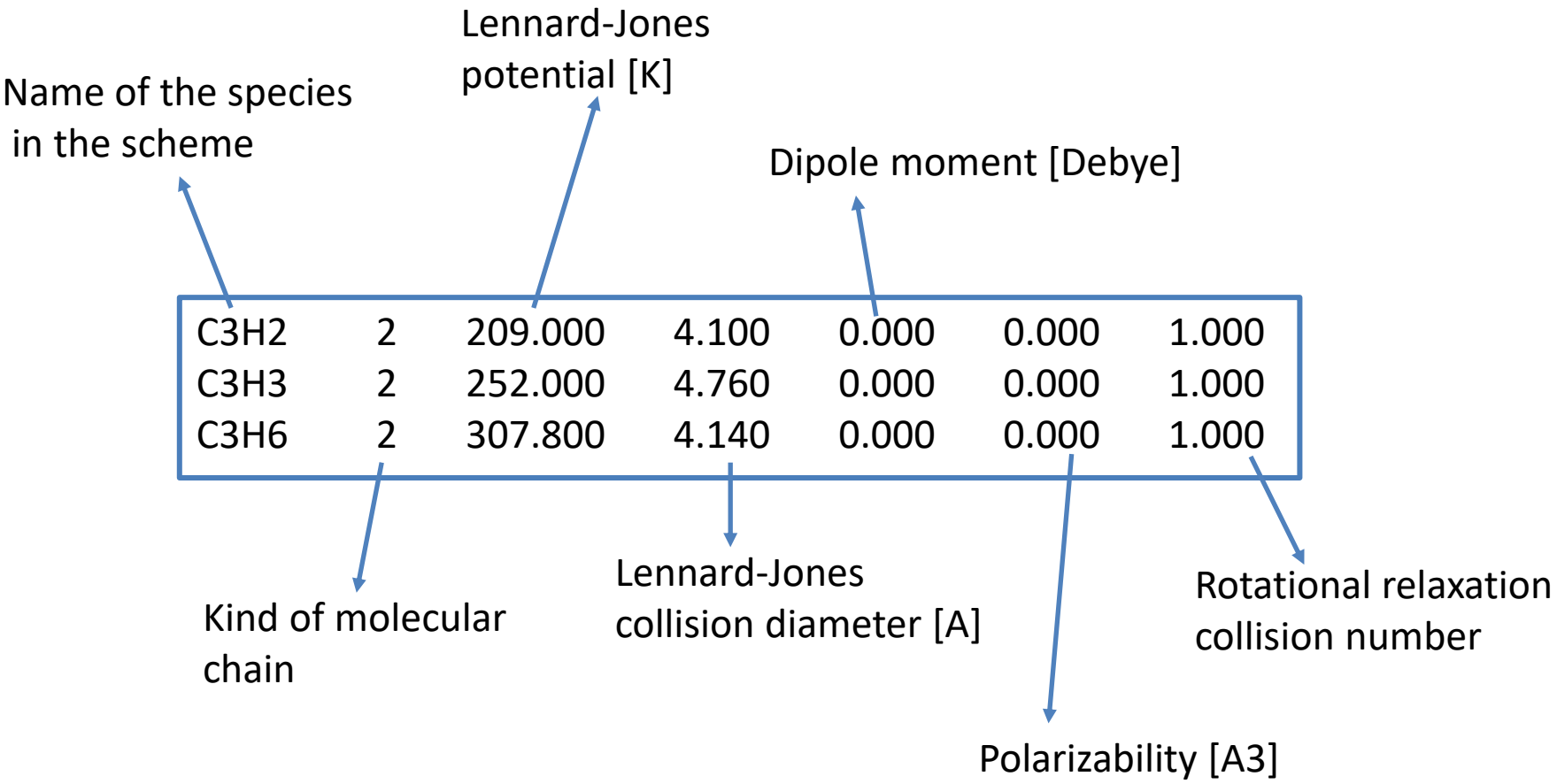
Ranges of low and high temperature

CH2CHO SAND86 O1 H3 C2 G 300.000 5000.000 1000.000

0.05975670E+02 0.08130591E-01 -0.02743624E-04 0.04070304E-08
-0.02176017E-12 0.04903218E+04 -0.05045251E+02 0.03409062E+02
0.10738574E-01 0.01891492E-04 -0.07158583E-07 0.02867385E-10
0.15214766E+04 0.09558290E+02

# Microkinetic mechanism

## Transport properties



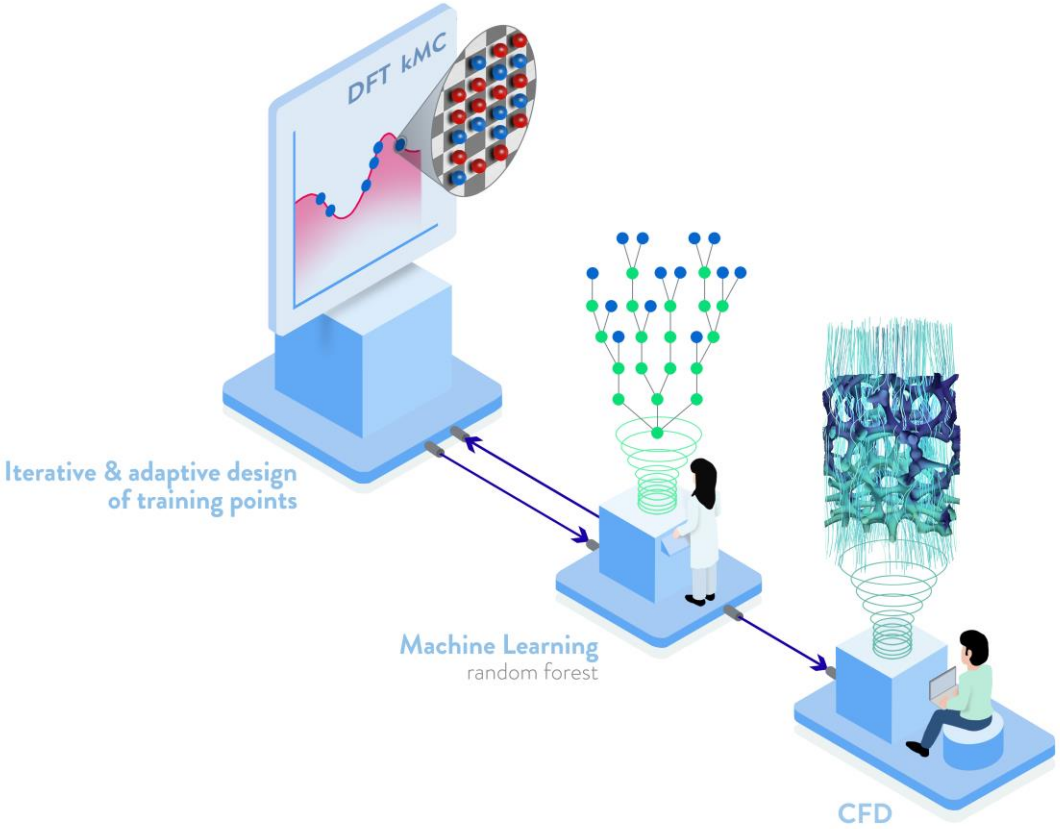
# Microkinetic mechanism

```
Dictionary PreProcessor
{

        @Thermodynamics         Thermo.tdc;
        @Kinetics               Kinetics.kin;
        @Surface                Surface.sur;
        @Transport              Transport.tra;

        @Output                         kinetics;
}
```

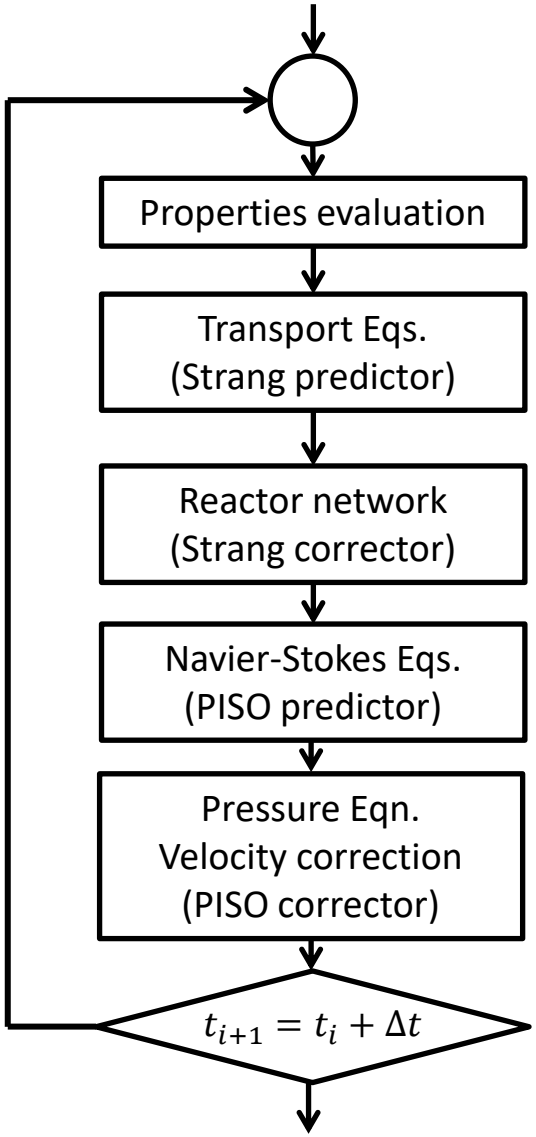
***OpenSMOKEpp_kinetic_PreProcessor***

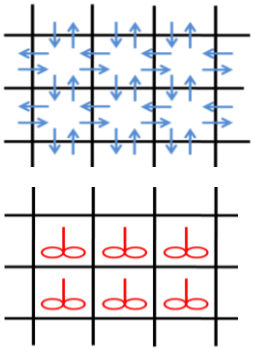
# kinetic Monte Carlo

**kMC schemes can be integrated into CFD simulations as macroscopic reaction rates where the TOF values are computed via tabulation method**



# Solver flowchart



**Main features**:

- Solution of the Navier-Stokes equations (laminar and turbulent regime)
- No limit to the number of species and reactions
- No limit in geometry


# An example of simulation

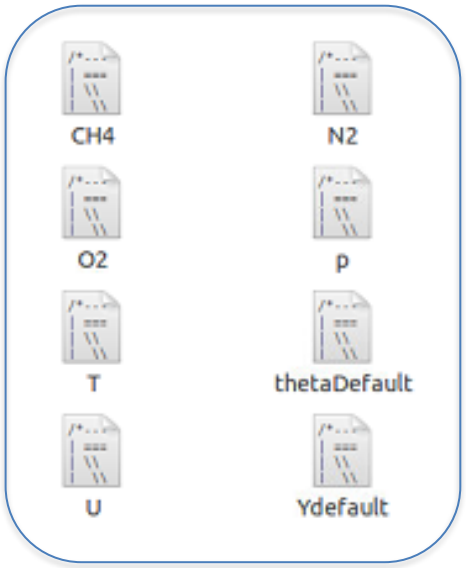
Mauro Bracconi – Matteo Maestri

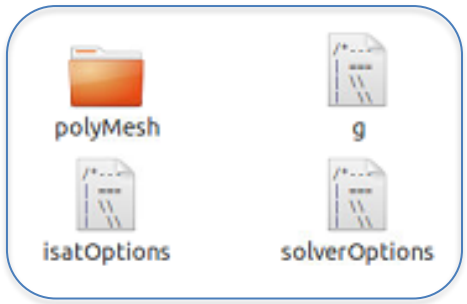
# Simulation folder structure

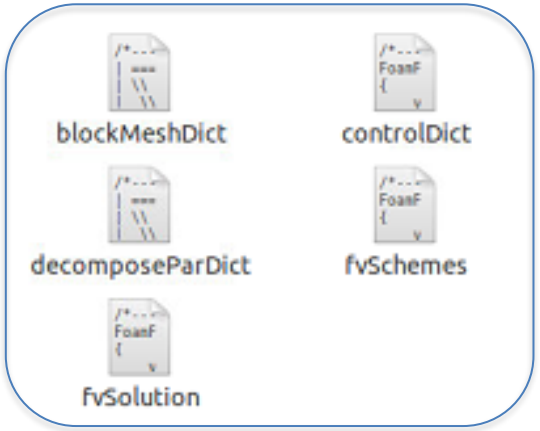
0

constant

system



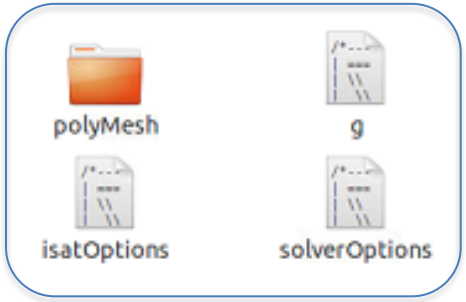




CatalyticFOAM

# constant folder

**Models** and **computational domain**:

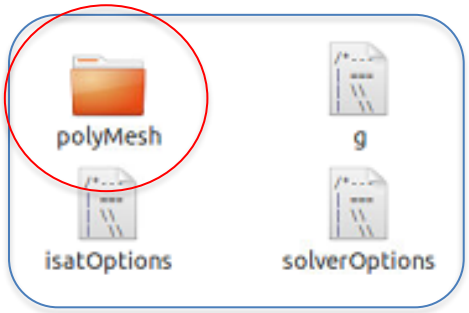
- Mesh folder
- Solver settings
- ISAT settings

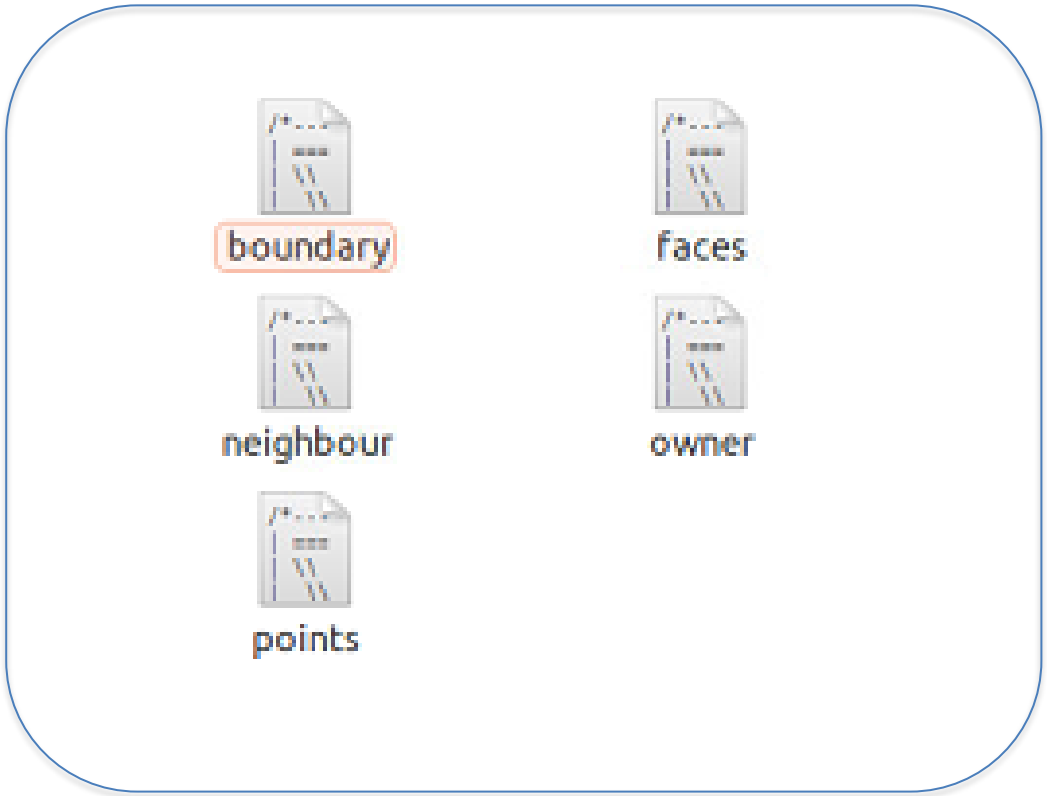


# constant folder

**Models** and **computational domain**:

- Mesh folder
- Solver settings
- ISAT settings

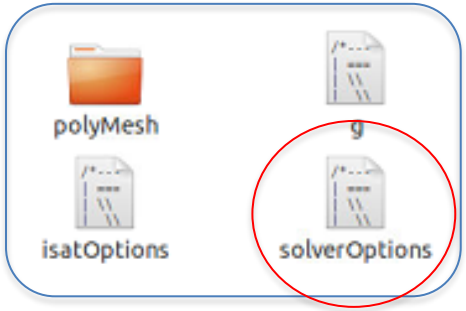


# constant folder

**Models** and **computational domain**:

- Mesh folder
- Solver settings
- ISAT settings

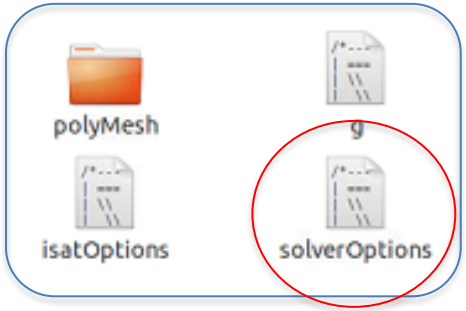



```
Kinetics
{
        folder                          "kinetic/kinetics";  // path to precompiled kineti
        inertSpecies                    N2;                  // inert species
}

PhysicalModel
{
        // Operator splitting algorithm
        strangAlgorithm                 "TransportReactionMomentum";

        // Switch for homogeneous and heterogeneous chemistries
        homogeneousReactions            off;
        heterogeneousReactions          on;

        // Catalytic load (m2_cat/m2_geom)
        alfaCatalyst                    10.0;

        // Name of reacting patches
        catalyticWalls (reactingWall);

        // Switch for energy equation
        energyEquation                          off;
        reactionHeatFromHeterogeneousReactions  on;

        // Consider the catalytic cell as constant pressure | volume (true | false)
        constPressureBatchReactor               true;

        // Consider mass diffusion in energy equation
        massDiffusionInEnergyEquation   on;

        // Physical model for diffusion
        diffusivityModel                "multi-component";              // lewis-numbers

        // Consider DpDt term in energy balance
        includeDpDt                     off;

        // Consider Soret's effect
        soretEffect                     off;
}
```

