Gas injection into LHC

Chiara Lucarelli

Università e INFN, Firenze

Fixed Target experiments at LHC –Strong2020 Workshop,
24 June 2022

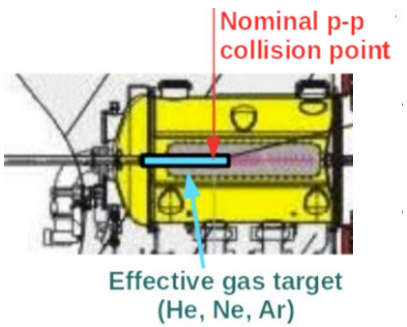
Prelude

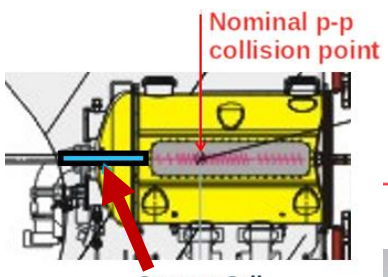
During Run 1-2 LHCb was able to inject small amount of gas in LHC beam pipe around (±20 m) the LHCb IP: **SMOG**.

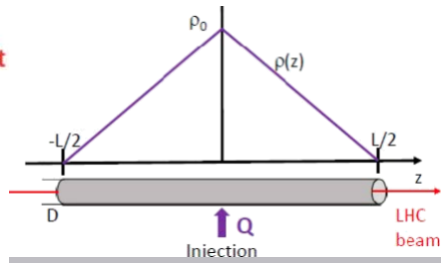
- Only noble gases at low pressure (2×10^{-7} mbar).
- No direct $\mathcal{L}$ measurement due to the lack of precise gauges for injected gas.

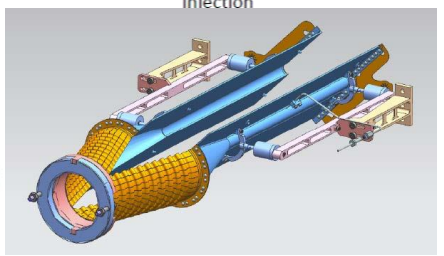
SMOG2: upgrade for Run 3 with the installation of a gas confinement cell upstream the IP in the [-500, -300] mm region:

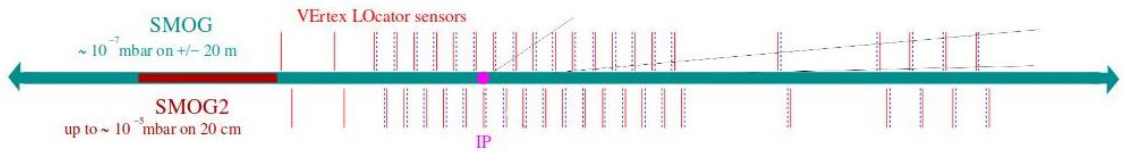
- Higher average gas density (and luminosity).
- Direct and precise gas pressure and temperature measurement.
- More injectable gases.











Prelude

During Run3, SMOG2 plans to inject new gases into the storage cell: H_2, D_2, N_2, O_2, Ar, Kr, Xe (+ He, Ne).

→ Understand and quantify limitations to the gas flux injection.

(Some of the) Open issues:

- *H_2 and D_2*: NEG Embrittlement and sticking coefficient saturation.
- *Getterable gases (N_2 and O_2)*: Sticking coefficient saturation e NEG SEY.

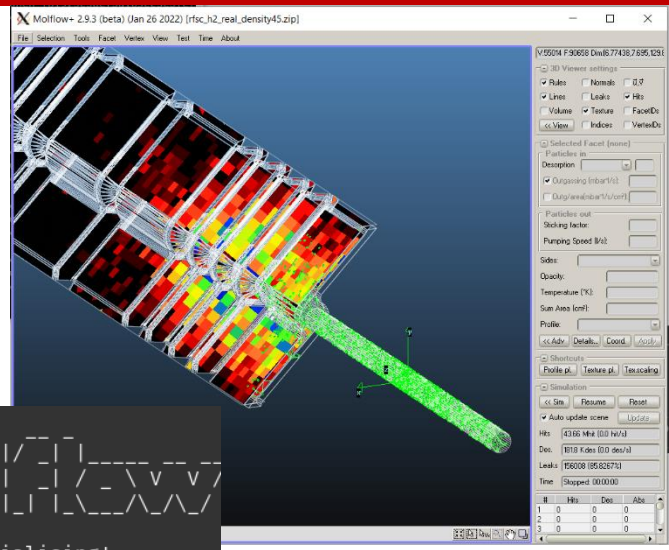
→ Detailed molecular flow simulations needed:

- Detailed local geometry model of SC+RF.
- Update surface properties (e.g. sticking coefficient) dynamically during simulation

Prelude

Already available:

- Molflow+: GUI (geometry manipulation) and CLI (automated control of simulation parameters and flow).
- 3D CAD model of SC+RF (Marton Ady and Roberto Karsevan).



BUT

Molflow+ doesn't allow time dependent simulation with dynamic update of parameters.

- Simulation divided into time steps and parameters updated after each step based on the step results.

  → Python script to automatize sticking coefficient evolution.

Outline

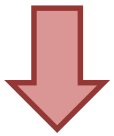
1. **Sticking model:**
    - Problem overview
    - Model determination: Hydrogen-like gases
2. Python script:
3. Data analysis and (preliminary) results

Sticking model: problem overview

NEG coating works as a pump to adsorb molecules on its surface. The sticking coefficient and its evolution depend on the gas and on the surface roughness (surface capacity):

- *H_2 and D_2:* diffuse in the film bulk → sticking coefficient decreases slowly with the concentration of H in the bulk (H_{atom}/NEG_{atom}) and depends on the pressure on the film; an excess of hydrogen in the bulk may result in embrittlement.
- *Getterable gases (CO, N_2 and O_2):* stay on the surface → sticking coefficient decreases with the coverage.

Impurities and partial saturation from a different gas species modify the sticking coefficient → Complex implementation in Molflow+.



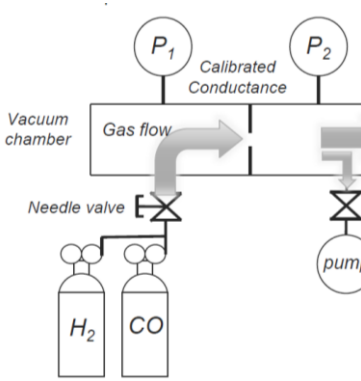
Simulation considers one gas injection at a time starting with NEG coating at “factory conditions” (“intrinsic” sticking coefficient).

Sticking model: hydrogen-like gases

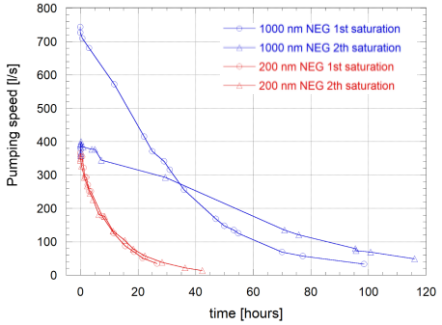
Pedro Costa Pinto Slides from PBC-FT WG

Direct measurement of the evolution of the H_2 sticking coefficient in a Fischer-Mommsen dome:

- The pumping speed of NEG is measured at constant pressure (~2e-6 mbar) for a 1000nm thick NEG (RF foil NEG: ~600 nm).
- The H concentration in the NEG bulk is obtained by integrating over time the flux of H_2 pumped.



