Numerical Simulation of Flow in SuperPak Family Packings

Jakub Smutek[1], Martin Isoz[1,2]

[1]Department of Mathematics, Faculty of Chemical Engineering, University of Chemistry and Technology, Studentska 5, 166 28 Prague, Czech Republic
[2]Institute of Thermomechanics of the Czech Academy of Sciences, Dolejskova 5, 182 00 Prague, Czech Republic

UNIVERSITY OF CHEMISTRY AND TECHNOLOGY PRAGUE

Introduction

The distillation unit operation is the most energy-consuming process in chemical industry. It is responsible for approximately 50 % of the total energy consumption of industrial separation processes in the US [1]. Commonly, the distillation is performed in the columns filled with a structured packing. The packing highly increases intricacy of the flow in the column. Consequently, it is nearly impossible to predict hydrodynamic properties of newly designed packings and all new packing designs have to be verified experimentally.

The present work is a part of a larger project [2, 3]. The aim of the project is to establish a CFD methodology, usable for evaluation of structured packing performance, to optimize existing packings, and to develop new packing designs.

In the present work, we concentrate on modeling the gas flow in a SuperPak family of structured packings. Our approach to model the gas flow in these packings is based on creating a geometry representation that resembles the real packing as much as possible, see Fig. 1.

Figure 1 : Comparison of laboratory scale SuperPak 250.Y packing and our representation of the packing

SuperPak type packing geometry generation algorithm

Require: Packing geometry parameters: $N^x_{arc}, N^y_{arc}, w_{arc}, l_{arc}, h_{arc}, m_{arc}, \theta_{ch}, N_{sh}, H_{pack}, D_{pack}$

Make a rectangle Re centered at origin with sides $(N^x_{arc}+1)\, w_{arc} \times (N^y_{arc}+1)\,(l_{arc}+m_{arc})$;
for $j = 0$ **to** $N^y_{arc} - 1$ **do**
  Cut a rectangular hole of dimensions $k\, w_{arc} \times l_{arc}$ into Re centered at $S_h = (-(l_{arc}+m_{arc})\frac{N^y_{arc}}{2} + j\,(l_{arc}+m_{arc}), 0, 0)$;
  for $i = 0$ **to** $N^x_{arc} - 1$ **do**
    Create an arc A_{ij} of dimensions l_{arc} and h_{arc} and with center at $S_{arc} = (-(l_{arc}+m_{arc})\frac{N^y_{arc}}{2} + j\,(l_{arc}+m_{arc}), -\frac{w_{arc}}{2}N^x_{arc} + i\,w_{arc}, (-1)^i\,(R_{arc}-h_{arc}))$;
  end for
end for
Join all arcs A_{ij} and rectangle Re into one sheet $S \leftarrow (\bigcup_{ij} A_{ij}) \bigcup Re$;
Make a rectangular cuboid Rc with center at origin and of dimensions $D_{pack} \times H_{pack} \times 3h_{arc}$;
Rotate S by an angle θ_{ch} around z axis;
Create first sheet of a packing S_0 by intersecting sheet S with Rc. $S_0 \leftarrow S \cap Rc$;
for $i = 1$ **to** N_{sh} **do**
  Create S_i by copying S_0;
  Translate S_i by $-\frac{(N_{sh}-1)}{2}2h_{arc} + 2h_{arc}\,i$ on the z axis;
  if $i \mod 2 = 1$ **then**
    Rotate S_i by π around the y axis;
  end if
  Create a rectangular cuboid Rc_i, centered at origin of dimensions $\sqrt{\frac{D^2_{pack}}{4} - (-(N_{sh}-1)\,h_{arc} + 2h_{arc}\,i)^2} \times H_{pack} \times D_{pack}$;
  Cut S_i by making an intersection of S_i with Rc_i: $S_i \leftarrow S_i \cap Rc_i$;
end for
Create packing P by joining all sheets S_i together: $P \leftarrow (\bigcup_{i=1}^{N_{sh}} S_i)$;
Extrude packing P by th_{arc} in the z axis direction;
return Geometry representation of the packed bed suitable for the snappyHexMesh utility