# constant folder

**Models** and **computational domain**:

- Mesh folder
- Solver settings
- ISAT settings

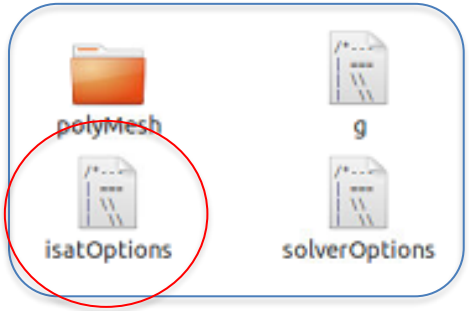



```
OdeHomogeneous
{
    odeSolver   "OpenSMOKE";
    relTolerance    1e-7;
    absTolerance    1e-12;
    maximumOrder    5;
    fullPivoting    false;
}

OdeHeterogeneous
{
    odeSolver   "OpenSMOKE";
    relTolerance    1e-7;
    absTolerance    1e-12;
    maximumOrder    5;
    fullPivoting    false;
}

#include "isatOptions"

LewisNumbers
{
    O2  1.0;
    H2  1.1;
    H2O 1.2;
    N2  1.3;
}
```

# constant folder

## **Models** and **computational domain**:

- Mesh folder
- Solver settings
- ISAT settings

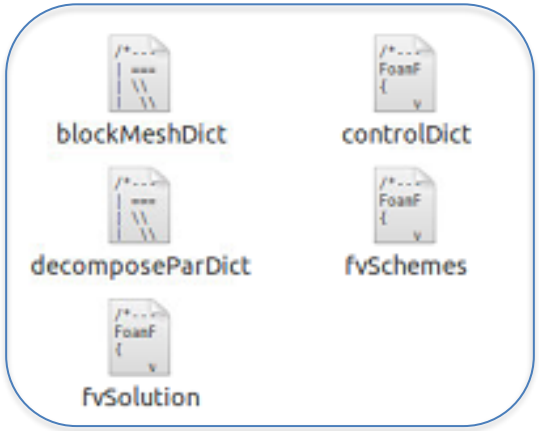



```
// Basic parameters
ISAT                    off;            // ISAT on/off
tolerance               1e-4;       // ISAT tolerance [default: 1e-4]
numberSubSteps              1;      // ISAT number of substeps for calculating gradient mapping [default: 1]

// Lists of useful leaves
searchMRU               on;         // search for MRU (Most Recently Used) leaves [default: on]
searchMFU               on;         // search for MFU (Most Frequently Used) leaves [default: on]
maxSizeMRU              100;        // max size of MRU (Most Recently Used) [default: 100]
maxSizeMFU              100;        // max size of MFU (Most Frequently Used) [default: 100]
maxSearchMRU                10;     // max numbers of leaves to be tested (MRU) [default: 10]
maxSearchMFU                30;     // max numbers of leaves to be tested (MFU) [default: 30]

// What to do when the tre is full?
maxSizeBT               100000;     // max size of binary tree [default: 100000]
clearingIfFull              off;            // if on the tree is cleared when full [default: off]

// Balancing coefficients
cleanAndBalance             on;         // clean and balance [default: on]
balanceFactor               2.0;        // number of events (add, grow, retrieve) * maxSizeBT
                                        //between each cleaning procedure (set to have cleaning every 100-1000 time step)

// This parameters influence how severe is the cleaning procedure
maxTimeOldCoeff             0.7;        // leaf used for the last time   [default: 0.7]
maxGrowCoeff                0.5;        // leaf grown more than maxGrowCoeff*maxSizeBT times [default: 0.5]
minUsedCoeff                0.01;       // leaf used less than minUsedCoeff*(nUse of last leaf in the MFU) [default: 0.01]
maxHeightCoeff              5.;         // imbalance between the depth of each main branches of the tree (use value larger tha

// Factorizations
luFactorization             "Partial";  // LU factorization of dense matrices (Partial || Full) [default: Partial]
qrFactorization             "Full";     // QR factorization of dense matrices (NoPivoting || Partial || Full) [default: Full]

// Scaling factors
scalingFactors
{
    T   1.e4;
    others  1.;
}

// Absoute weights to scale the errors
scalingErrors
{
    T   1.;
    others  1.;
}
```

# system folder

**Simulation control** and **numerical scheme/solvers**:

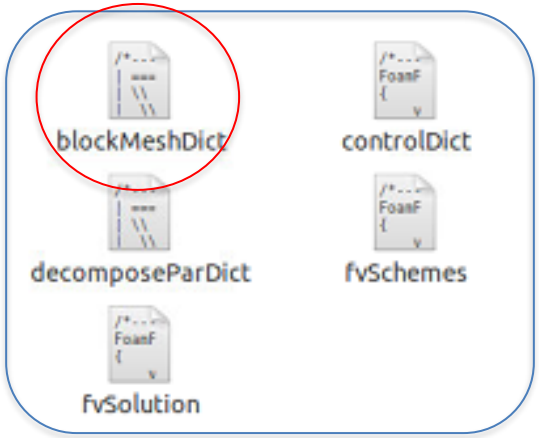
- Mesh set-up
- Simulation control
- Parallelization
- Numerical schemes
- Numerical solvers



# system folder

**Simulation control** and **numerical scheme/solvers**:

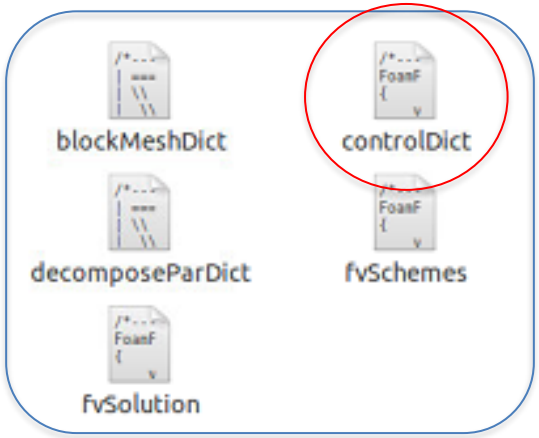
- Mesh set-up
- Simulation control
- Parallelization
- Numerical schemes
- Numerical solvers



# system folder

## **Simulation control** and **numerical scheme/solvers**:

- Mesh set-up
- Simulation control
- Parallelization
- Numerical schemes
- Numerical solvers

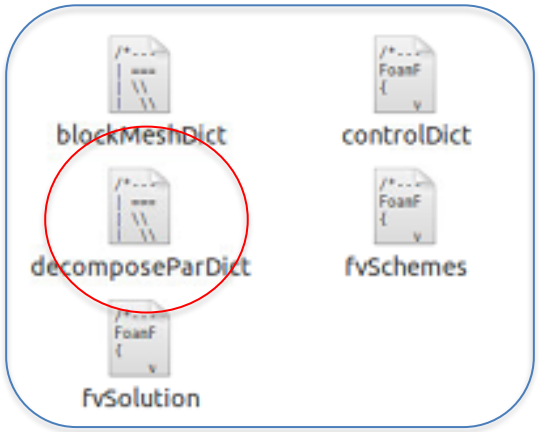



```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      controlDict;
}
// * * * * * * * * * * * * * * * * * *

application     catalyticFoam;

startFrom       latestTime;

startTime       0;

stopAt          endTime;

endTime         1.0;

deltaT          1.e-8;

writeControl    adjustableRunTime;

writeInterval   0.01;

purgeWrite      0;

writeFormat     ascii;

writePrecision  18;

writeCompression uncompressed;

timeFormat      general;

timePrecision   6;

runTimeModifiable yes;

adjustTimeStep  yes;

maxCo           0.1;

libs ("libCatalyticWall.so");
```

# system folder

## **Simulation control** and **numerical scheme/solvers**:

- Mesh set-up
- Simulation control
- Parallelization
- Numerical schemes
- Numerical solvers

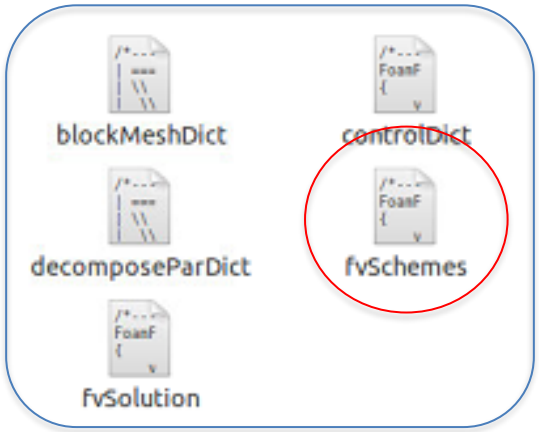



```
numberOfSubdomains 2;

method          hierarchical;

simpleCoeffs
{
    n               (2 2 8);
    delta           0.001;
}

hierarchicalCoeffs
{
    n               ( 2 1 1);
    delta           0.00001;
    order           zyx;
}

manualCoeffs
{
    dataFile        "";
}

distributed     no;

roots           ( );
```

# system folder

## **Simulation control** and **numerical scheme/solvers**:

- Mesh set-up
- Simulation control
- Parallelization
- Numerical schemes
- Numerical solvers

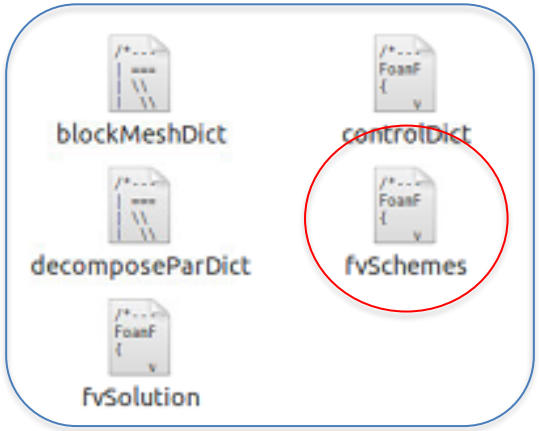



```
ddtSchemes
{
    default         Euler;
}

gradSchemes
{
    default         Gauss linear;
    grad(p)         Gauss linear;
    grad(U)         Gauss linear;
}

divSchemes
{
    default                             none;

    div(phi,U)                          Gauss linearUpwindV grad(U);
    div(phi,T)                          Gauss limitedLinear 1;

    div(phi,Yi_h)                       Gauss limitedLinear01 1;
    div(Jc,Yi)                          Gauss limitedLinear01 1;

    div((gas::mu*dev2(grad(U).T())))    Gauss linear;

}
```

# system folder

## **Simulation control** and **numerical scheme/solvers**:

- Mesh set-up
- Simulation control
- Parallelization
- Numerical schemes
- Numerical solvers

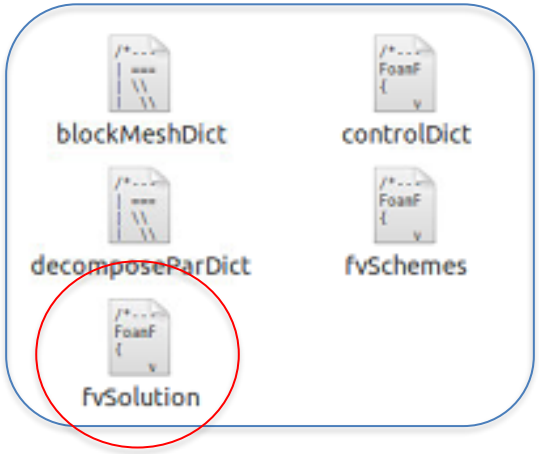



```
laplacianSchemes
{
    default                     Gauss linear orthogonal;
}

interpolationSchemes
{
    default         linear;
}

snGradSchemes
{
    default         orthogonal;
}

fluxRequired
{
    default         no;
    p               ;
}
```

# system folder

## **Simulation control** and **numerical scheme/solvers**:

- Mesh set-up
- Simulation control
- Parallelization
- Numerical schemes
- Numerical solvers




```
p
{
    solver                  GAMG;
    tolerance               1e-6;
    relTol                  0.1;
    smoother                GaussSeidel;
    cacheAgglomeration      true;
    nCellsInCoarsestLevel   10;
    agglomerator            faceAreaPair;
    mergeLevels             1;
};
```

```
T
{
    solver          PBiCG;
    preconditioner  DILU;
    tolerance       1e-07;
    relTol          1e-07;
}
```

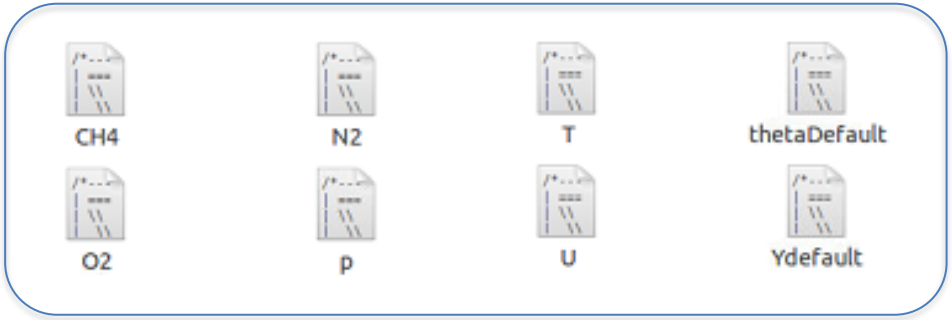
```
U
{
    solver          PBiCG;
    preconditioner  DILU;
    tolerance       1e-07;
    relTol          0.;
}
```

```
Yi
{
    solver          PBiCG;
    preconditioner  DILU;
    tolerance       1e-09;
    relTol          1e-07;
}
```

# 0 folder

**Initial** and **boundary conditions** for:

- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions



# 0 folder

**Initial** and **boundary conditions** for:

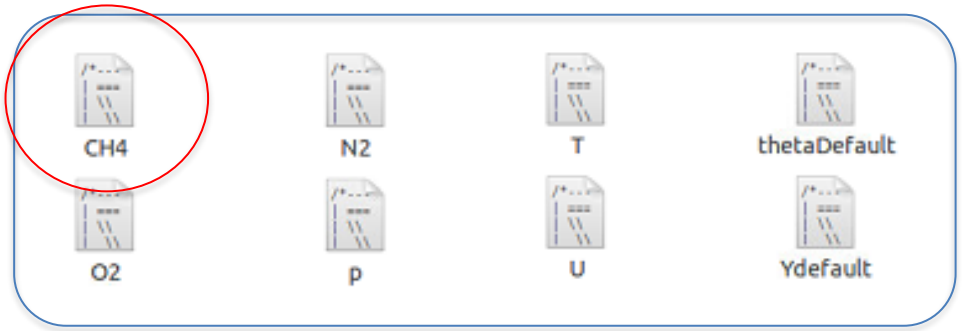
- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions


```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      CH4;
}
// * * * * * * * * * * * * * * * * * * * * *

dimensions      [0 0 0 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    reactingWall
    {
        type            catalyticWall;
    }
    inertWall
    {
        type            zeroGradient;
    }
    inlet
    {
        type            fixedValue;
        value           uniform 0.17;
    }
    outlet
    {
        type            zeroGradient;
    }

}
```