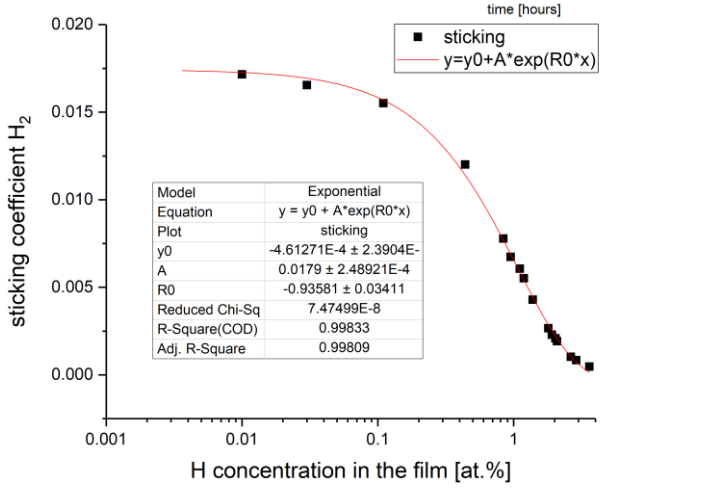
$$S_{NEG} = \frac{(P_1 - P_2)}{P_2} c\text{-}S_{pump}$$



$$S_{H_2} = -4.61 \times 10^{-4} + 1.79 \times 10^{-2} \times e^{-0.936 \times c_H}$$

$c_H \rightarrow H$ *atomic concentration in the NEG film* [at.%]

→ It should be repeated for each gas species + NEG sub, with experimental data.



Outline

1. Sticking model:
    - Problem overview
    - Model determination: Hydrogen-like gases
2. **Python script:**
    - Molflow+
    - Algorithm
3. Data analysis and (preliminary) results

Python script: Molflow+

A brief explanation of Molflow's algorithm:

- Test Particle Monte Carlo method: simulation of virtual test particles (vp). Only collisions with walls (characterized by temperature, opacity, sticking coefficient). Physical quantities derived scaling from virtual to real physical molecules:

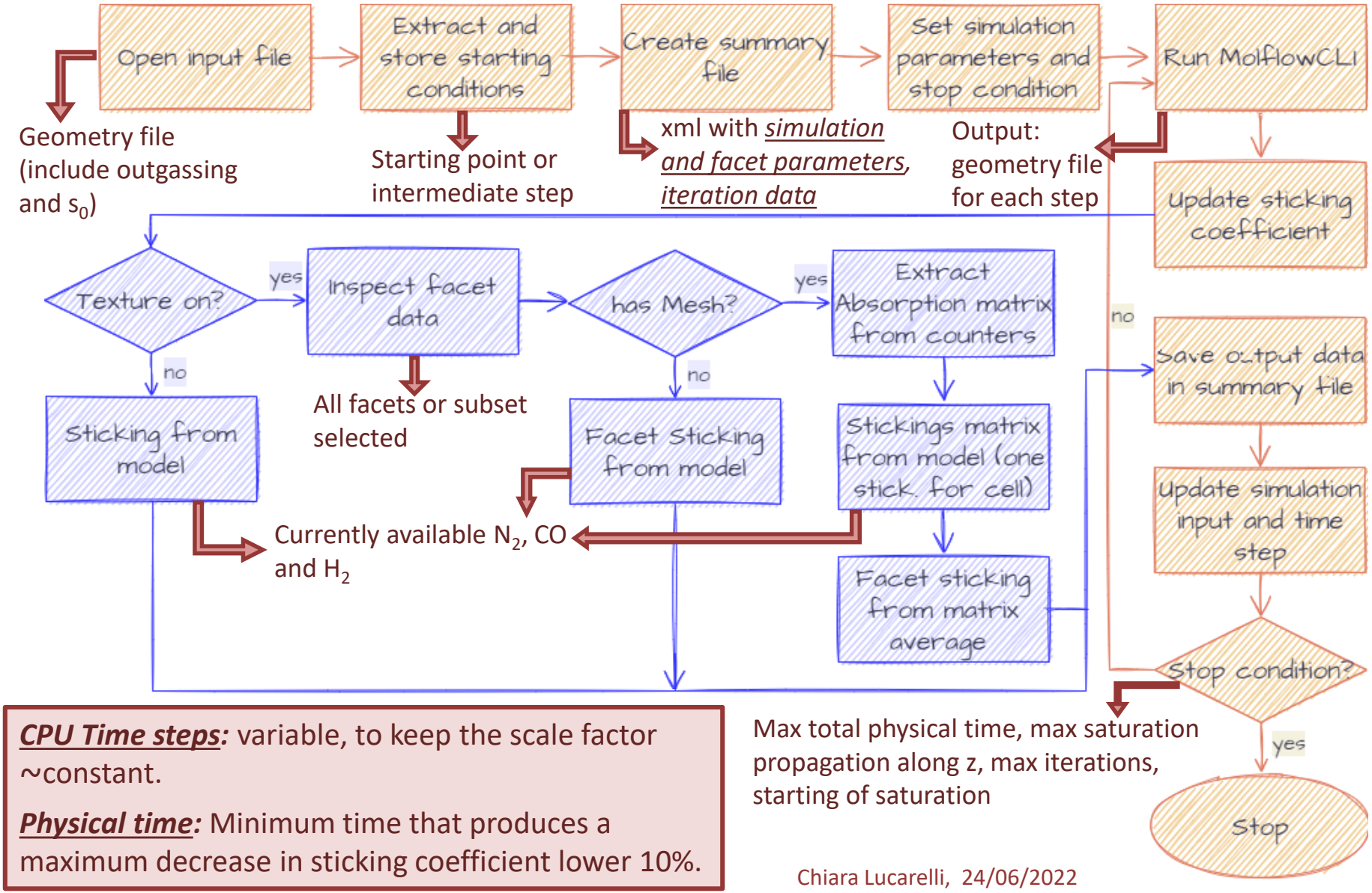
$$\frac{df_{real}}{dt} = scale * f_{vp}, \qquad scale = outgassing\ rate/\#\ desorbed\ vp$$

- Steady-state simulation: simulation of system at equilibrium. Continuous influx of gas particles (constant outgassing rate) and pumping speed.
  - **Only rates are simulated!** Impingement rate, absorption rate, …
  - Absolute quantities (i.e. # absorbed particles by a facet) can be obtained multiplying the rate by an arbitrary time (***physical time***).
- Statistical accuracy of simulation roughly connected to # hits per facets and on the scale factor:

Fix # desorbed vp:
higher #desorbed = lower scale factor, but simulation time can diverge.

Fix simulation time (timeCPU):
longer timeCPU ≈ better statistic, but no real control on scale factor.

Python script: algorithm



CPU Time steps: variable, to keep the scale factor ~constant.

Physical time: Minimum time that produces a maximum decrease in sticking coefficient lower 10%.

Outline

1. Sticking model:
    - Problem overview
    - Model determination: Hydrogen-like gases
2. Python script:
    - Molflow+
    - Algorithm
3. **Data analysis and (preliminary) results**
    - Simulation model
    - Preliminary results for H_2

Model for first results

First simulation on Cell+RF foil:

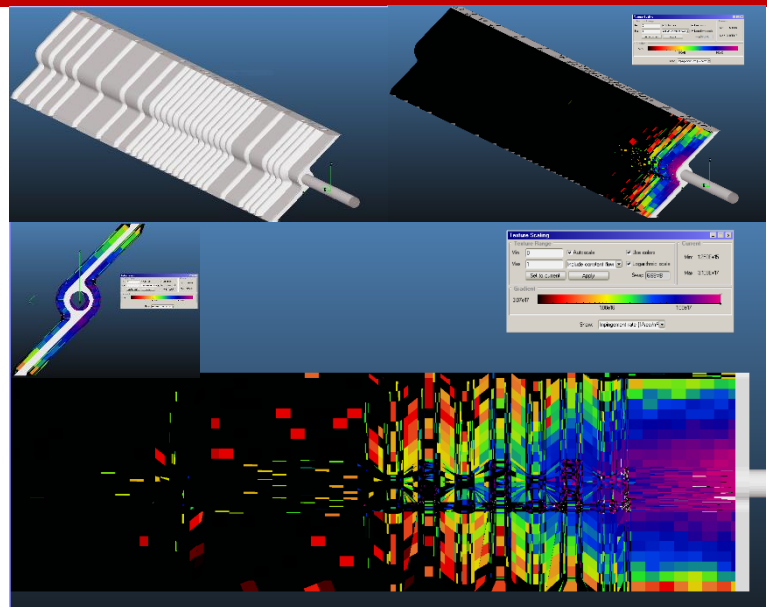
- H_2 injection (1.5e-4 mbar·l/s); s0 = 0.0179, s=0 for H_{conc}=3.9%.
- Real geometry (3.5 mm gap b/w Cell and RF foil).
- Simulated 96h of continuous gas injection.