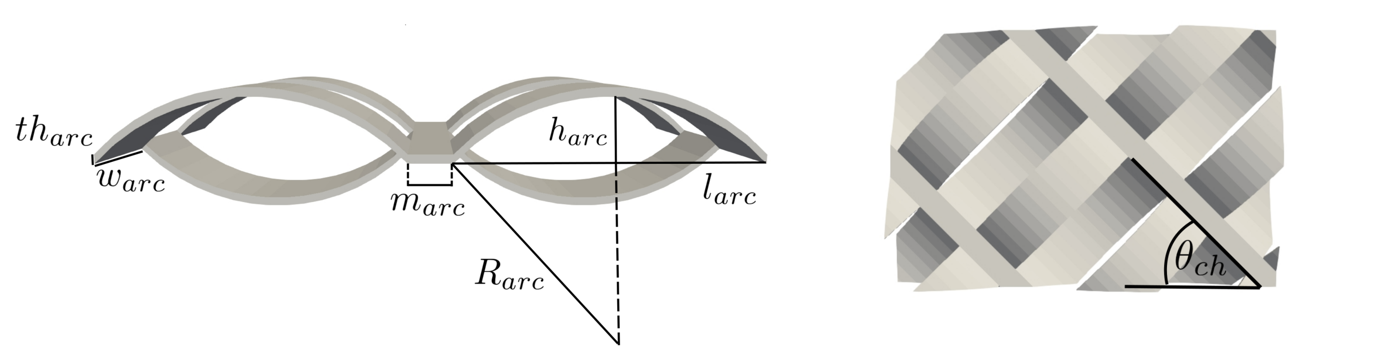


Figure 2 : Depiction of parameters required to define the SuperPak type packing geometry

Acknowledgments

This work was supported by The Ministry of Education, Youth and Sports from the Large Infrastructures for Research, Experimental Development and Innovations project "IT4Innovations National Supercomputing Center– LM2015070". The work of M. Isoz was supported by the Centre of Excellence for nonlinear dynamic behaviour of advanced materials in engineering CZ.02.1.01/0.0/0.0/15_003/0000493 (Excellent Research Teams) in the framework of Operational Programme Research, Development and Education. Finally, the authors would like to express theirs deepest thanks to the Mass Transfer Laboratory of UCT Prague for providing experimental data for the model validation as well as for granting an access to their experimental equipment and for many helpful consults on the studied topic.

Simulation set up

To estimate dry pressure losses, we need to solve a set of isothermal, turbulent, steady-state Navier-Stokes equations for an incompressible Newtonian fluid [5],

$$\nabla \cdot (\overline{\mathbf{U}} \otimes \overline{\mathbf{U}}) - \nabla \cdot (\overline{\mathbb{T}} + \mathbb{T}') = -\nabla \overline{p}$$
$$\nabla \cdot \overline{\mathbf{U}} = 0\,,$$

where $\mathbb{T}'$ is the Reynolds stress tensor. In this work we used the k-ω SST model [4] for the closure of the problem.

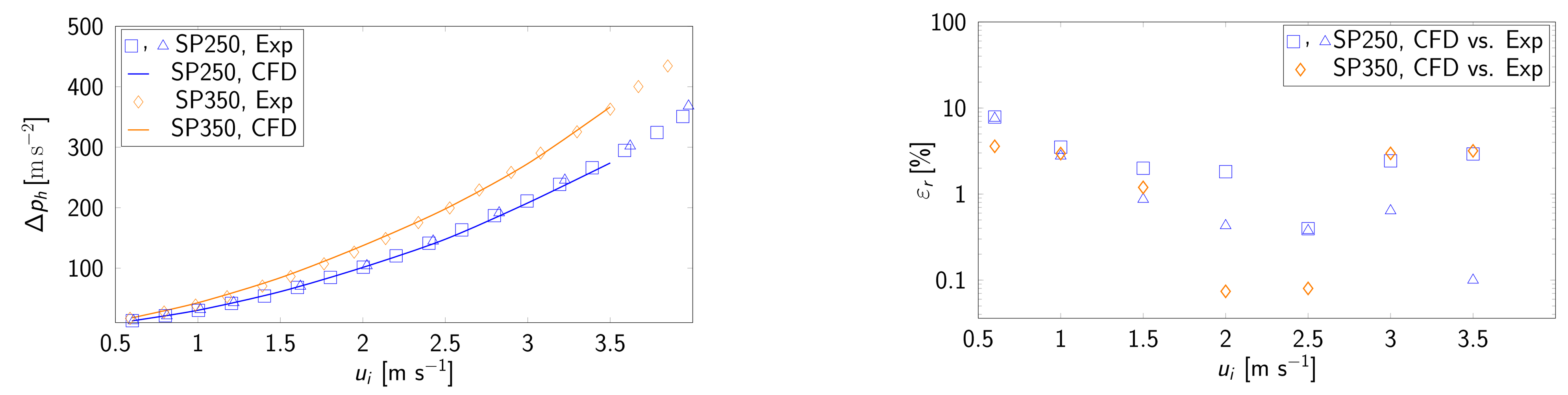
The flow governing equations together with the equations for the turbulence variables k and ω were solved via the simpleFoam solver from the OpenFOAM toolbox . The used finite volume (FV) mesh was created using the snappyHexMesh software, which is available in the OpenFOAM installation. The needed representation of a packing geometry in STL format was prepared via SuperPak type packing geometry generation algorithm implemented in the Blender software. The applied boundary conditions are listed in Tab. 1.