# 0 folder

**Initial** and **boundary conditions** for:

- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions




**BC for heterogeneous chemistry**

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      CH4;
}
// * * * * * * * * * * * * * * * * * * * * *

dimensions      [0 0 0 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    reactingWall
    {
        type            catalyticWall;
    }
    inertWall
    {
        type            zeroGradient;
    }
    inlet
    {
        type            fixedValue;
        value           uniform 0.17;
    }
    outlet
    {
        type            zeroGradient;
    }

}
```

# 0 folder

**Initial** and **boundary conditions** for:

- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions


**Neumann BC for inert wall**

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      CH4;
}
// * * * * * * * * * * * * * * * * * * * * *

dimensions      [0 0 0 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    reactingWall
    {
        type            catalyticWall;
    }
    inertWall
    {
        type            zeroGradient;
    }
    inlet
    {
        type            fixedValue;
        value           uniform 0.17;
    }
    outlet
    {
        type            zeroGradient;
    }

}
```

# 0 folder

**Initial** and **boundary conditions** for:

- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions


**Dirichlet BC for inlet**

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      CH4;
}
// * * * * * * * * * * * * * * * * * * * * *

dimensions      [0 0 0 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    reactingWall
    {
        type            catalyticWall;
    }
    inertWall
    {
        type            zeroGradient;
    }
    inlet
    {
        type            fixedValue;
        value           uniform 0.17;
    }
    outlet
    {
        type            zeroGradient;
    }

}
```

# 0 folder

**Initial** and **boundary conditions** for:

- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions

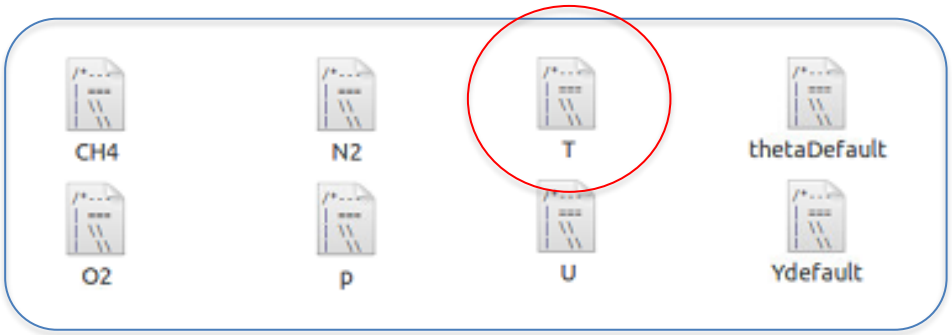

**Neumann BC for reactor outlet**

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      CH4;
}
// * * * * * * * * * * * * * * * * * * * * *

dimensions      [0 0 0 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    reactingWall
    {
        type            catalyticWall;
    }
    inertWall
    {
        type            zeroGradient;
    }
    inlet
    {
        type            fixedValue;
        value           uniform 0.17;
    }
    outlet
    {
        type            zeroGradient;
    }

}
```

# 0 folder

**Initial** and **boundary conditions** for:

- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions



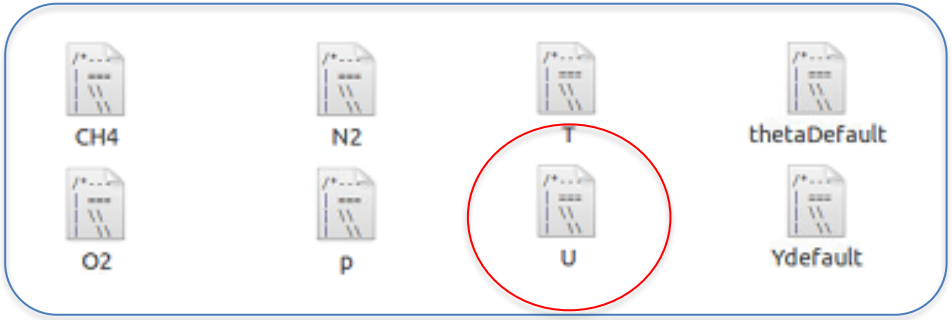

**BC for heterogeneous chemistry**

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      T;
}
// * * * * * * * * * * * * * * * * * * * * *

dimensions      [0 0 0 1 0 0 0];

internalField   uniform 573.15;


boundaryField
{
    reactingWall
    {
        type            catalyticWall;
    }
    inertWall
    {
        type            zeroGradient;
    }
    inlet
    {
        type            fixedValue;
        value           uniform 573.15;
    }
    outlet
    {
        type            zeroGradient;
    }
}
```

# 0 folder

**Initial** and **boundary conditions** for:

- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions

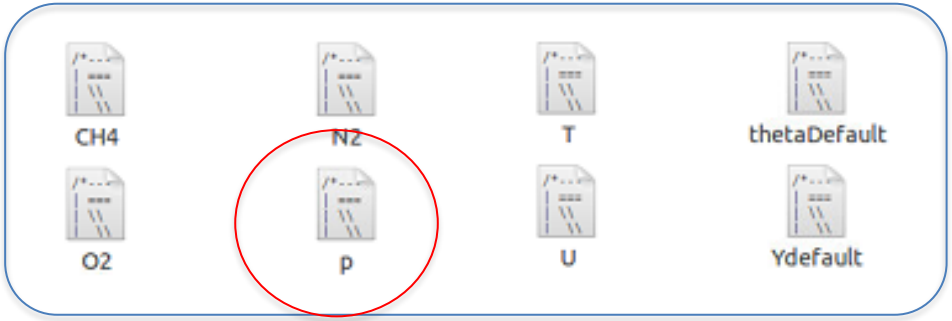



```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
    location    "0";
    object      U;
}
// * * * * * * * * * * * * * * * * * * * * * * * * *

dimensions      [0 1 -1 0 0 0 0];

internalField   uniform (0 0 0);


boundaryField
{
    reactingWall
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }
    inertWall
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }
    inlet
    {
        type            fixedValue;
        value           uniform (1. 0 0);
    }
    outlet
    {
        type            zeroGradient;
    }
}
```

# 0 folder

**Initial** and **boundary conditions** for:

- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions

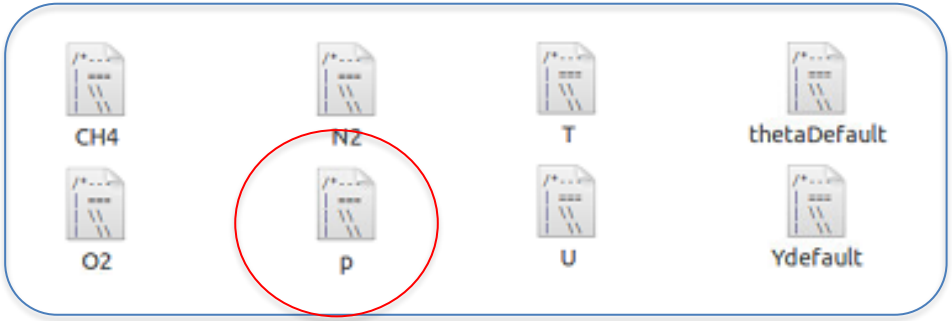



```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      p;
}
// * * * * * * * * * * * * * * * * * * * * * * * * *

dimensions      [1 -1 -2 0 0 0 0];

internalField   uniform 101325;


boundaryField
{
    reactingWall
    {
        type            zeroGradient;
    }
    inertWall
    {
        type            zeroGradient;
    }
    inlet
    {
        type            zeroGradient;
    }
    outlet
    {
        type            fixedValue;
        value           uniform 101325;
    }
}
```

# 0 folder

**Initial** and **boundary conditions** for:

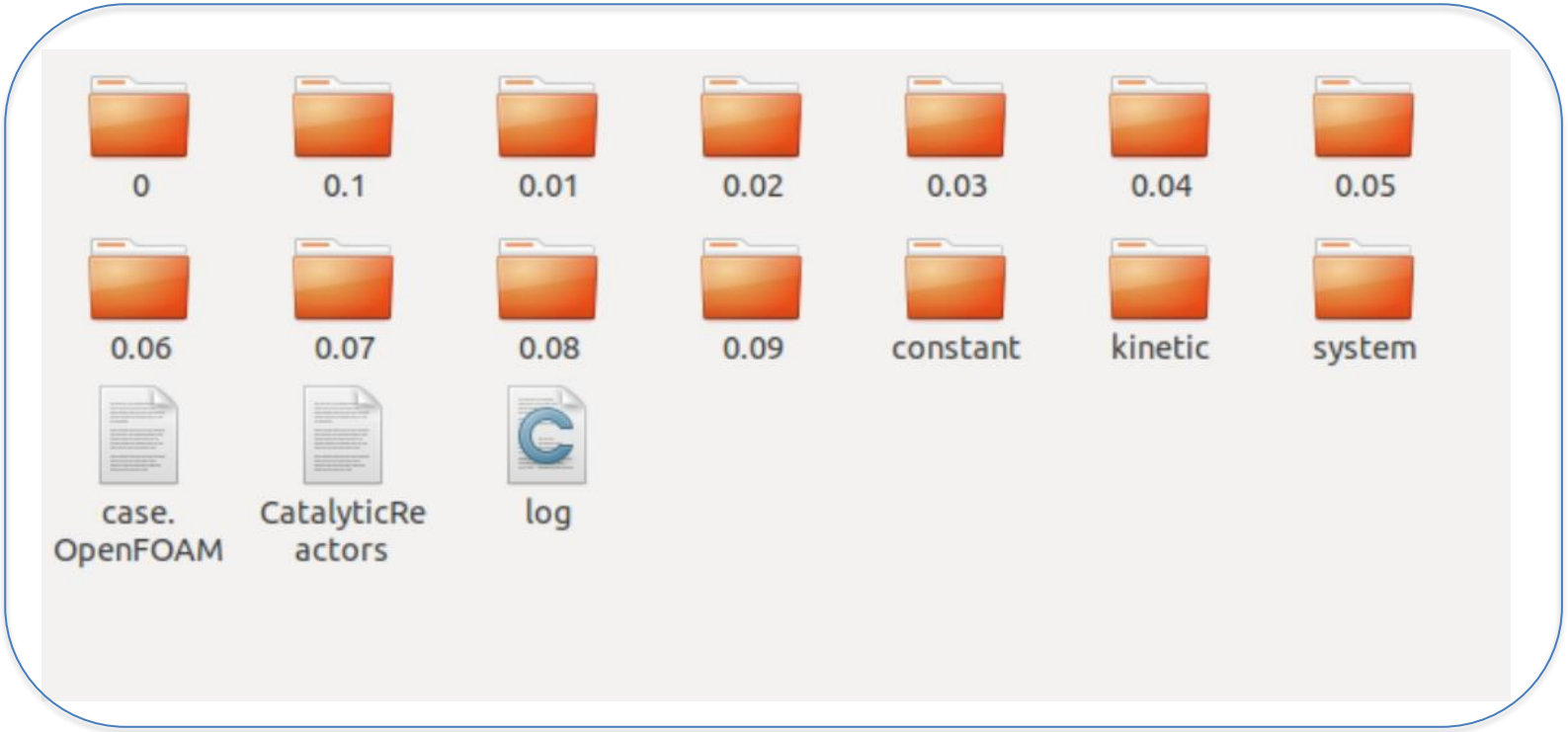
- Temperature – T
- Pressure – p
- Fluid velocity – U
- Mass fractions
- Site fractions




```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      p;
}
// * * * * * * * * * * * * * * * * * * * * * * * * *

dimensions      [1 -1 -2 0 0 0 0];

internalField   uniform 101325;