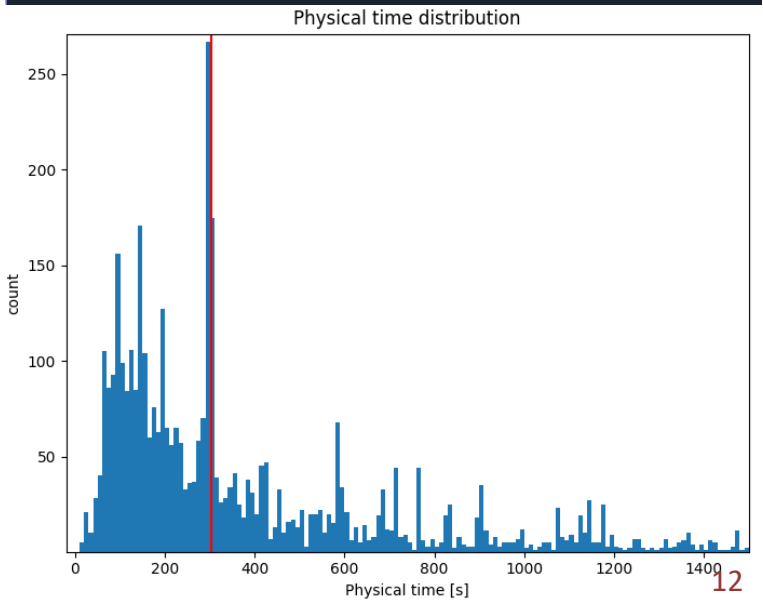
RF foil geometry highly segmented: >90000 facets and 85% of facets (~12% of total area) have an area lower than 1 mm^2.

- Slow simulation, high fluctuations.

IDEA! Using median of physical time distribution instead of minimum.

- Faster simulation (1 day → 30 minutes) BUT some facets may **oversaturate**.



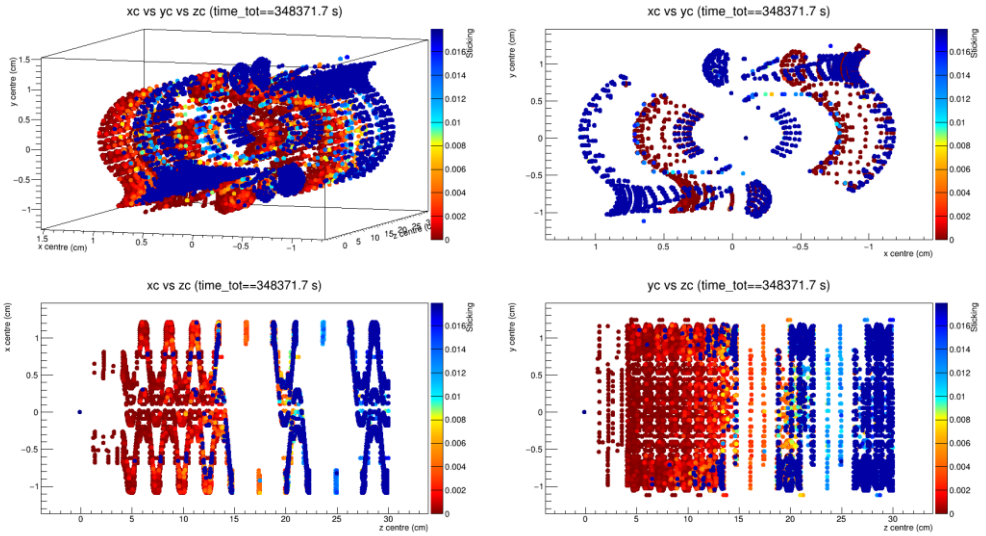


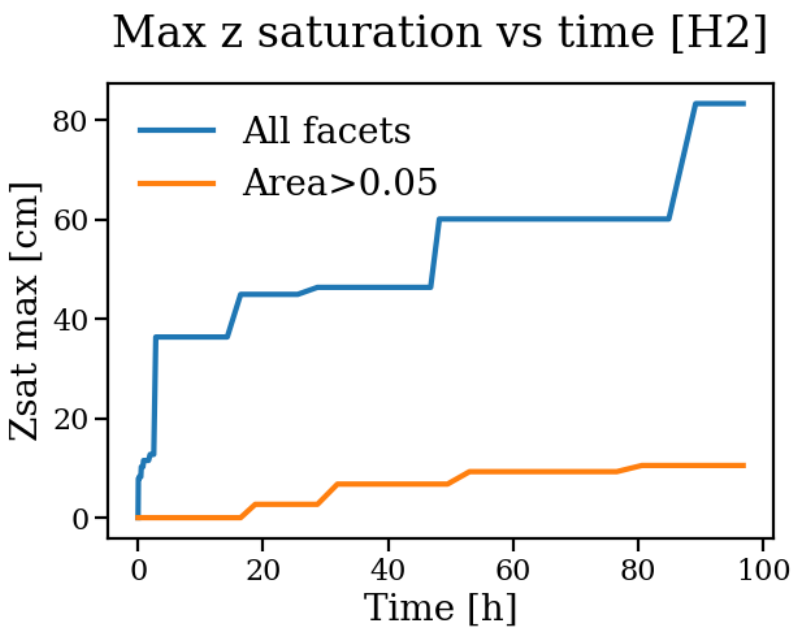
(Preliminary) results: Saturation

Max Z saturation vs time:

Almost complete saturation after 96h but from heat maps saturation is predominant up to 15cm.

- It's an "artifact" due to micro-facets.





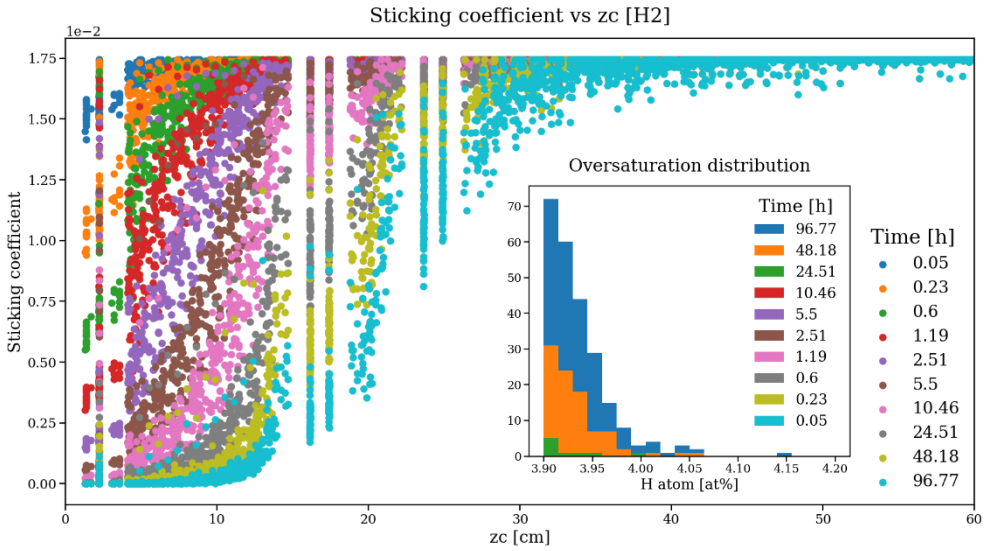
Momentary solution → Cut on the facets' area:

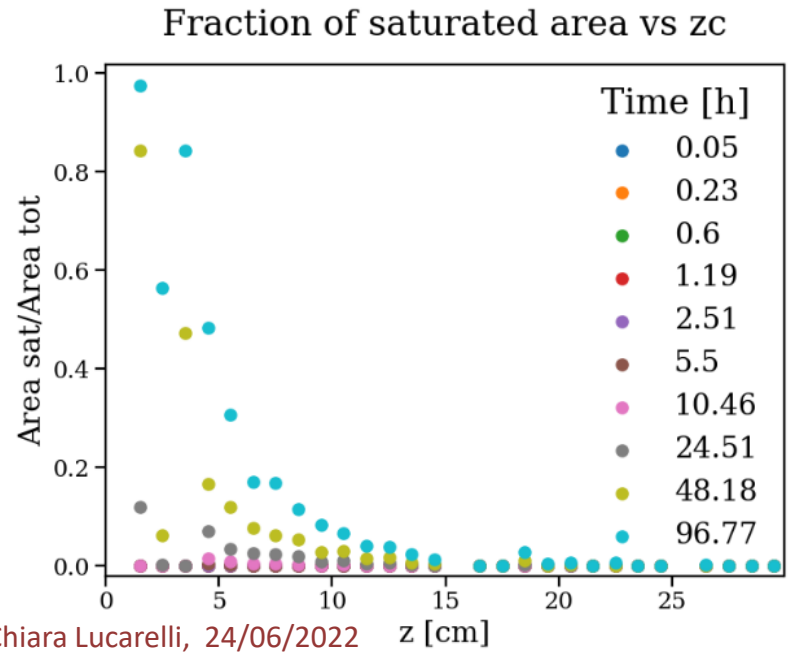
- 👎 Considering only ~ 10% of facets
- 👎 Disregarding cluster of saturated micro-facets
- 👍 Facets considered have better statistical accuracy (higher # hits and # absorbed)
- 👍 Max Z vs t reproduces better heat maps.

(Preliminary) results: Saturation

Sticking coefficient vs z, filtered (A>5 mm^2):

- After 96h, around 15cm of NEG saturated.
- Oversaturation is limited (4% atomic concentration corresponds to ~ 2% of oversaturation).
- 54% of facets saturate with more than 100 absorbed virtual particles.





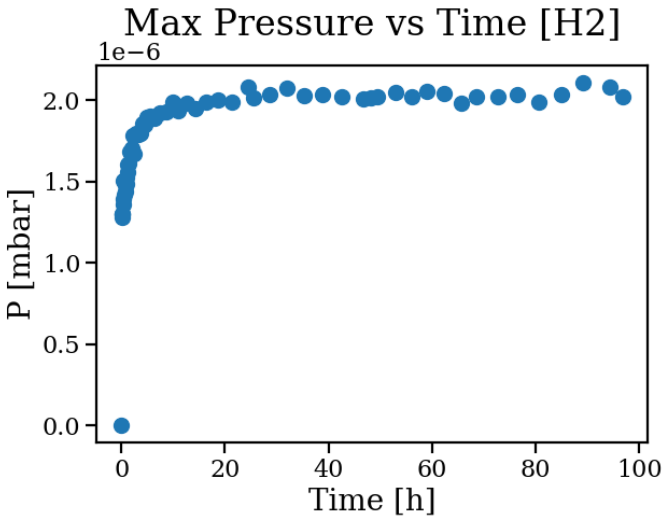
Checks (on complete dataset):

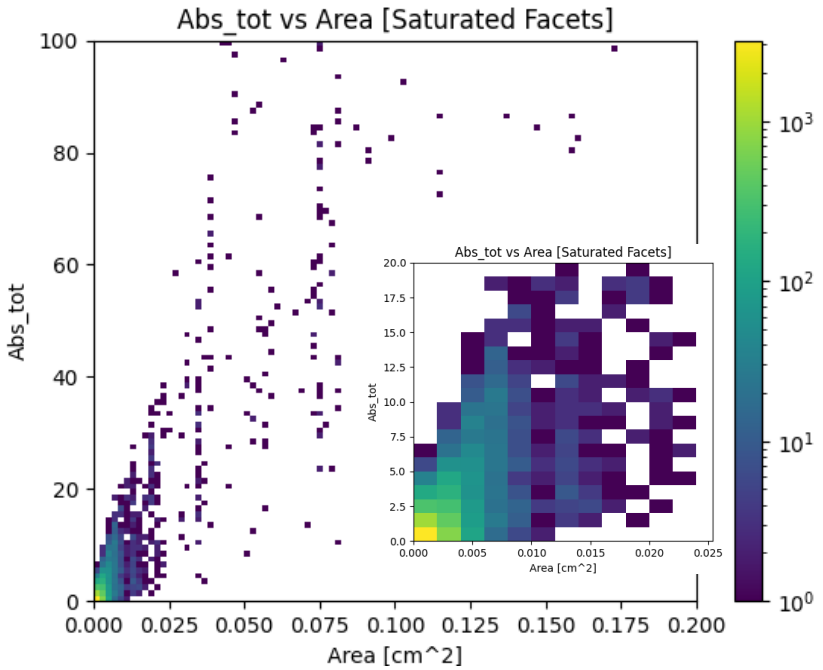
- Fraction of saturated area: after 15 cm, less than 10% of NEG is saturated for each z-bin.
- Fraction of discarded facets: discarded micro-facets represent 25% of total saturated area (but they represent the 97% of the number of saturated facets).

(Preliminary) results: Checks

Max pressure vs time:

- Pressure always lower than 2e-6 mbar (limit of validity for the model).





Area vs total absorbed virtual particle (Saturated facets):

- 91% of micro-facets saturate with less than 5 absorbed virtual particles

  → **Statistical accuracy is still very limited!**

Conclusion and Next Steps

Done:

- Defined sticking coefficient saturation model for CO and N_2 as reference for getterable gases; measured sticking coefficient saturation for H_2 at constant pressure (thanks Pedro!)
- Python script for iterative molflow simulation stable and validated against independent analysis.
  - Easy scalability and inclusion of new models/parameters/controls.
- First preliminary results with H_2 on new cell+RF foil geometry (tot. phys. time: 96h):
  - "Median" method: speed up simulation at cost of possible oversaturation.
  - After 96h, around 15 cm of RF foil is saturated.
  - "Micro-facets" limit statistical accuracy.

Next steps:

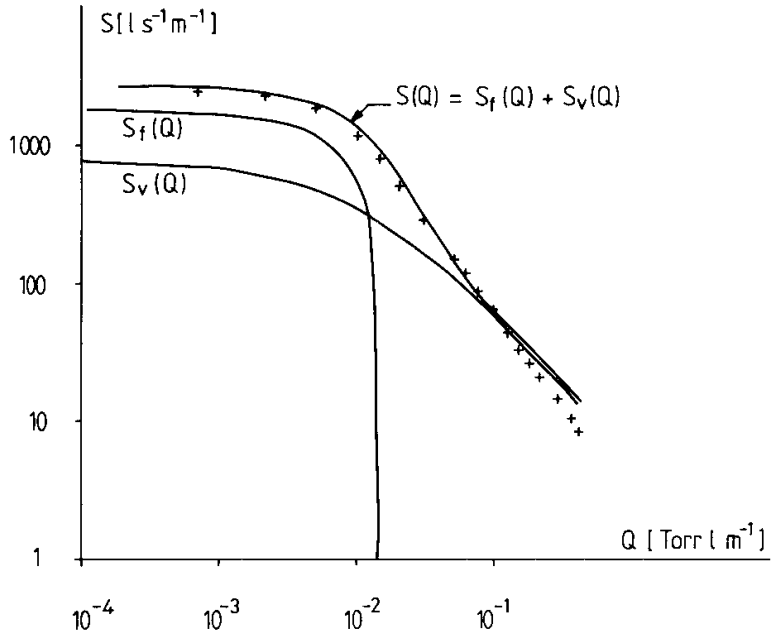
- Validation of simulation on easy geometries against experimental results.
- Simplification of geometry in order to solve the "micro-facets" problem: lower simulation time, reliable and statistical accurate results.