boundaryField
{
    reactingWall
    {
        type            zeroGradient;
    }
    inertWall
    {
        type            zeroGradient;
    }
    inlet
    {
        type            zeroGradient;
    }
    outlet
    {
        type            fixedValue;
        value           uniform 101325;
    }
}
```

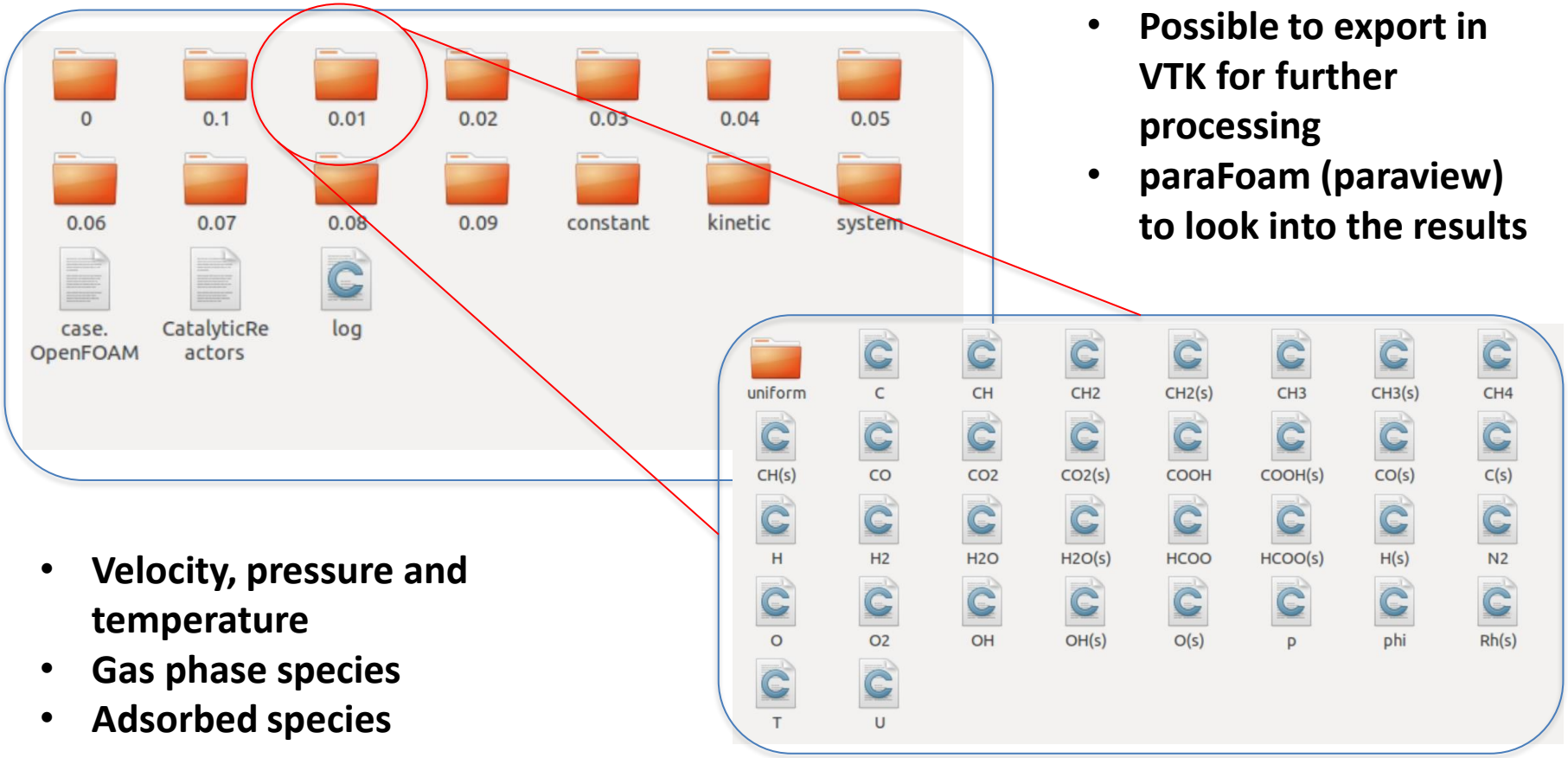
# Simulation output

- **Time folders saved according to the settings in the controlDict file**



# Simulation output

- **Each time folder contains simulation results**




- **Possible to export in VTK for further processing**
- **paraFoam (paraview) to look into the results**

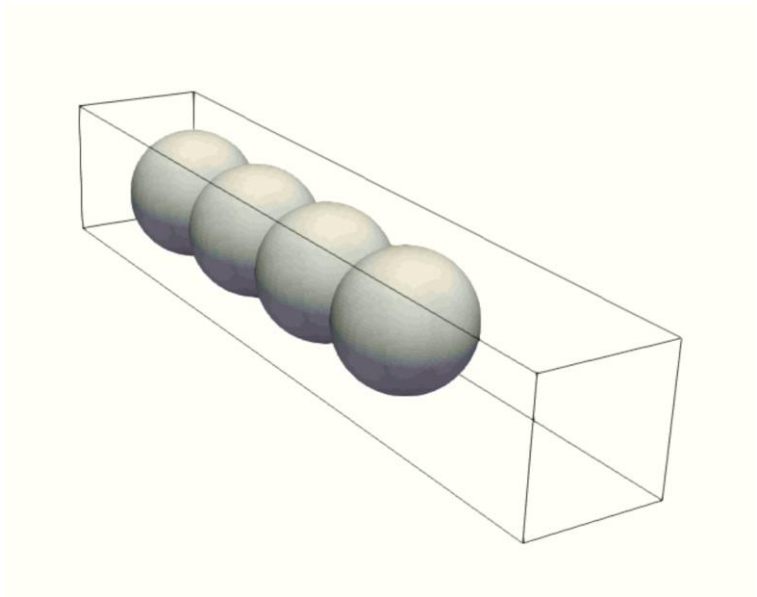
- **Velocity, pressure and temperature**
- **Gas phase species**
- **Adsorbed species**


# A simple case study

Mauro Bracconi – Matteo Maestri

# Case study

## CFD simulation of WGS on Rh in a 4-spheres string reactor



**Operating conditions**:

- Isothermal simulation (850 K)
- $CO/H_2O$ = 1:1 – CO = 2 % (mass fraction)
- $N_2$ inert
- Flowrate = 2.4 NmL/s
- Laminar conditions
- Pellet size = 4 mm

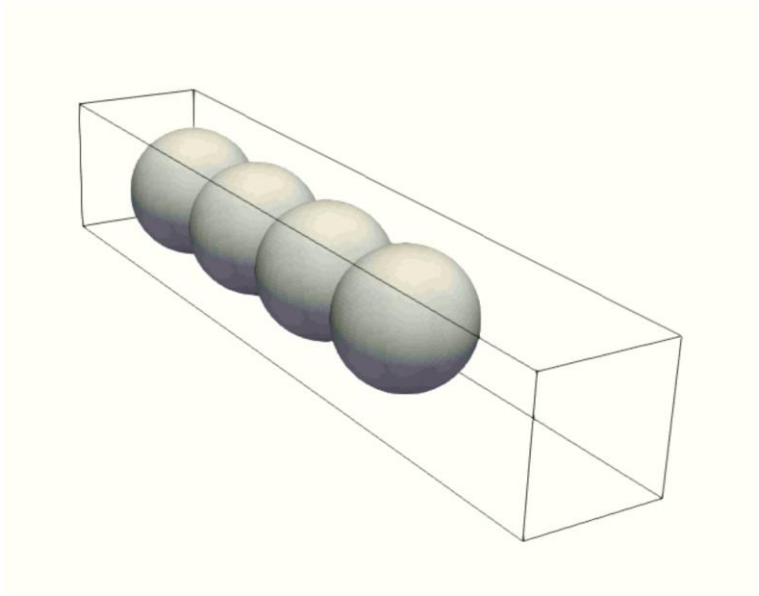
**Kinetic model**:

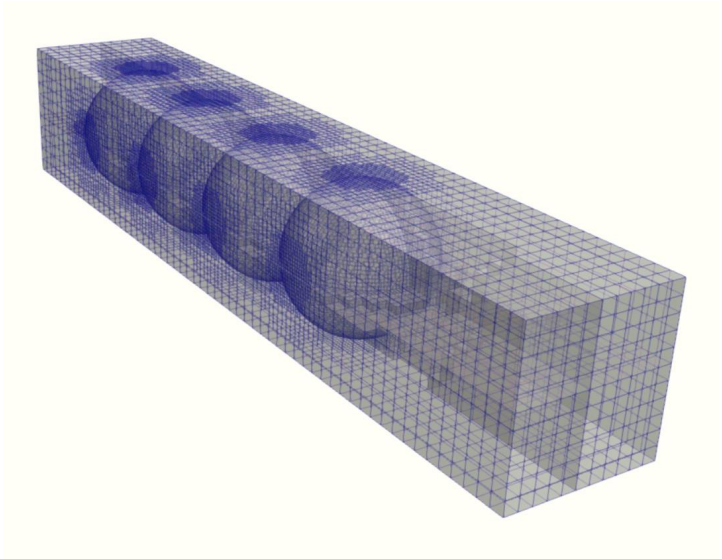
- UBI microkinetic model for $C_1$ activation on Rh [1]
- 16 gas species
- 11 adsorbed species
- 80 elementary steps

[1] M. Maestri et al., *AIChE Journal*, 55 2009

# Case study – mesh generation

## Moving from the geometry to the computational domain





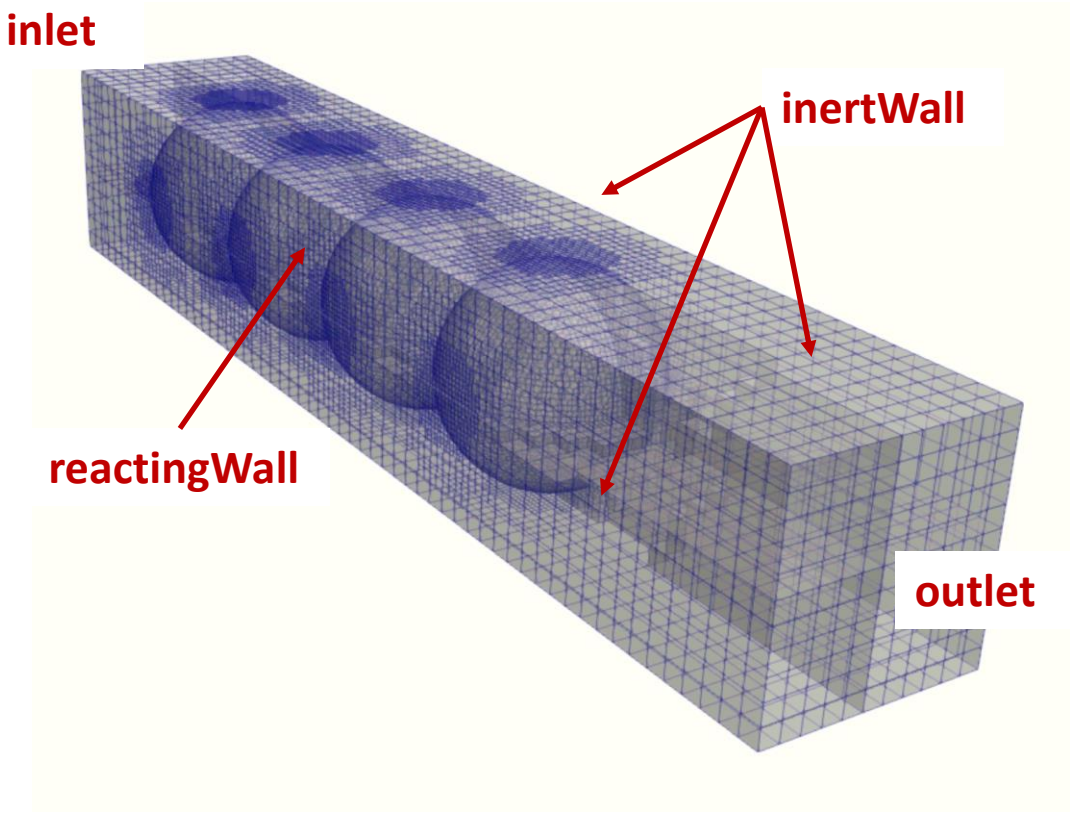
**Meshing:**

- OpenFOAM built-in tool:
  - blockMesh
  - snappyHexMesh
- Several commercial softwares (ANSYS, Pointwise, Gambit, GridGen, Tetgen ….)

***"Who holds the mesh holds the solution"***

# Case study – mesh generation

- Each external faces of the mesh is defined a patch
- During the meshing generation procedure each faces is named for the following steps



# Case study – boundary conditions

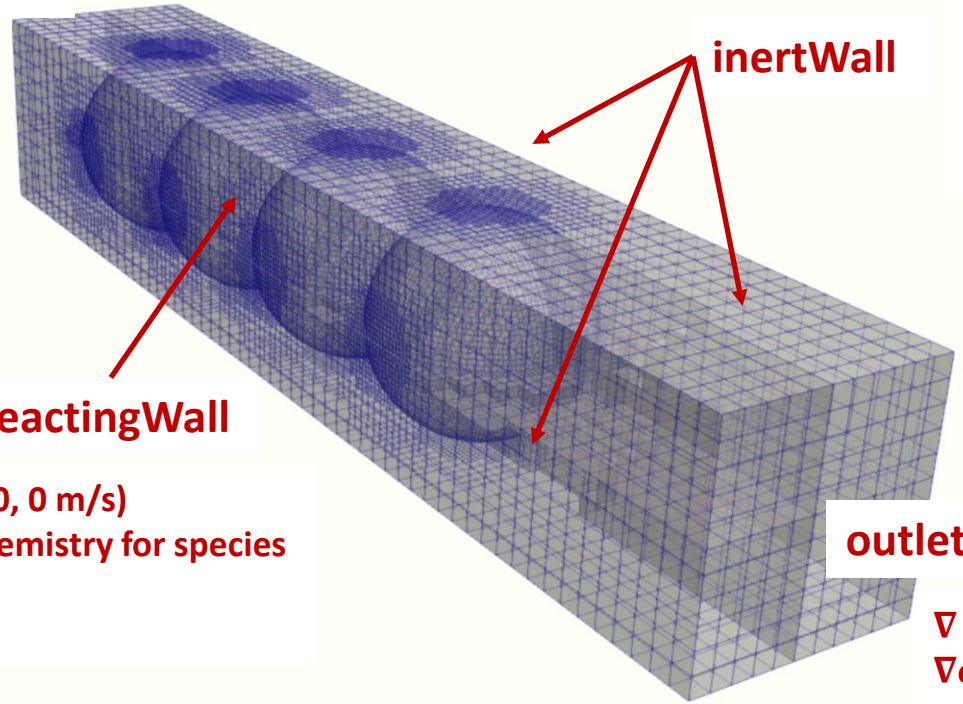
- **Define initial conditions:**

  U = (0, 0, 0.5 m/s)
  P = 1 atm
  T = 850 K
  CO = 0.02 (mass fraction)
  H2O = 0.02 (mass fraction)
  N2 = 0.96 (mass fraction)

- **Define boundary conditions:**

**inlet**

U = (0, 0, 0.5 m/s)
CO = 0.02 (mass fraction)
H2O = 0.02 (mass fraction)
N2 = 0.96 (mass fraction)
$\nabla p = 0$
$\nabla T = 0$



**inertWall**

U = (0, 0, 0 m/s)
$\nabla \omega = 0$
$\nabla p = 0$
$\nabla T = 0$

**reactingWall**

U = (0, 0, 0 m/s)
Het. Chemistry for species
$\nabla p = 0$
$\nabla T = 0$

**outlet**

$\nabla U = 0$
$\nabla \omega = 0$
p = 1 atm
$\nabla T = 0$

# Case study – Setting up input files

- **Translate the boundary & initial conditions into catalyticFoam input files**

**pressure**

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      p;
}
// * * * * * * * * * * * * * * * * * * * * * * * * *

dimensions      [1 -1 -2 0 0 0 0];

internalField   uniform 101325;

boundaryField
{
    reactingWall
    {
        type            zeroGradient;
    }
    inertWall
    {
        type            zeroGradient;
    }
    inlet
    {
        type            zeroGradient;
    }
    outlet
    {
        type            fixedValue;
        value           uniform 101325;
    }
}
```

**water**

```
FoamFile
{
    version     2.0;
    format      binary;
    class       volScalarField;
    location    "0";
    object      O2;
}
// * * * * * * * * * * * * * * * * * * * * * * * * *

dimensions      [0 0 0 0 0 0 0];

internalField   uniform 0.02;

boundaryField
{
    inlet
    {
        type            fixedValue;
        value           uniform 0.02;
    }
    outlet
    {
        type            zeroGradient;
    }

    inertWall
    {
        type            zeroGradient;
    }
    reactingWall
    {
        type            catalyticWall;
    }

}
```

# Case study – Setting up input files

- **Select physical model in constant/solverOptions**

```
Kinetics
{
        folder                          "WGS_UBI/kinetics";  // path to precompiled kinetic
        inertSpecies                    N2;                  // inert species
}

PhysicalModel
{
        // Operator splitting algorithm
        strangAlgorithm                 "TransportReactionMomentum";

        // Switch for homogeneous and heterogeneous chemistries
        homogeneousReactions            off;
        heterogeneousReactions          on;

        // Catalytic load (m2_cat/m2_geom)
        alfaCatalyst                    5.0;

        // Name of reacting patches
        catalyticWalls (reactingWall);

        // Switch for energy equation
        energyEquation                          off;
        reactionHeatFromHeterogeneousReactions  on;

        // Consider the catalytic cell as constant pressure | volume (true | false)
        constPressureBatchReactor               true;

        // Consider mass diffusion in energy equation
        massDiffusionInEnergyEquation   on;

        // Physical model for diffusion
        diffusivityModel                "multi-component";              // lewis-numbers

        // Consider DpDt term in energy balance
        includeDpDt                     off;

        // Consider Soret's effect
        soretEffect                     off;
}
```

# Case study – Simulation

- **Running the simulation**

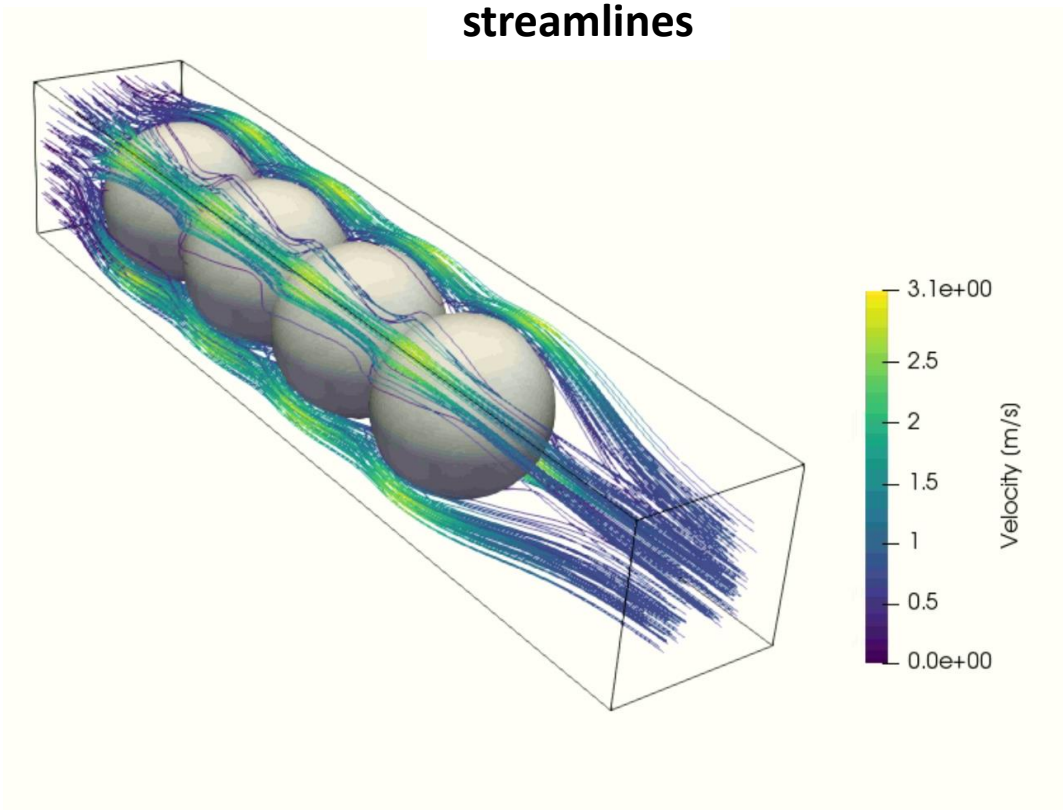
***catalyticPimpleFoam***

```
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  5.x                                   |
|   \\  /    A nd           | Web:      www.OpenFOAM.org                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
Build  : 5.x-7f7d351b741b
Exec   : catalyticPimpleFOAM
Date   : Jul 15 2020
Time   : 13:11:11
Host   : "mauro-OptiPlex-7020"
PID    : 15906
I/O    : uncollated
Case   : /home/mauro/Simulations/Solver/githubs/catalyticFoam/example/sphere
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 10)
allowSystemOperations : Allowing user-supplied system call operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time

Create mesh for time = 0


PIMPLE: Operating solver in PISO mode
```

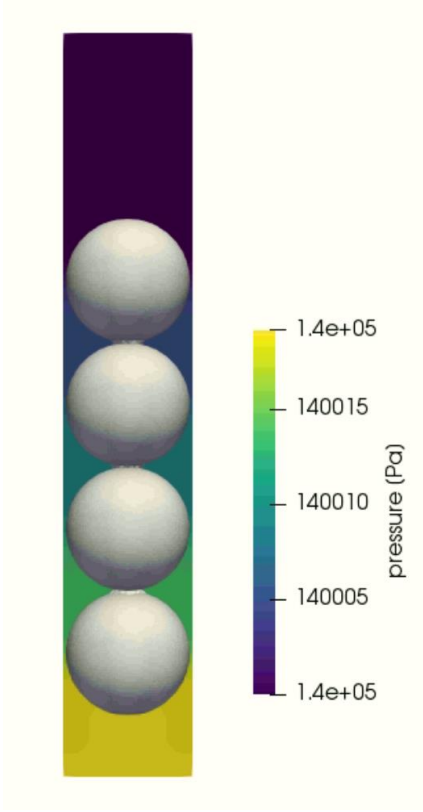
# Case study – Results

- **Results are post-processed with paraview software (or by python postprocessing of VTK files)**

**Velocity streamlines**



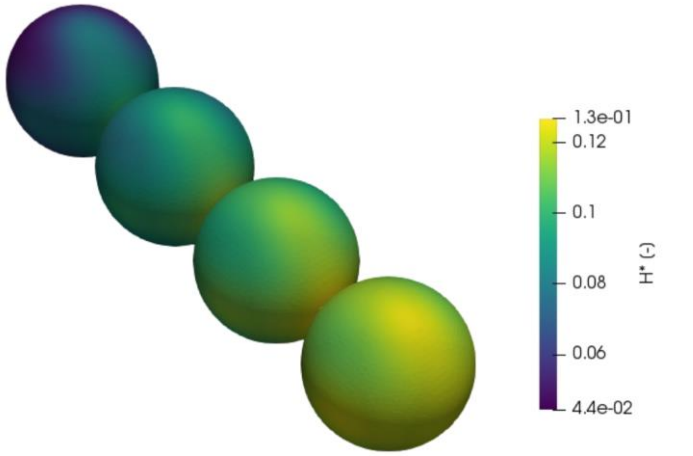

**pressure**



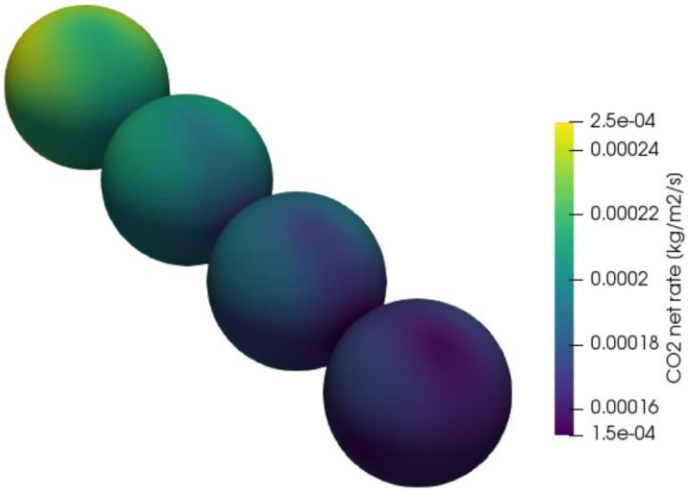
# Case study – Results

- **Results are post-processed with paraview software (or by python postprocessing of VTK files)**

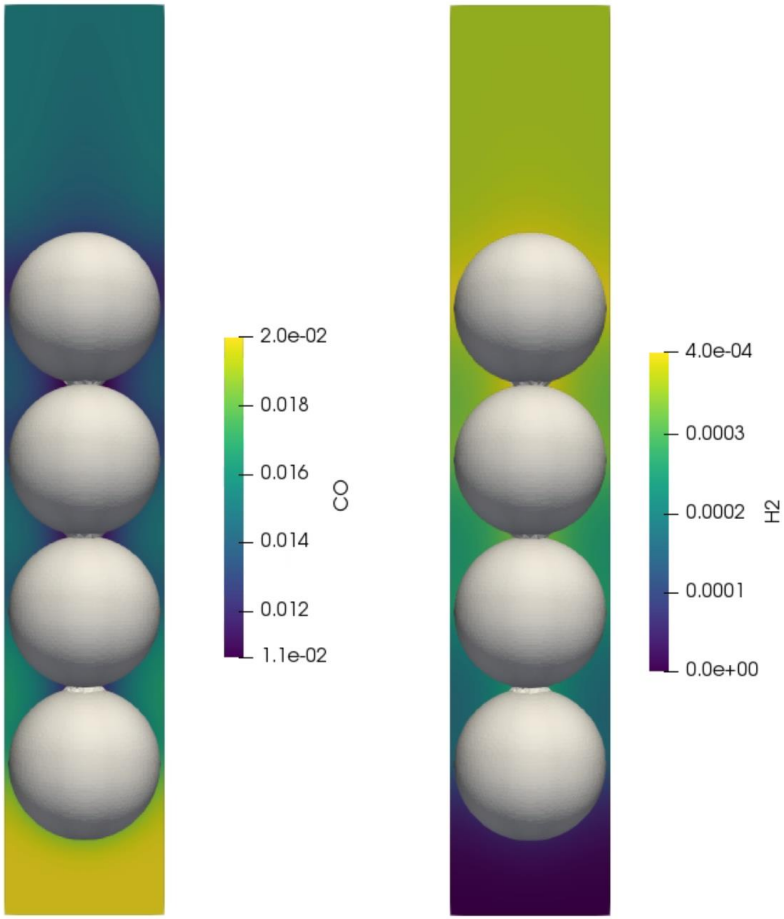
**H* coverage**




**$CO_2$ net production rate**



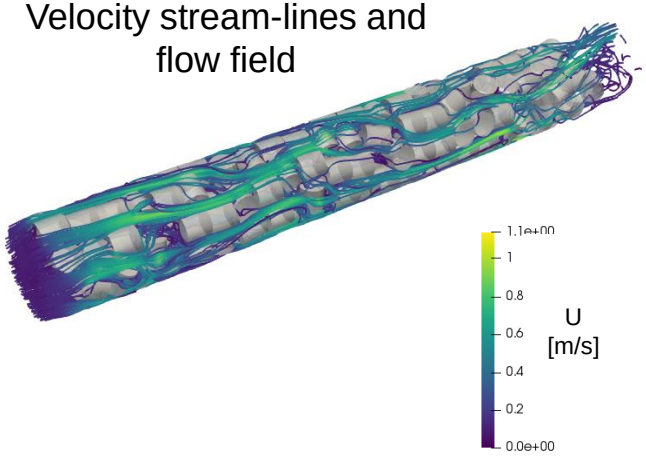

**Mass fractions**

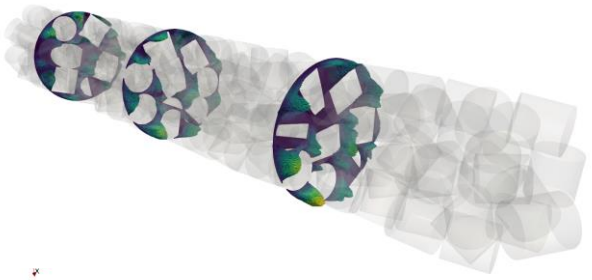


# Case study – Packed bed for methanol synthesis

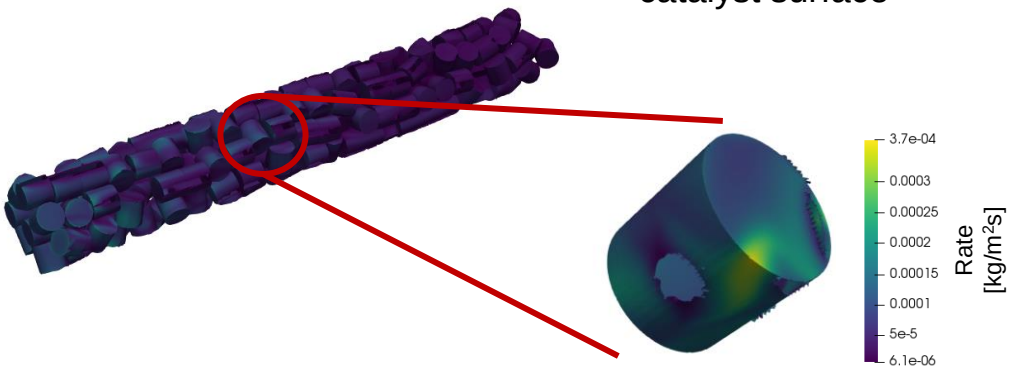
**catalyticFoam handles more complex geometry!**

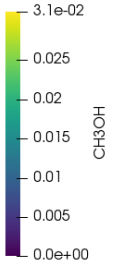
Velocity stream-lines and flow field



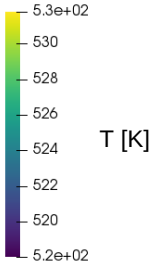



Methanol production rate on the catalyst surface



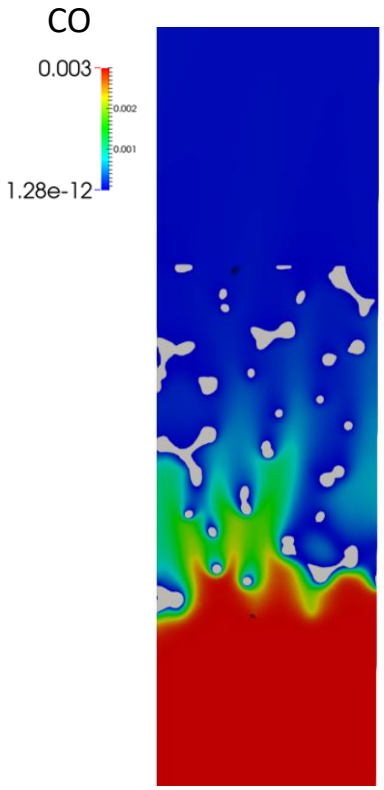

Gas phase maps

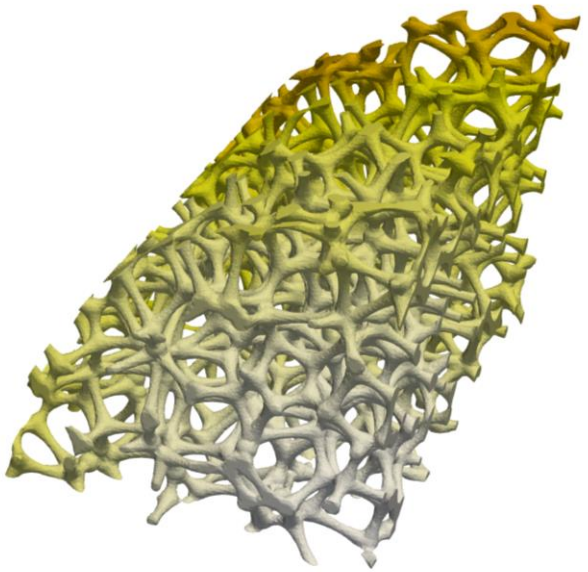



**Methanol synthesis**
**p = 30 bar**
**$T^0$ = 518 K**
**CO:CO2:H2 = 0.05:0.10:0.55**
**191 cylindrical pellets**

# Case study – Catalytic Oxidation of CO on open-cell foams

**CO oxidation in adiabatic tubolar reactor with fluid-solid region coupling**



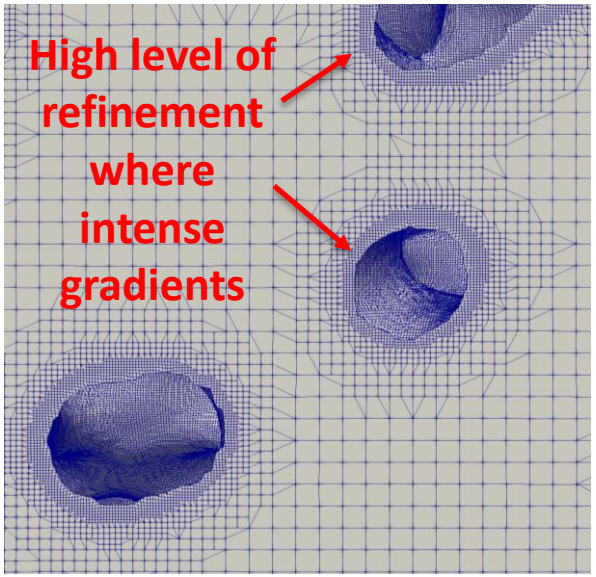
# Case study – mesh generation

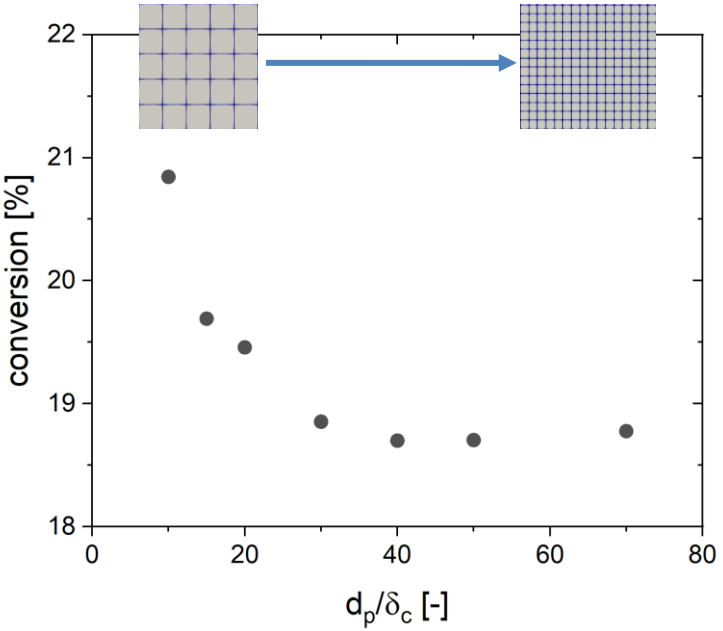
**Mesh generation is crucial for achieving physically sounds results**

- Mesh has to be refined where the gradients are expected to be present
- Prism layers might be necessary to fully describe boundary layer phenomena

**Mesh convergence analysis is pivotal achieve meaning full results**

- Several meshes with different cell size have to be employed to compute the target quantities (e.g., conversion)
- The correct mesh resolution is achieved when the results are independent from the discretization




# Effective tabulation of kinetic models via Machine Learning

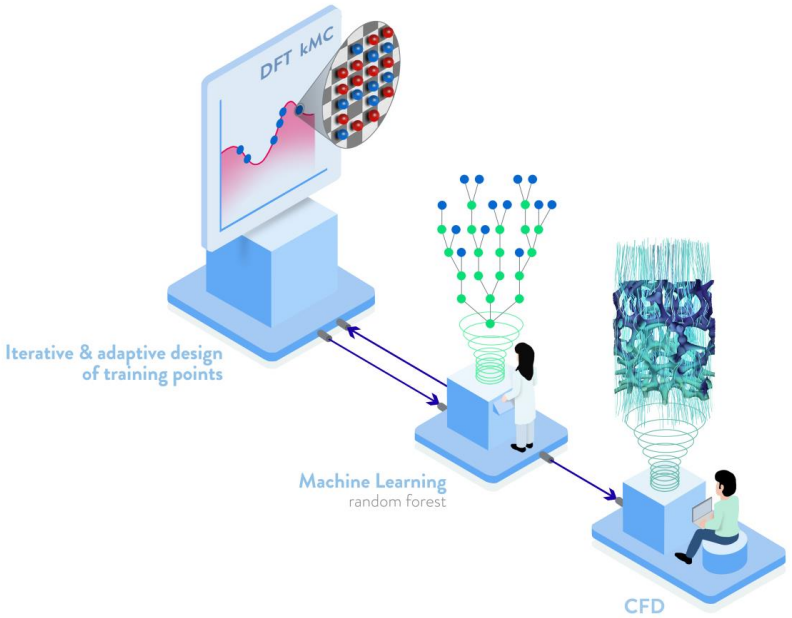
Mauro Bracconi – Matteo Maestri

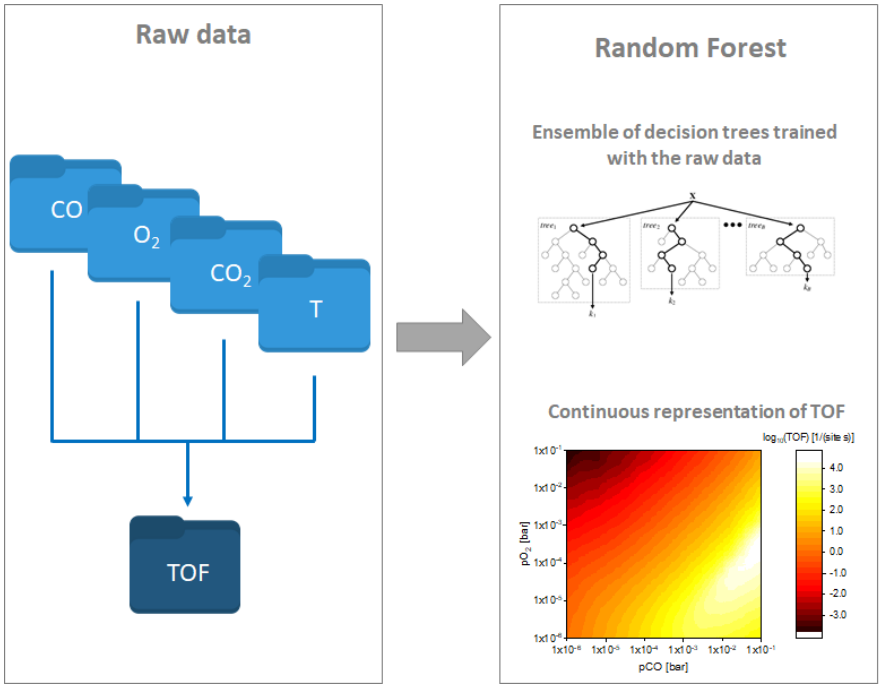
POLITECNICO MILANO 1863

# Case study – adaptive design procedure

**Machine Learning (ML) is employed for the effective tabulation complex and large kinetic models**

**Adaptive design procedure** of training points is employed for the generation of the dataset for the ML model generation