Backup

Sticking model: theory of sticking coefficient saturation

NEG surface can be more or less porous according to its production. General model:

- NEG modelled as a flat outer surface with multitude of circular holes; each hole represents the end aperture of a straight cylindrical void, the inner surface of which has the same sticking coefficient as the flat outer surface.
- Initially, the molecules are adsorbed either on the flat surface (s_f) or inside the voids (s_v); when the outer surface saturates, adsorption persists in the voids. Approaching complete saturation, the decrease of the sticking coefficient is enhanced (not modelled).



$$s = s_f + s_v$$

$$s_f \sim s_{0,f}\left(1 - x/x_{0,f}\right)^p \ , \ s_v \sim s_{0,v}/(1+x)$$

Sticking model: getterable gases

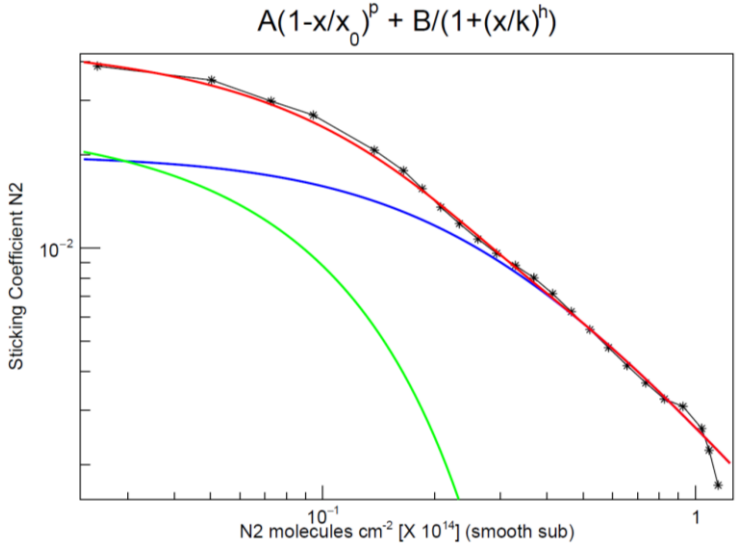
Starting from an empirical model (J.Vac.Sci.Technol.A.6.2528) and experimental data for CO and N_2 (j.tsf.2005.12.218, cern-thesis-2002-042), parameters are determined through data fit:

Model function:

$$\begin{cases} s = A\left(1 - x/x_0\right)^p + B/\left(1 + (x/k)^h\right) & x < x_0 \\ s = B/\left(1 + (x/k)^h\right) & x > x_0 \end{cases}$$

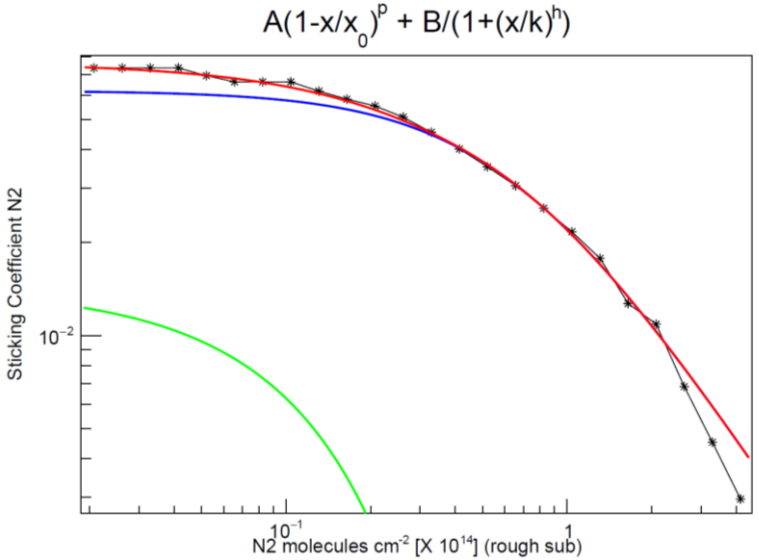
- Values for high coverage ($x>x_0$) are fitted to determine B, k, h.
- The whole dataset is fitted with B, k, h fixed.
- Value for p fixed from theory.
- Final fit values renormalized to intrinsic sticking coefficient (s_{max}) and surface capacity (x_{max}) measured on LHC-like NEG samples.

$$\begin{cases} A + B = s_{max} \Rightarrow scale = s_{max}/\left(A + B\right) \\ s_{sat} = B/\left(1 + (x_{max}/k)^h\right) \end{cases}$$



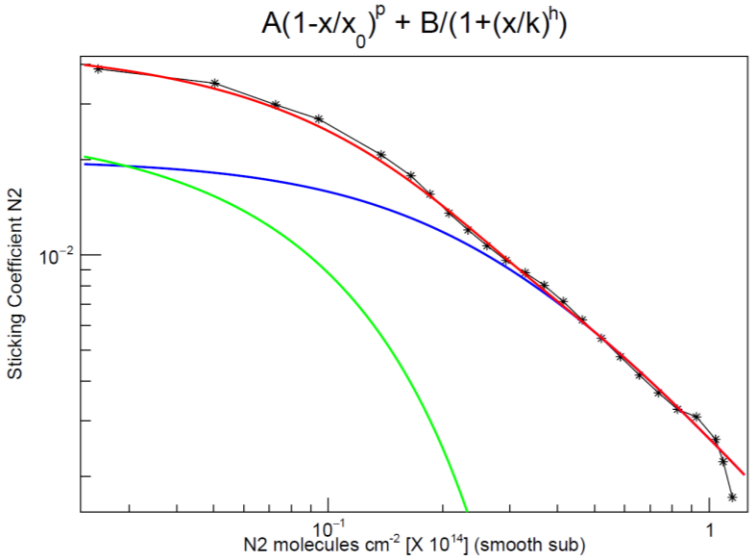
→ It should be repeated for each gas species + NEG sub, with experimental data.