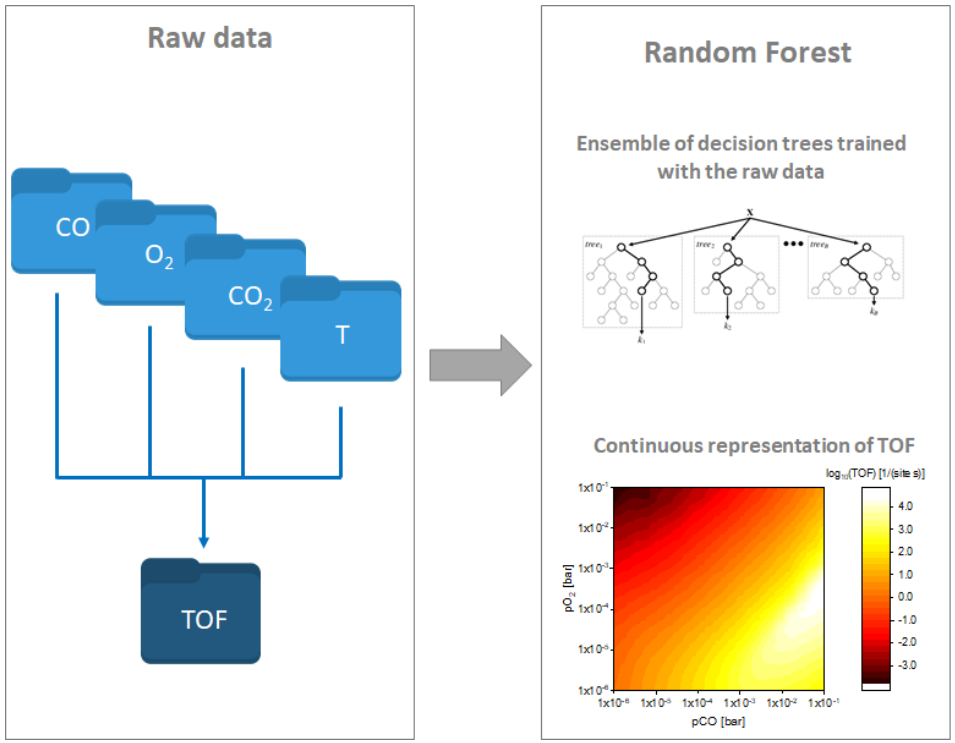


# Case study – adaptive design procedure

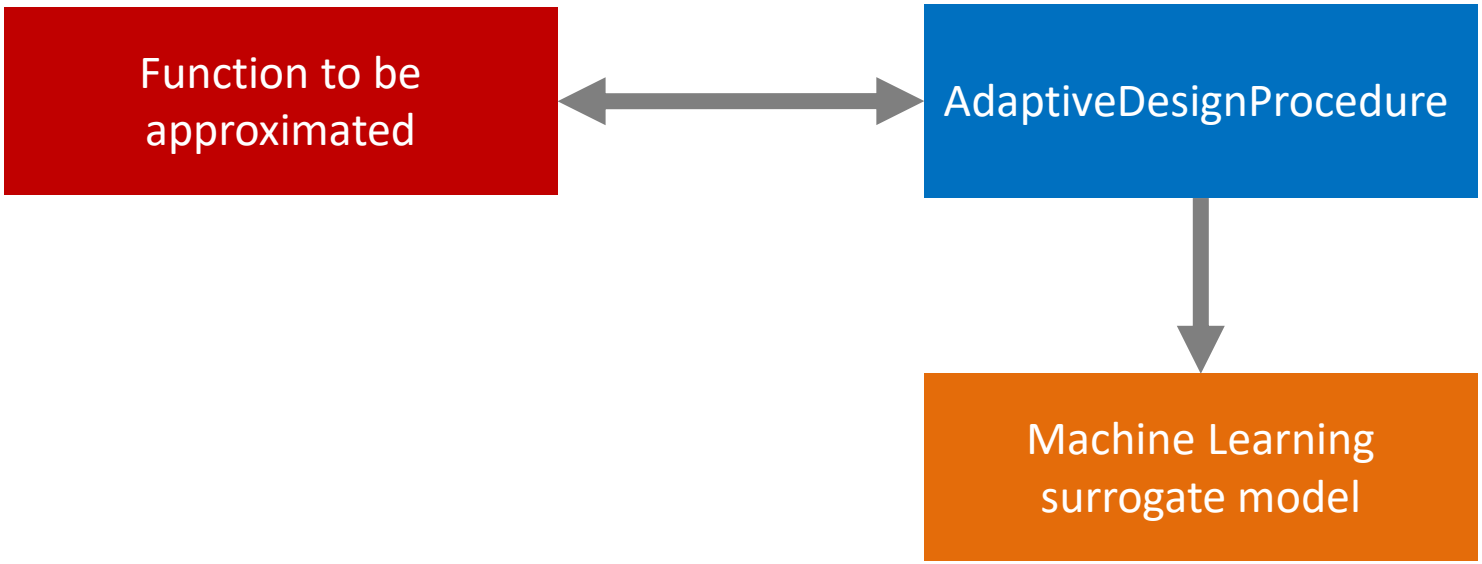
**AdaptiveDesignProcedure is available on github**
**(**https://github.com/multiscale-catalysis-polimi/adaptiveDesignProcedure**)**

**It can be employed for the tabulation of complex and expensive functions such as microkinetic models, kinetic Monte Carlo**



# Case study – adaptive design procedure

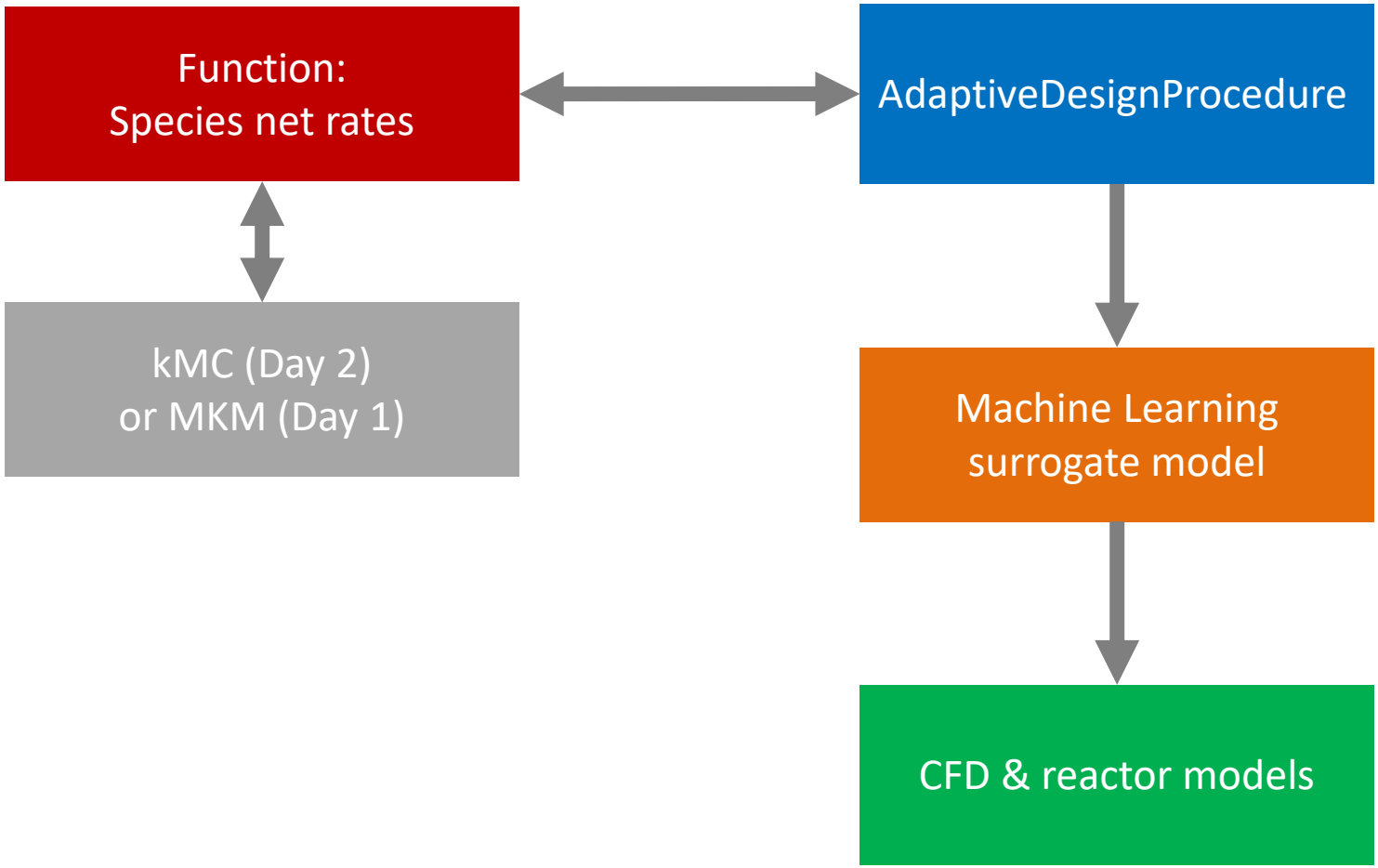
**A simple example of the usage of the procedure is available on github as a jupyter notebook**




M.Bracconi & M. Maestri, *Chemical Engineering Journal*, 400 (2020), 125469

# Case study – adaptive design procedure

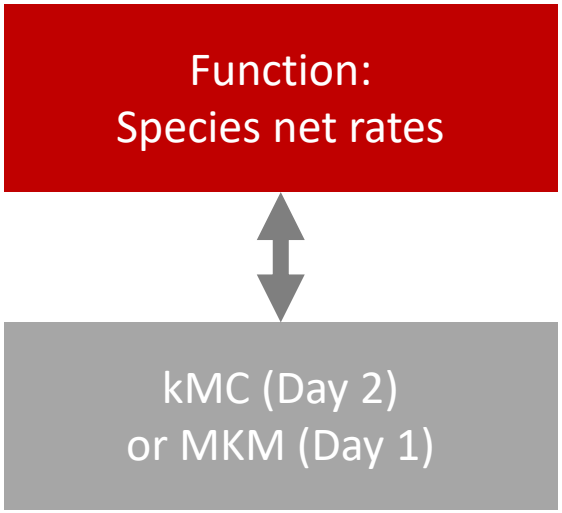
**The AdaptiveDesignProcedure is readily coupled with microkinetic models and kMC simulations**




M.Bracconi & M. Maestri, *Chemical Engineering Journal*, 400 (2020), 125469

# Case study – adaptive design procedure

**Create a getRate function able to run kMC simulations and post-process the relevant results**




```
def getRate(x):
    """Run kMC simulations according to a specific set of input (x),
       post-process results and return the corresponding information

        Parameters
        ----------
            x : np.array[number records, number input variables]
                Input data: set of operating conditions

        Return
        ----------
            y : np.array[number records, number tabulation variables]
                Function values: kMC information

    """

    #Pseudo-code
    # for every input x
    #    i/ run kMC simulation (Zacros) with that set of operating conditions
    #    ii/ post-process results
    #    iii/ compute quanity of interest and return
```

# Case study – adaptive design procedure – kMC example

**Setup input & output variables according to the procedure nomenclature**

**Target: tabulation of kMC CO oxidation over $RuO_2(110)$ [1] TOF for isothermal simulations at 600 K for an interval of partial pressures between [1e-6; 1e-1] for both CO and $O_2$**

**Initial dataset: Each variable is initially discretized with 3 points**

[1] K. Reuter & M. Scheffler, *Physical Review B*, 73 2006

# Case study – adaptive design procedure – kMC example

**Setup input & output variables according to the procedure nomenclature**

**Target: tabulation of kMC CO oxidation over $RuO_2$(110) [1] TOF for isothermal simulations at 600 K for an interval of partial pressures between [1e-6; 1e-1] for both CO and $O_2$**

**Initial dataset: Each variable is initially discretized with 3 points**

AdaptiveDesignProcedure

```
# Parameters to build an example kMC RF table
# Forest paramters
forestParams={
        'Ntree'       : 20,
        'tps'         : 1,
        'fraction'    : 0.8,
    }

# Algorithm paramters
algorithmParams={
        'dth'         : 0.5,     # thresold first derivative
        'd2th'        : 1,     # thresold second derivative
        'VIth'        : 0.15,    # thresold variable importance
        'errTh'       : 1e-6,    # thresold for MRE error evaluation (remove from MRE calculation record below this value)
        'OOBth'       : 0.1,    # termination criterium on OOBnorm
        'RADth'       : 60,      # termination criterium on Relative Approximation Error (RAD) [%]
        'maxTDSize'   : 40000,   # maximum allowed size of the training data
        'AbsOOBTh'    : 0.2,     # maximum variations between OOB for two different tabulation variables
    }

# Files (input, training, query and benckmark)
trainingFile    = 'train.dat'
forestFile      = 'ml_kMC_COox.pkl'
queryFile       = 'query_input_kmc.dat'
queryRest       = 'query_output_kmc.dat'

# Independent variables (i.e., descriptors)
input_var = ( { 'name' : 'CO', 'min' : 1e-6, 'max' : 0.1, 'num' : 3, 'typevar' : 'log'},
              { 'name' : 'O2', 'min' : 1e-6, 'max' : 0.1, 'num' : 3, 'typevar' : 'log'})

# Tabulated variables
tabulation_var = (  {'name' : 'TOF', 'typevar' : 'log'}, )
```

[1] K. Reuter & M. Scheffler, *Physical Review B*, 73 2006

# Case study – adaptive design procedure – kMC example

**Create the AdaptiveDesignProcedure object and start the generation of the dataset and surrogate model!**

AdaptiveDesignProcedure

```
# Initialize ADPforML class
adpML = adp.adaptiveDesignProcedure(input_var,
                                    tabulation_var,
                                    forestFile,
                                    trainingFile,
                                    forestParams,
                                    algorithmParams,
                                    getRate,
                                    queryFile,
                                    queryRest)

# Create training and RF
adpML.createTrainingDataAndML()
```

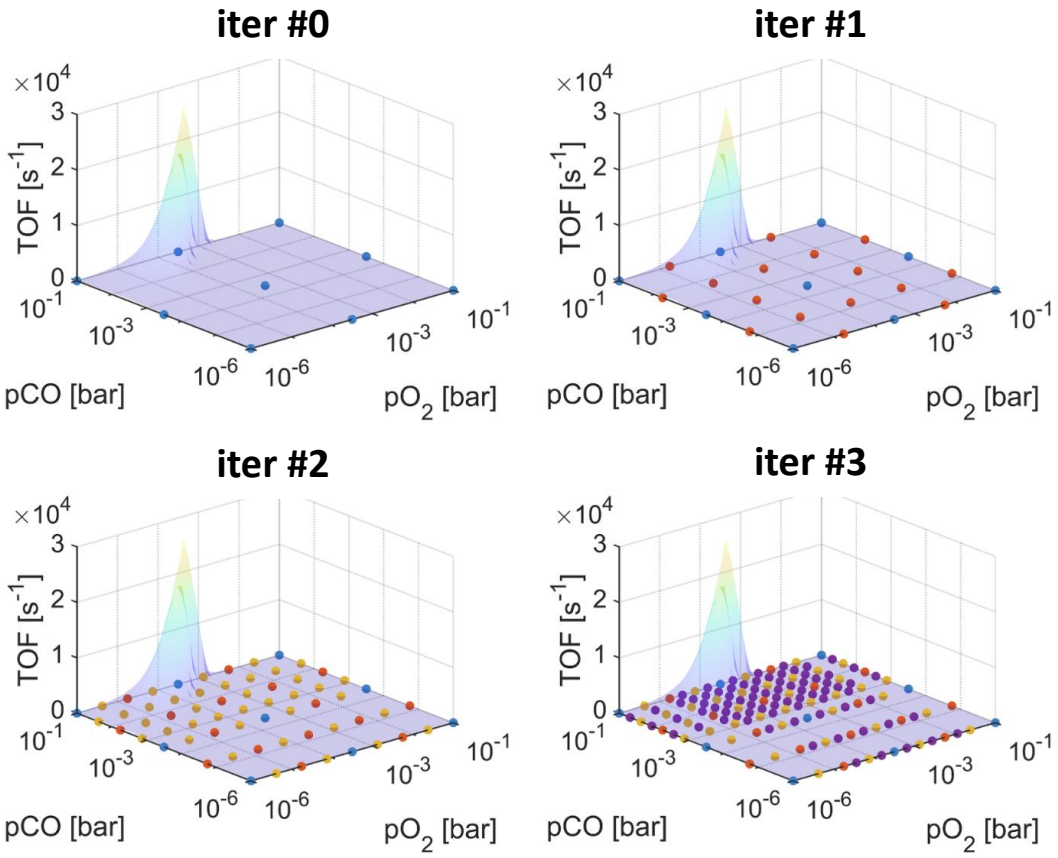
**At the end, the procedure will provide a ML surrogate model of the system in terms of ExtraTrees**
**The ML model is saved as a pickle file to be easily copied, transfered and archived.**

**The ML model is coupled with reactor models & CFD simulations**

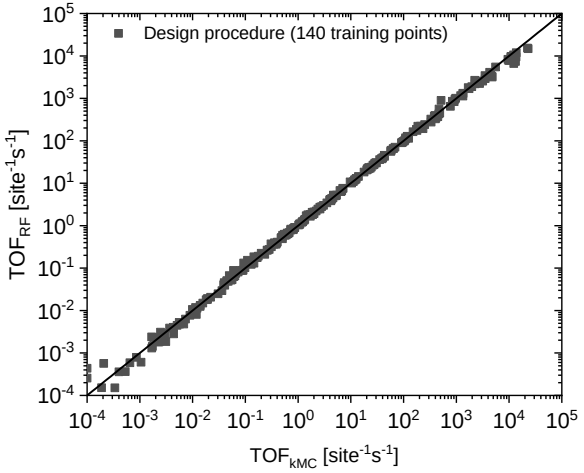
M.Bracconi & M. Maestri, *Chemical Engineering Journal*, 400 (2020), 125469

# Case study – adaptive design procedure – kMC results

**The iterative procedure leads to the following dataset evolution**



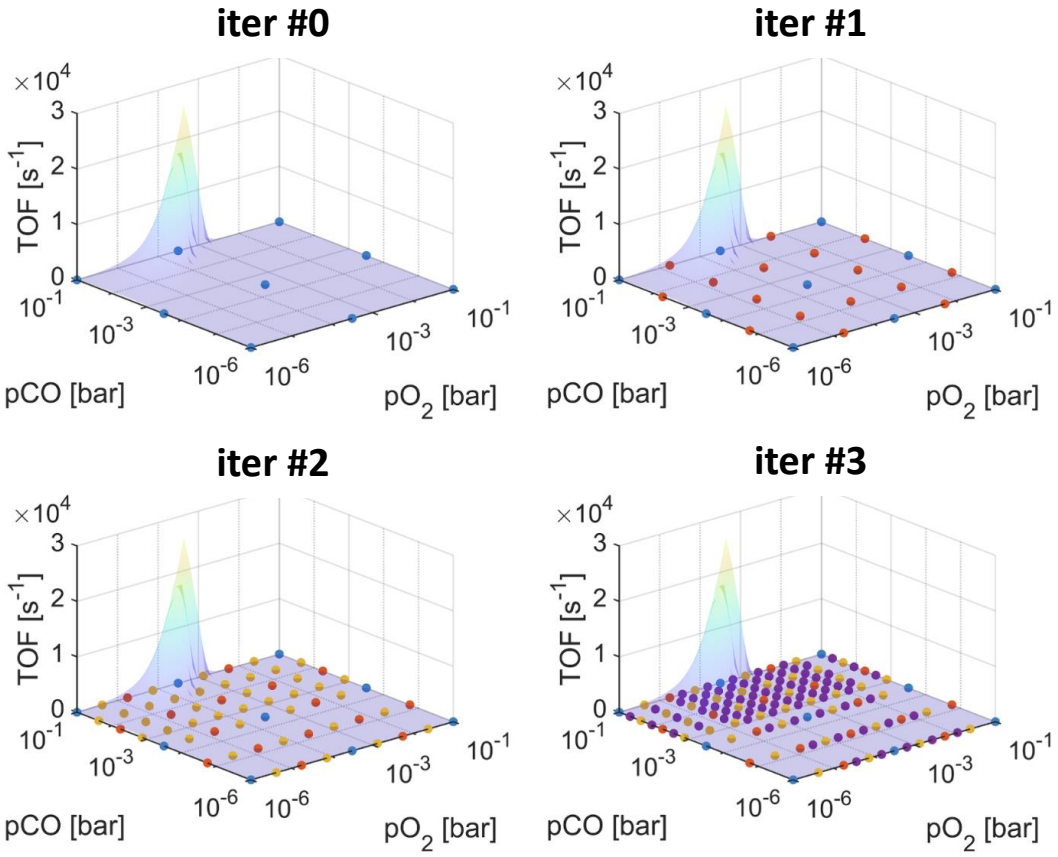

**TOF parity plot on benchmark data**



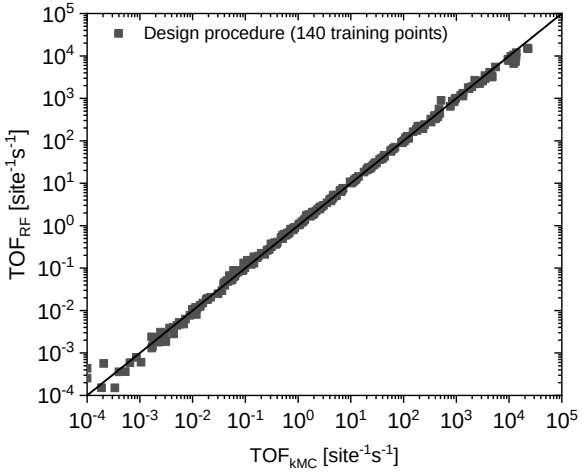

M.Bracconi & M. Maestri, *Chemical Engineering Journal*, 400 (2020), 125469

# Case study – adaptive design procedure – kMC results

**The iterative procedure leads to the following dataset evolution**




**TOF parity plot on benchmark data**

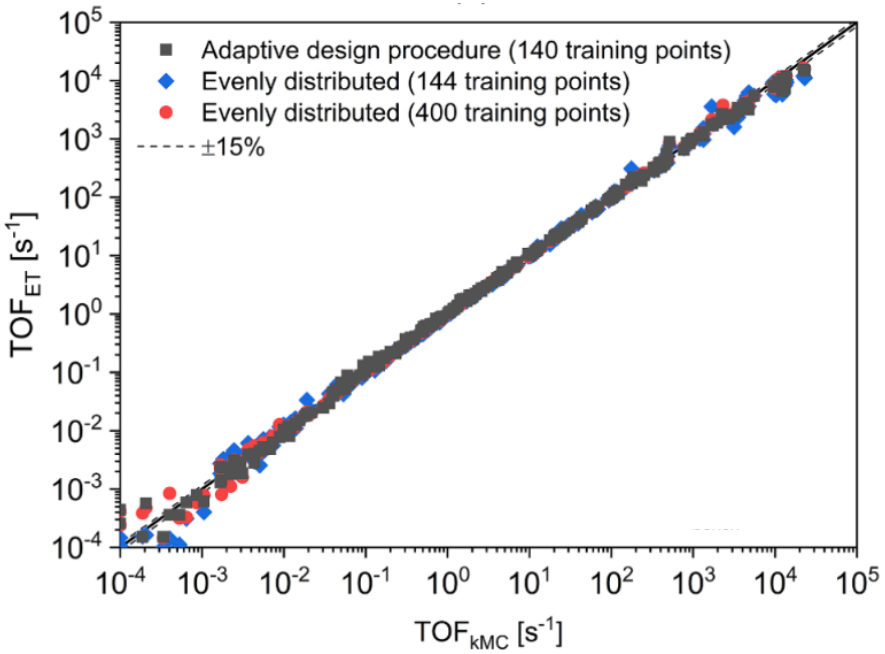


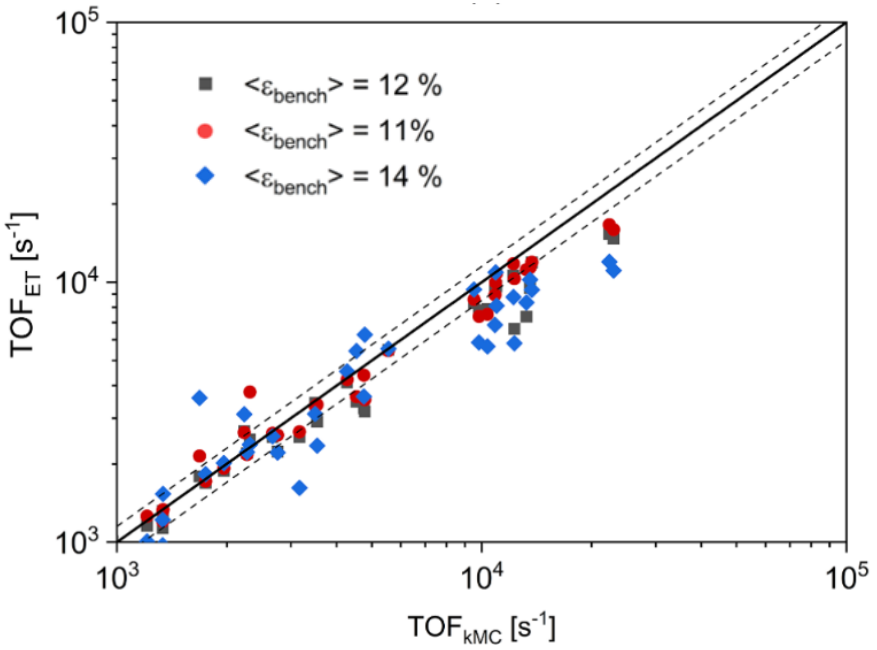

**How much is the actual improvement in accuracy and time due to the procedure?**

M.Bracconi & M. Maestri, *Chemical Engineering Journal*, 400 (2020), 125469

# Case study – adaptive design procedure – kMC results

**AdaptiveDesignProcedure leads to a benefit in the accuracy even with this low-dimensional system**

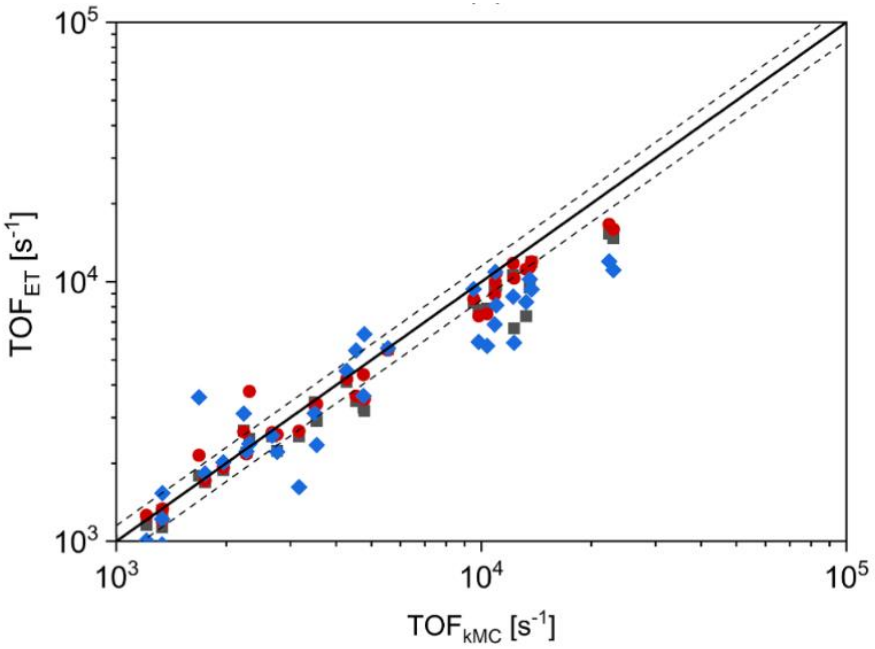


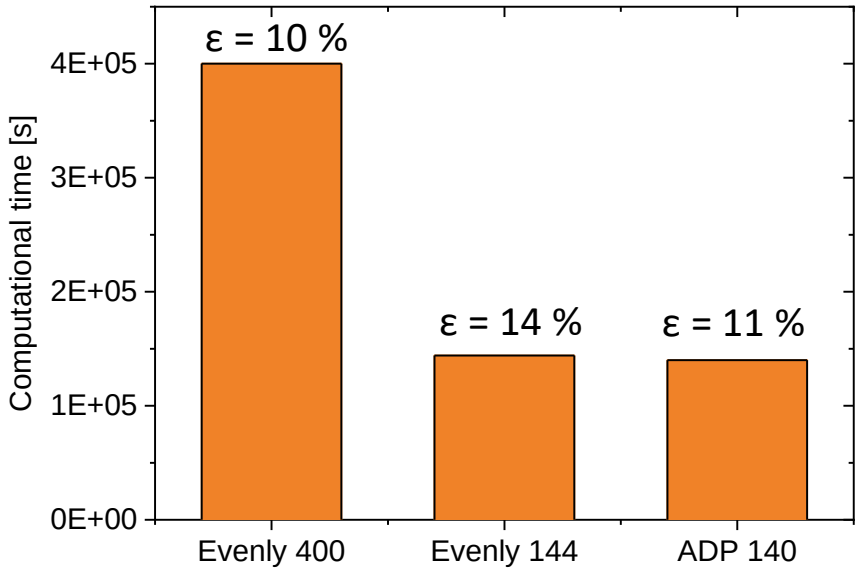




M.Bracconi & M. Maestri, *Chemical Engineering Journal*, 400 (2020), 125469

# Case study – adaptive design procedure – kMC results

**Significant reduction of the computational cost required for the generation of the ML model. Same accuracy with 65 % less computational time**



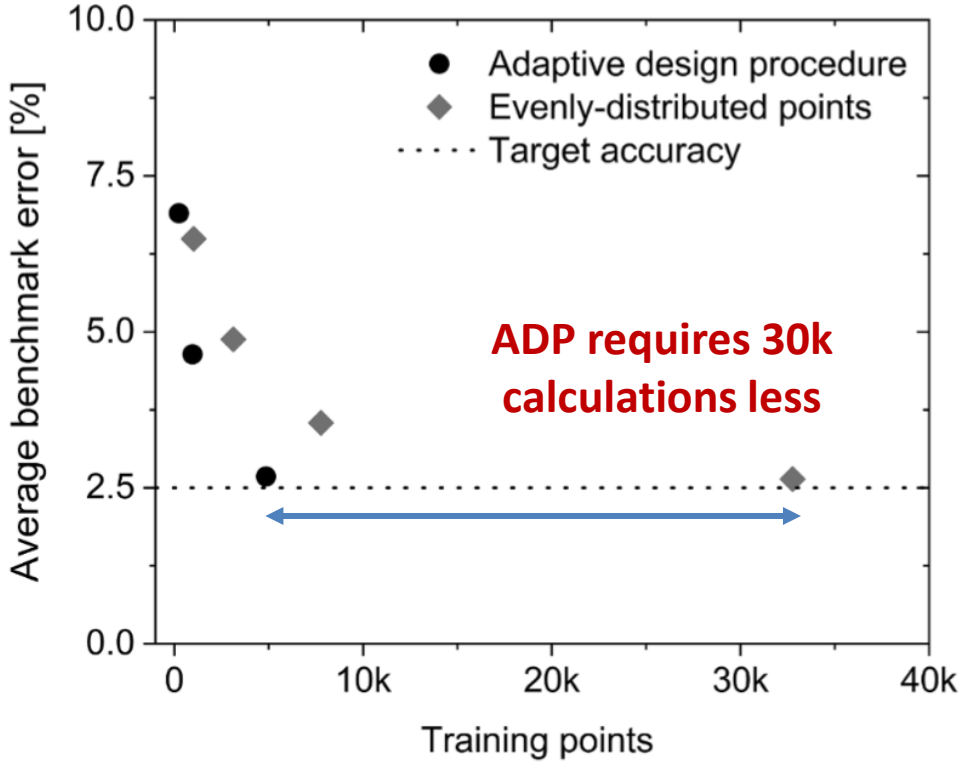




M.Bracconi & M. Maestri, *Chemical Engineering Journal*, 400 (2020), 125469

# Case study – adaptive design procedure

**High-dimensional systems reach higher performances due to accurate positioning of the training points**

**Target: MKM WGS on Rh [1] non-isothermal conditions**




| Variable | Range |
|---|---|
| H2 | $1 \cdot 10^{-6}$ - $1 \cdot 10^{-3}$ |
| CO2 | $1 \cdot 10^{-6}$ - $1 \cdot 10^{-3}$ |
| CO | $1 \cdot 10^{-2}$ - $1 \cdot 10^{-1}$ |
| H2O | $1 \cdot 10^{-2}$ - $1 \cdot 10^{-1}$ |
| T | 500 – 650 K |