Sticking model: N2 smooth and rough

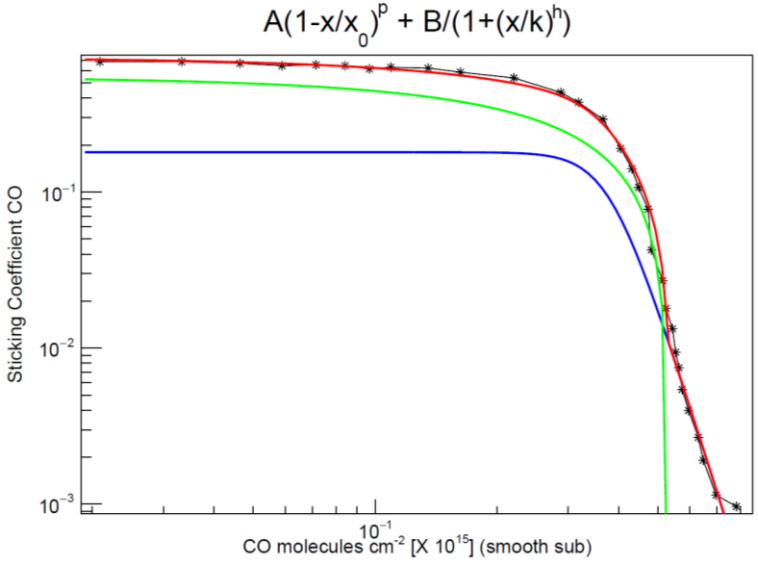


A	0.0146
x_0	1 x10^{14}
p	7
B	0.062
k	0.65 x10^{14}
h	1.39

A	0.0258
x_0	0.7 x10^{14}
p	7
B	0.020
k	0.26 x10^{14}
h	1.4



Sticking model: CO smooth



A	0.547
x_0	0.53 x10^{15}
p	1
B	0.18
k	0.38 x10^{15}
h	8.2

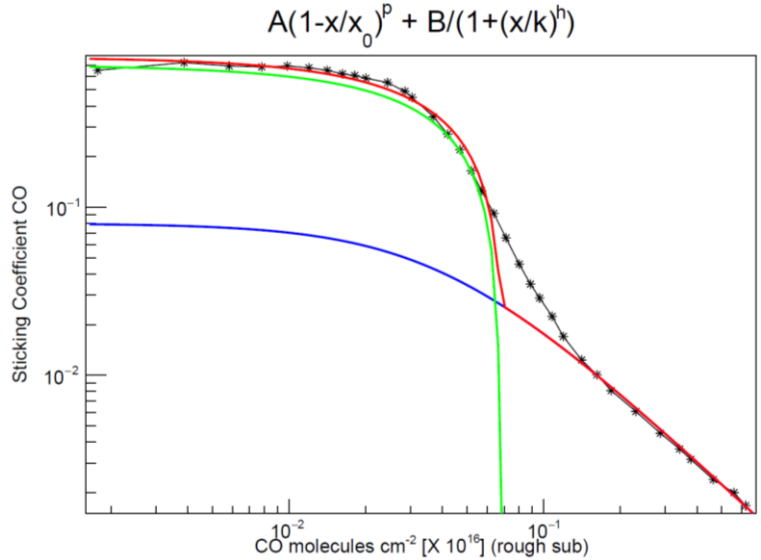
Sticking model: CO rough

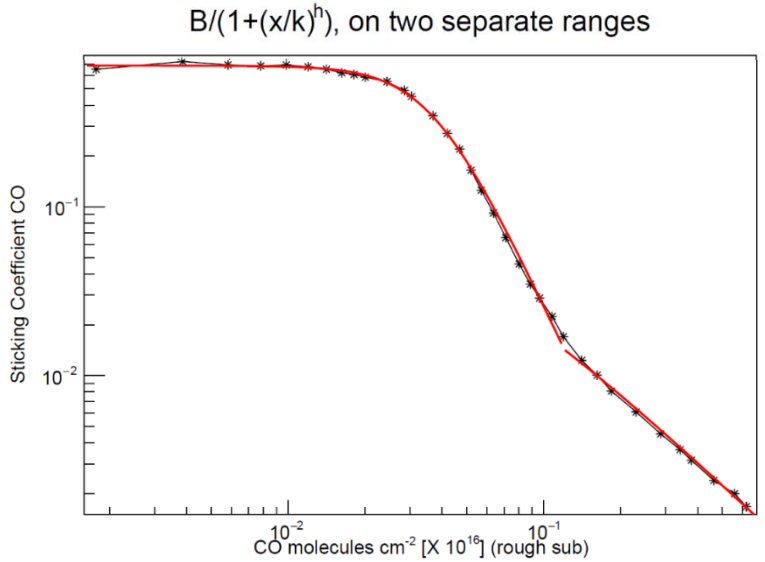
The model cannot reproduce CO on the rough substrate

→ The data are reproduced using $\begin{cases} s = B_1 / \left(1 + (x/k_1)^{h_1}\right) & x < 1.2 \times 10^{15} \\ s = B_2 / \left(1 + (x/k_2)^{h_2}\right) & x > 1.2 \times 10^{15} \end{cases}$

A	0.703	B	0.080
x_0	0.0677 x10^{16}	k	0.041 x10^{16}
p	1	h	1.42

B_1	0.687	B_2	0.080
k_1	0.0368 x10^{16}	k_2	0.041 x10^{16}
h_1	3.25	h_2	1.42





Python script: input and output

Input:

- xml/zip geometry file (from the Molflow GUI).
  → Must already include outgassing.
- Starting sticking coefficient.

Output:

- xml/zip MolflowCLI output file for each step (option: overwrite input file).
- xml summary file with relevant data:
  - simulation parameters: gas mass, total outgassing, input and output file.
  - *facet parameters*: id, temperature, area, centre coordinates.
  - *iteration data*: id, CPU time step, scale factor, total time, pressure, density, # hits (for iteration, and total), absorption rate, concentration, sticking.
    → *NB: starting conditions memorized with iteration id = -1*

Python script: parameters and controls

Parameters:

- ***CPU Time steps:*** It's possible to define variable CPU time steps that follows a predefined sequence or update CPU time steps so that the scale factor remains (almost) constant.
- ***Physical time (and total time):*** the physical time of each step is chosen in order to move along the sticking coefficient curve evenly for every facet (i.e. no extreme coefficient jump in one step) → Minimum time (for all facets) that produces a decrease in sticking coefficient lower than a fixed value (i.e. decrease of 10% of the sticking coeff.)
- ***Facets*** to be updated (indexes, intervals, selection groups).
- ***Sticking evolution model:*** N2, CO and H2.
- ***Stop condition:*** condition that interrupts the simulations loop. Currently available: maximum simulation time, maximum saturation propagation along z, maximum number of iterations and saturation in any of the facets.
- ***Starting point:*** the simulation can start from any intermediated simulation step

Python script: mesh and textures

Each facet can be subdivided into cells of a give size (***texture***) and physical quantities (density, pressure, impingement rate) can be memorized for each individual cell. Facets physical properties (T, opacity, sticking, outgassing) can be **ONLY** associated to the whole facet.

The simulation allows to consider the texture while updating the sticking coefficient:

- ***Texture***: if the facet has a texture, the sticking coefficient can be calculated for each cell in the mesh. The facet sticking coefficient is the average of the cells sticking coefficients. DEPRECATED! Since the sticking coefficient can be associated only to the whole facet, the texture method tends to produce oversaturation of the facet!

Validation strategy

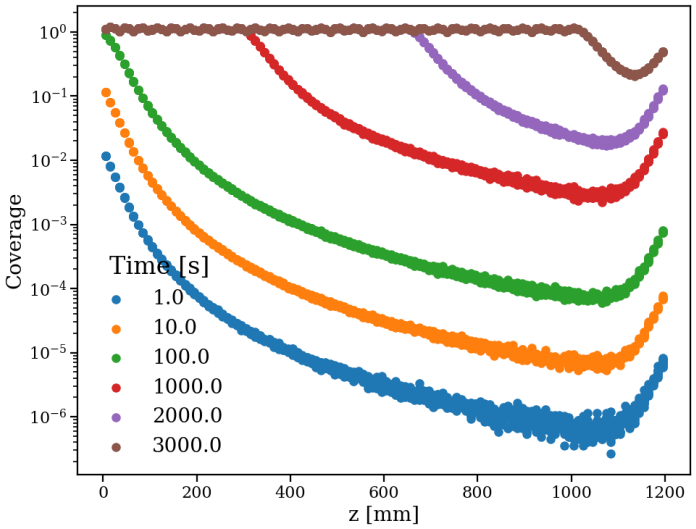
Validation of the script on a simple pipe in order to reproduce Yasunori Tanimoto results (presentation).

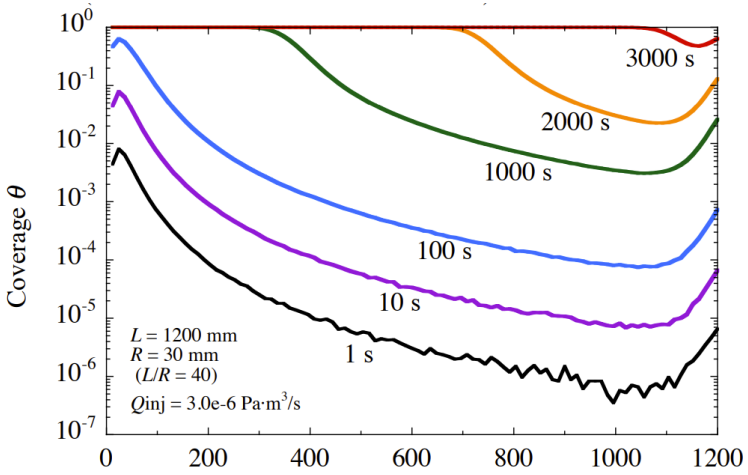
- Constant outgassing from one extreme; constant pumping speed from opposite extreme (7 l/s).
  - Starting sticking coefficient: 1.
  - Simple test model: s = 1 – coverage (=x/x_{max}).
  - Stop condition: complete saturation of the pipe.

Results:

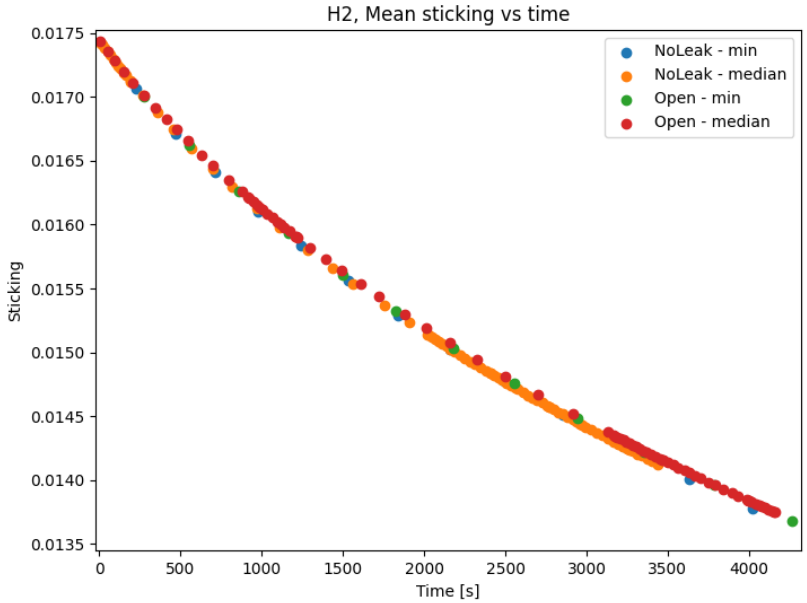
- **Good reproduction of results with independent simulation strategy.**

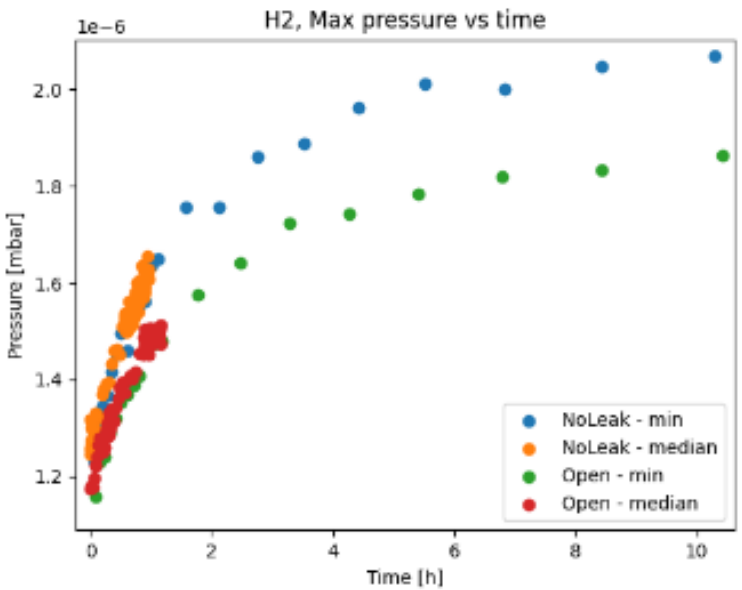
  → NEXT STEP: validation against experimental results





Check: median vs min physical time, real vs no leaks geometry





Comparisons (only for the central part, z < 20 cm):

- Mean sticking coefficient vs time.
- Maximum pressure vs time.

Results:

- At fixed geometry, **no differences between minimum and median methods**; NoLeak geometry decreases faster (as expected).

Analysis strategy

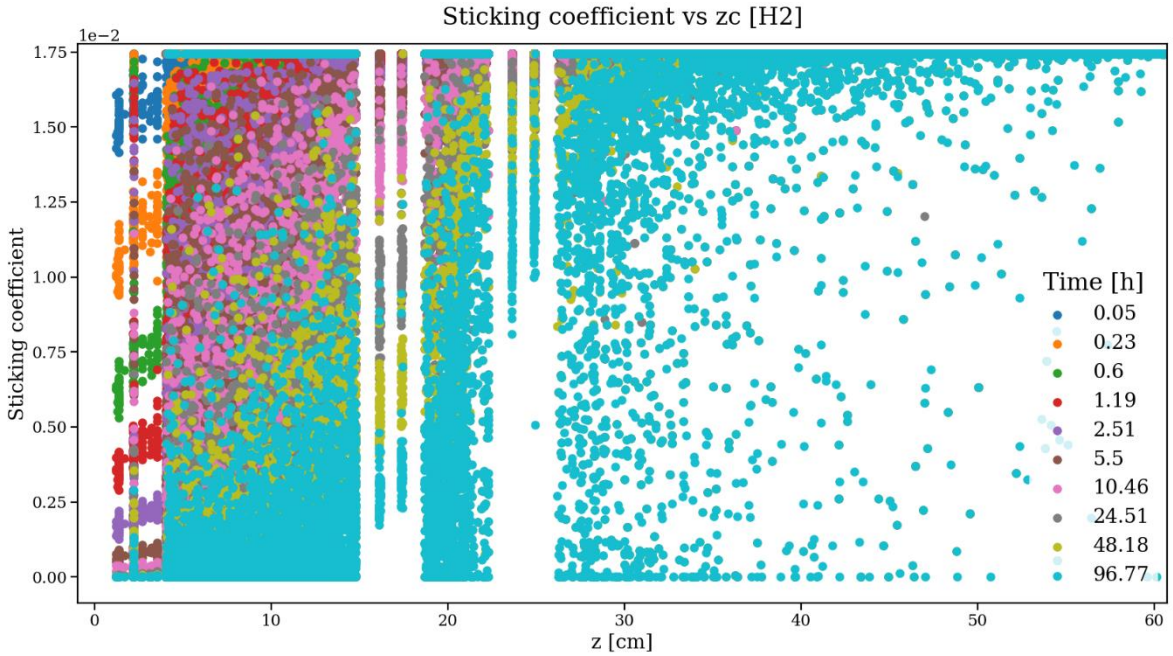
Saturation studies:

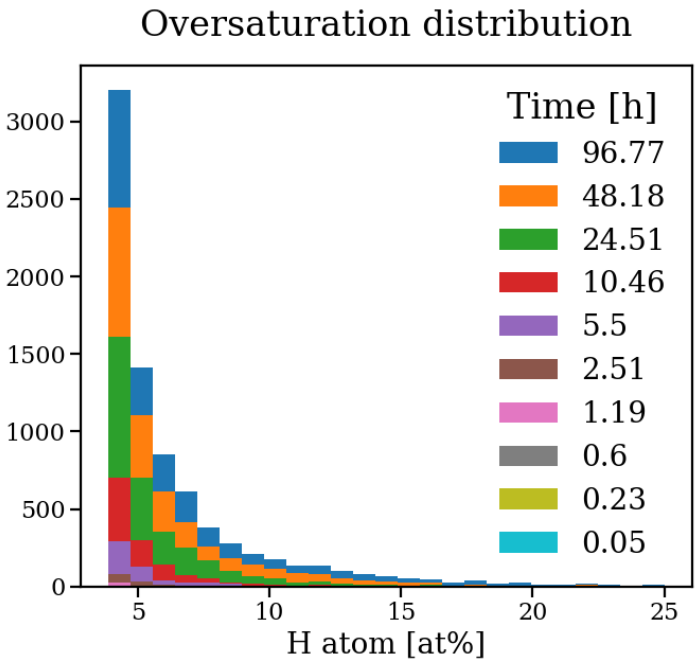
- Heat maps (3D visualization): each facet is represented as a point corresponding to its centre, the sticking value is represented through the colour intensity.
- Maximum z sat vs t
- Sticking coefficient vs z
- H concentration vs t

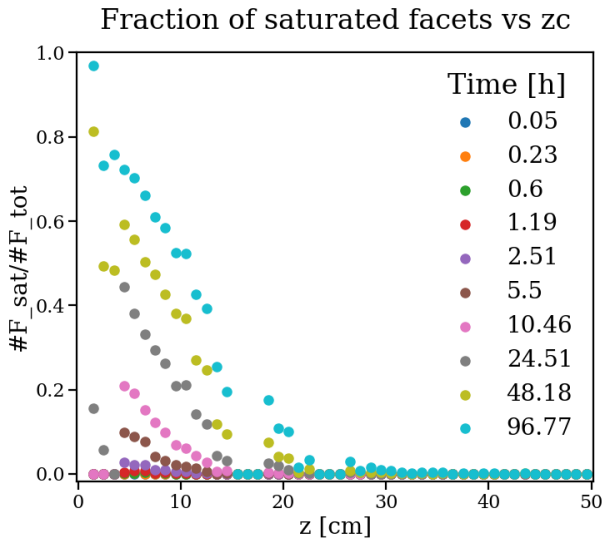
«Quality» checks:

- Fraction of saturated facets → Impact of “micro-facets”
- Area vs Abs tot at saturation → Statistical accuracy of saturation
- Max Pressure vs t → Validity of model
- Hits tot/Abs tot vs z and t → Statistical accuracy

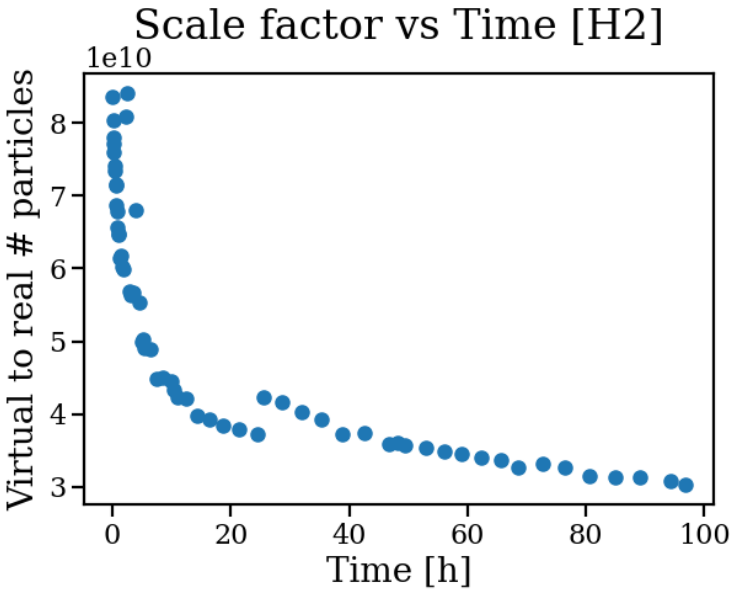
(Preliminary) results: Saturation with no filter on area

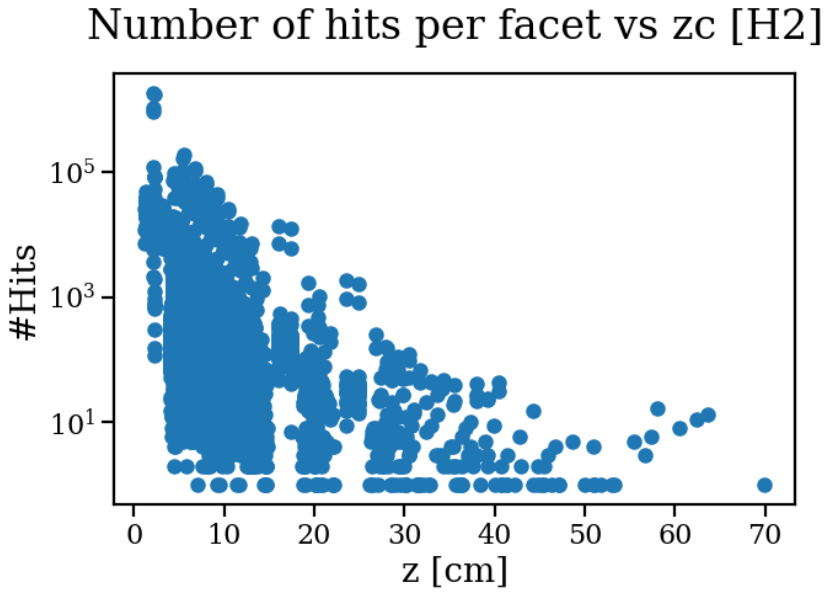


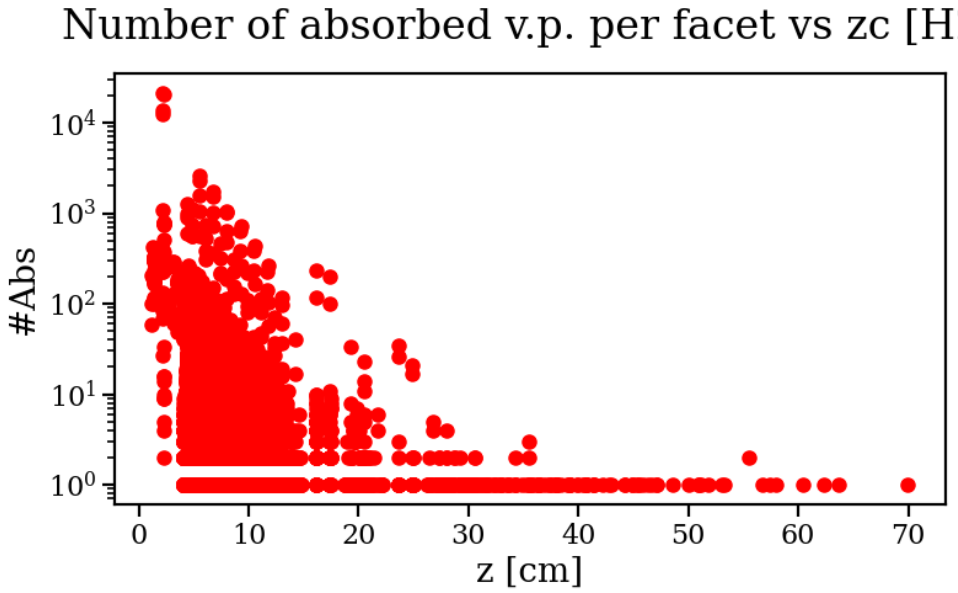




(Preliminary) results: scale factor







(Preliminary) results: Density profile

Density profile reaches "equilibrium" after around 20 h.