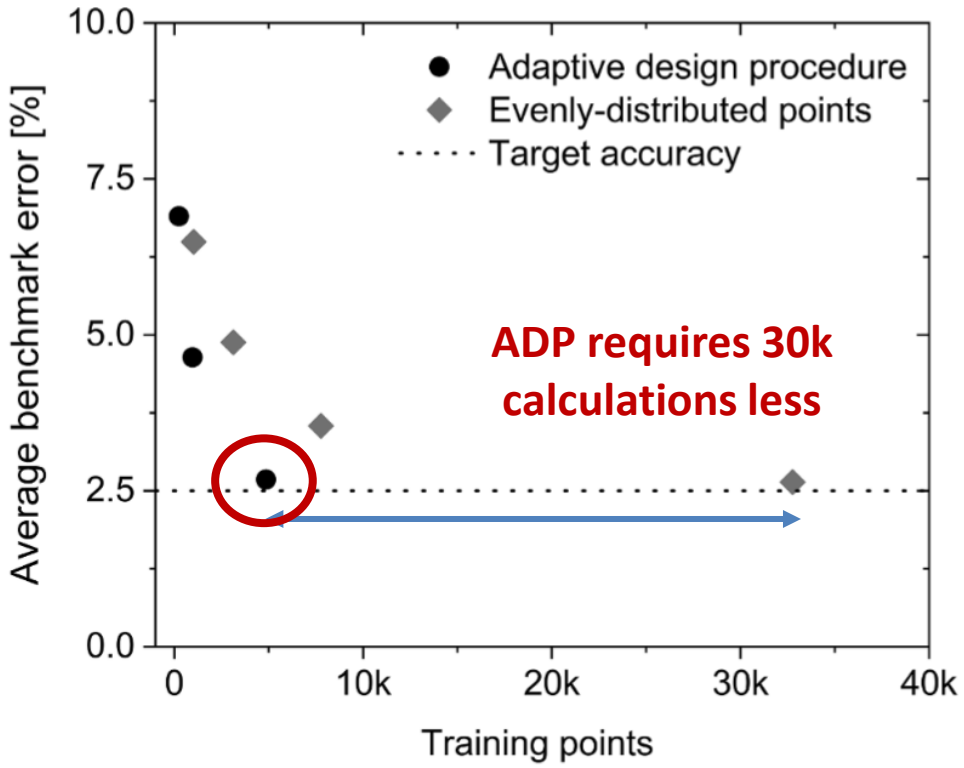
**How are the points distributed?**

[1] M. Maestri et al., *AIChE Journal*, 55 2009

# Case study – adaptive design procedure

**High-dimensional systems reach higher performances due to accurate positioning of the training points**

**Target: MKM WGS on Rh [1] non-isothermal conditions**




| Variable | Var. mp. | Points |
|---|---|---|
| CO | 0.74 | 6 |
| H2O | 1.00 | 6 |
| H2 | 0.33 | 5 |
| CO2 | 0.06 | 3 |
| T | 0.96 | 9 |

[1] M. Maestri et al., *AIChE Journal*, 55 2009

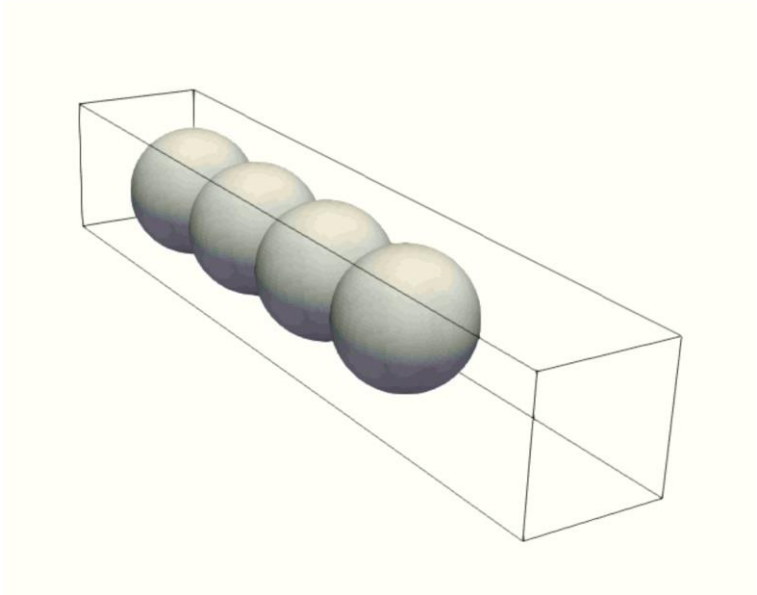

# Effective tabulation of kinetic models via Machine Learning

Mauro Bracconi – Matteo Maestri

POLITECNICO MILANO 1863

# Case study

## CFD simulation of CO oxidation on $RuO_2(110)$ in a 4-spheres string reactor



**Operating conditions**:

- Isothermal simulation (600 K)
- $CO/O_2$ = 5:1 – CO = 5 % (mass fraction)
- $N_2$ inert
- Flowrate = 2.4 NmL/s
- Laminar conditions
- Pellet size = 4 mm

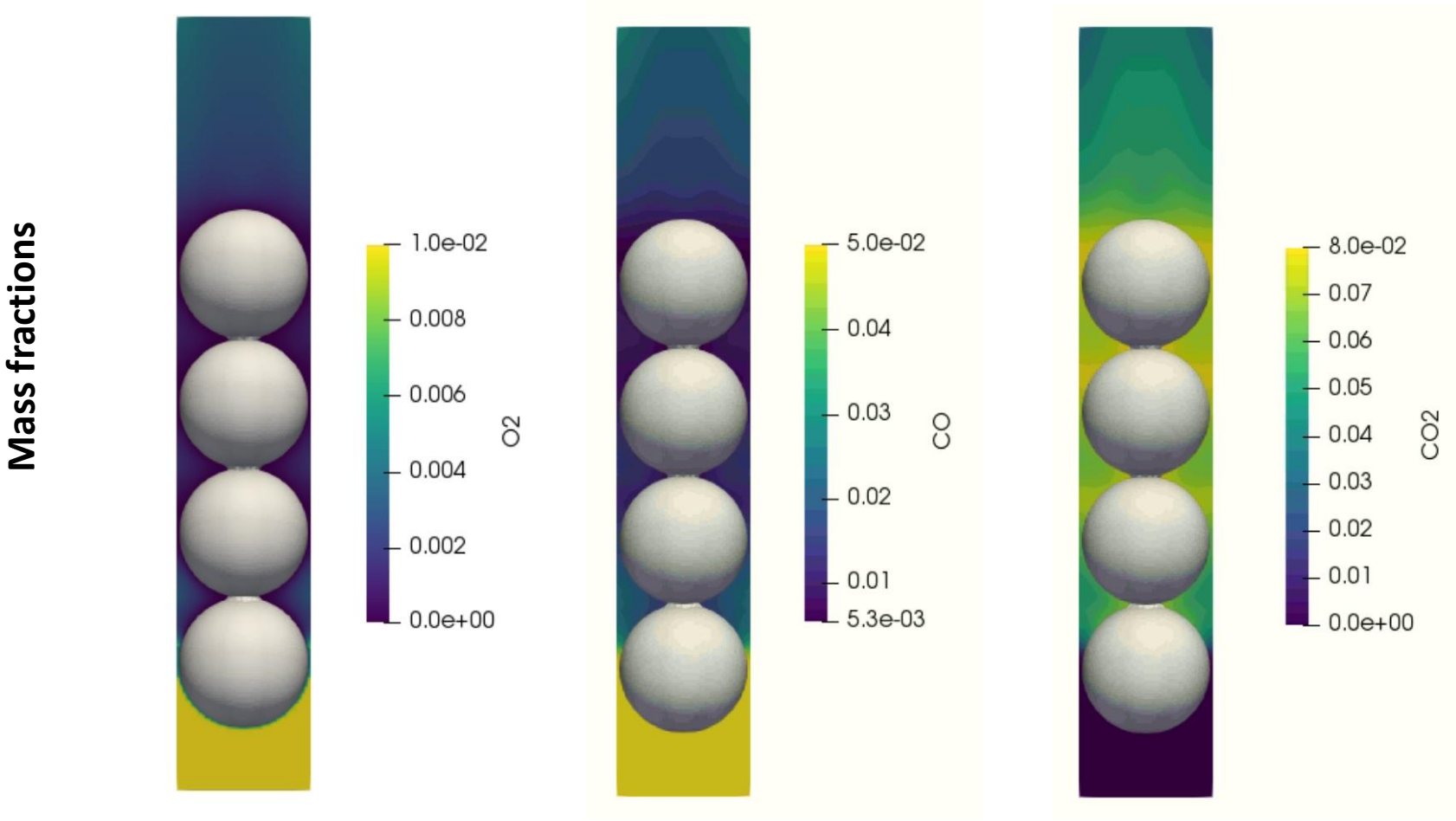
**Kinetic model [1]**:

- kinetic Monte Carlo model
- 3 gas species
- 2 adsorbed species
- 2 active sites (brg | cus)

[1] K. Reuter & M. Scheffler, *Physical Review B*, 73 2006

# Case study – Results

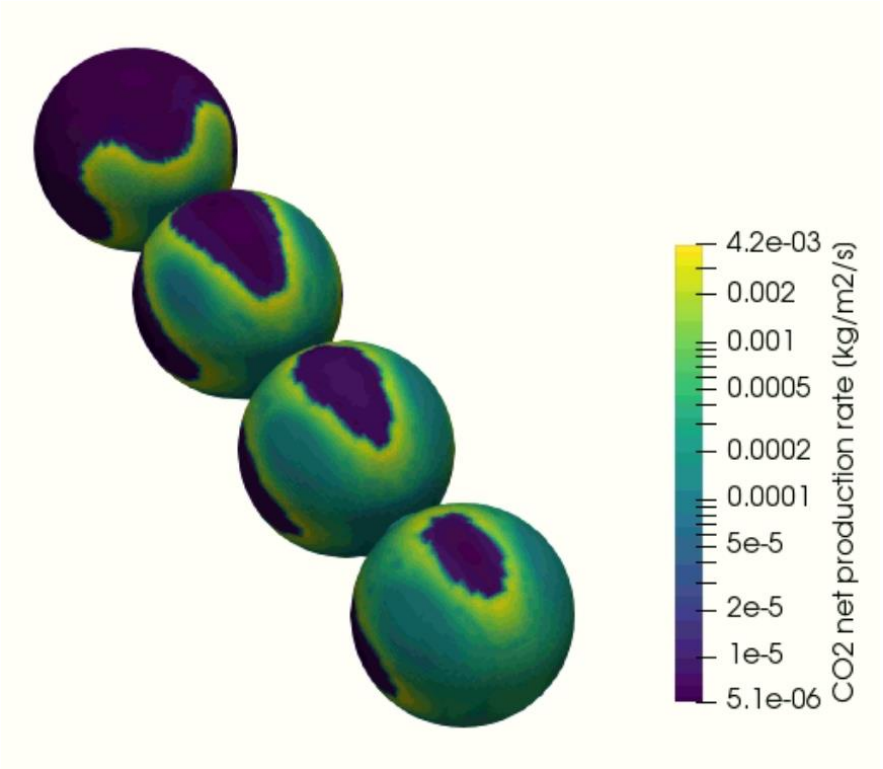
- **Results are post-processed with paraview software (or by python postprocessing of VTK files)**



# Case study – Results

- **Results are post-processed with paraview software (or by python postprocessing of VTK files)**

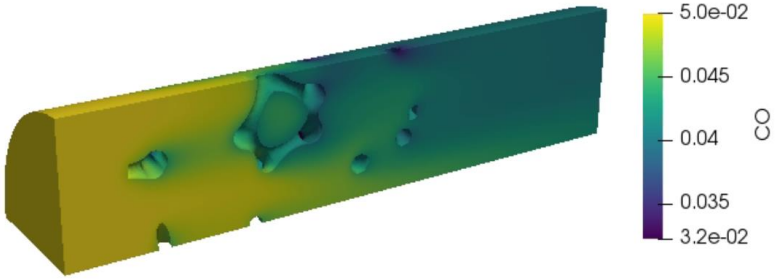
**$CO_2$ production rate**

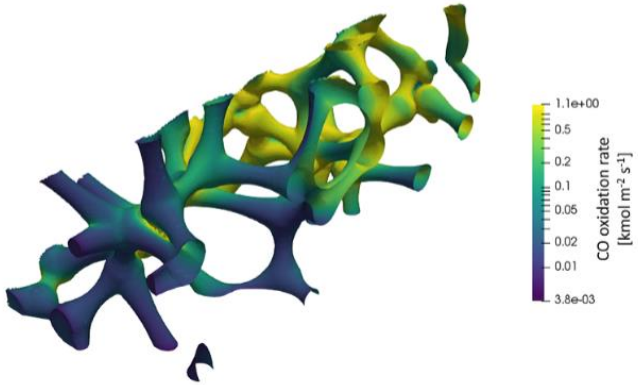


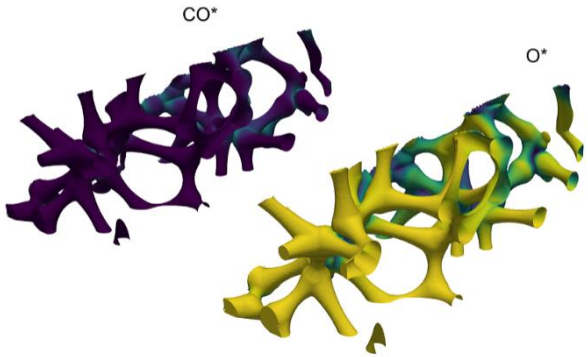
# Case study – More complex geometry

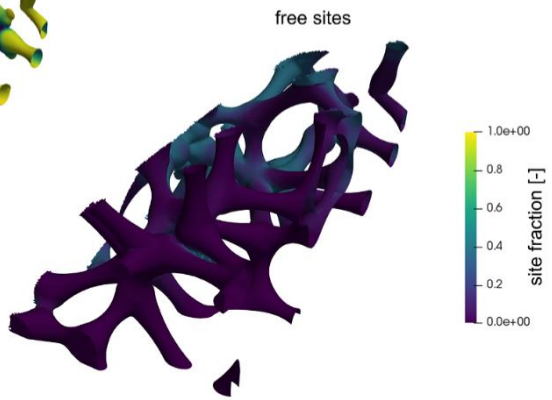
- kMC + CFD in more complex geometries are possible
- ML can tabulate also the local coverages from kMC simulations

**Open-cell foams with CO oxidation over $RuO_2(110)$**




M.Bracconi & M. Maestri, *Chemical Engineering Journal*, 400 (2020), 125469

# Tools

**https://github.com/multiscale-catalysis-polimi**

**catalyticFoam:**
**https://github.com/multiscale-catalysis-polimi/catalyticFoam**

**Tools for structure-dependent microkinetic modeling:**
**https://github.com/multiscale-catalysis-polimi/nanoparticles_ensembles**

**Machine Learning adaptive design procedure:**
**https://github.com/multiscale-catalysis-polimi/adaptiveDesignProcedure**

**Polimi-reactor-modeling suite:**
macroscopic reactor models (pseudo-homogeneous/heterogeneous models, intraphase phenomena; 1D-2D models, …)
*** it will be released by the end of the summer ***