Boundary	Condition
Inlet	$\mathbf{U} = (0, u_i, 0)^{\mathrm{T}},\ \mathbf{S}_f \cdot \nabla p = 0,\ k = k_0,\ \omega = \omega_0$
Outlet	$\mathbf{S}_f \cdot \nabla \mathbf{U} = (0,0,0)^{\mathrm{T}}$ if $\mathbf{\Phi} > 0$, $\mathbf{U} = (0,0,0)^{\mathrm{T}}$ else, $\mathbf{\Phi} = \mathbf{S}_f \cdot \mathbf{U}$ $p = 0,\ \mathbf{S}_f \cdot \nabla k = 0,\ \mathbf{S}_f \cdot \nabla \omega = 0$
Wals	$\mathbf{U} = (0,0,0)^{\mathrm{T}},\ \mathbf{S}_f \cdot \nabla p = 0,\ \mathbf{S}_f \cdot \nabla k = 0,\ \mathbf{S}_f \cdot \nabla \omega = 0$

Table 1 : Applied boundary conditions. The column height and radius are denoted as H_{col} and r_{col}, respectively. The symbol $\mathbf{S}_f$ marks an outer normal vector to the boundary.

$$k_0 = \frac{3}{2}(u_f l_0)^2 \qquad l_0 = 0.0853\mathrm{Re}^{-0.0727}$$
$$\mathrm{Re} = \frac{u_f l_{arc}}{\nu} \qquad u_f = \frac{u_i}{\varepsilon_{\mathrm{pk}} \sin \theta_{\mathrm{ch}}}$$
$$\omega_0 = \frac{\epsilon_0}{k_0} \qquad \epsilon_0 = \frac{0.1643 k_0^{3/2}}{0.07 l_{arc}}$$

Model validation



(a) Dependence of pressure drop on inlet velocity

(b) Relative errors of calculated pressure drops

Figure 3 : Comparison of measured and computed dry pressure loss.

Results

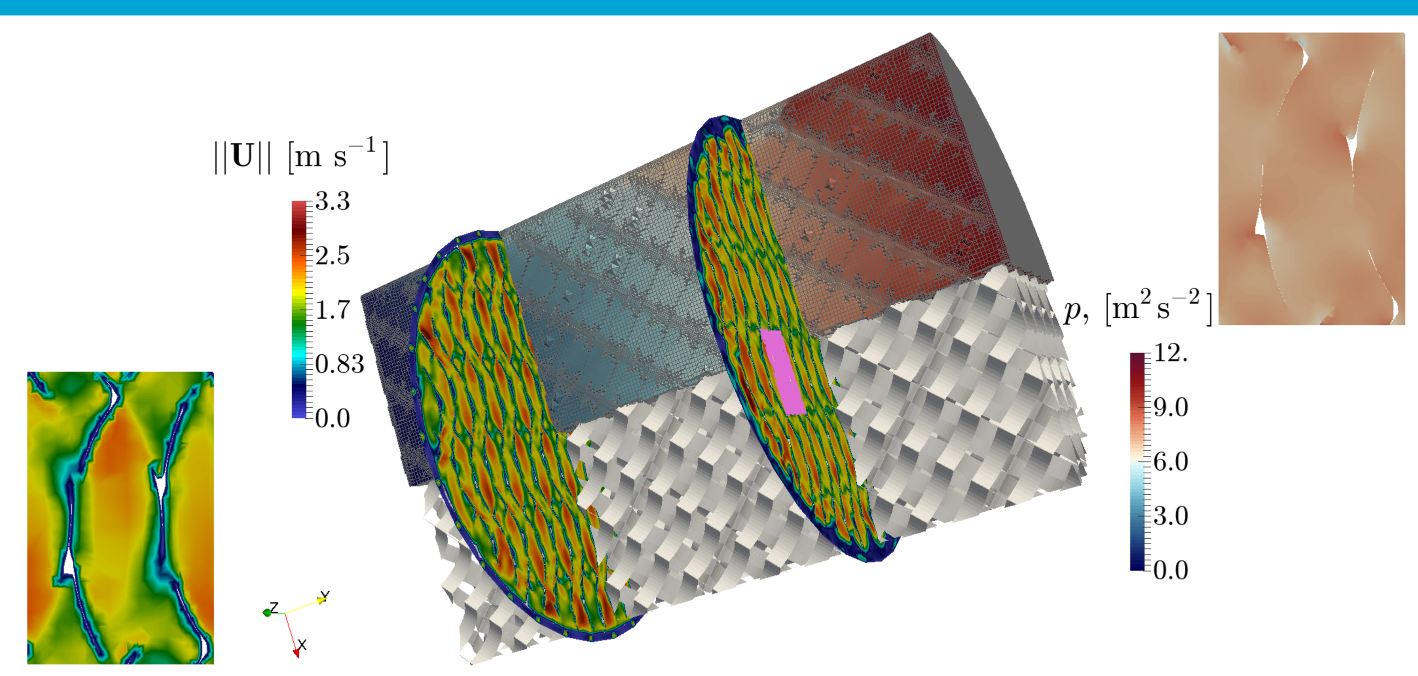
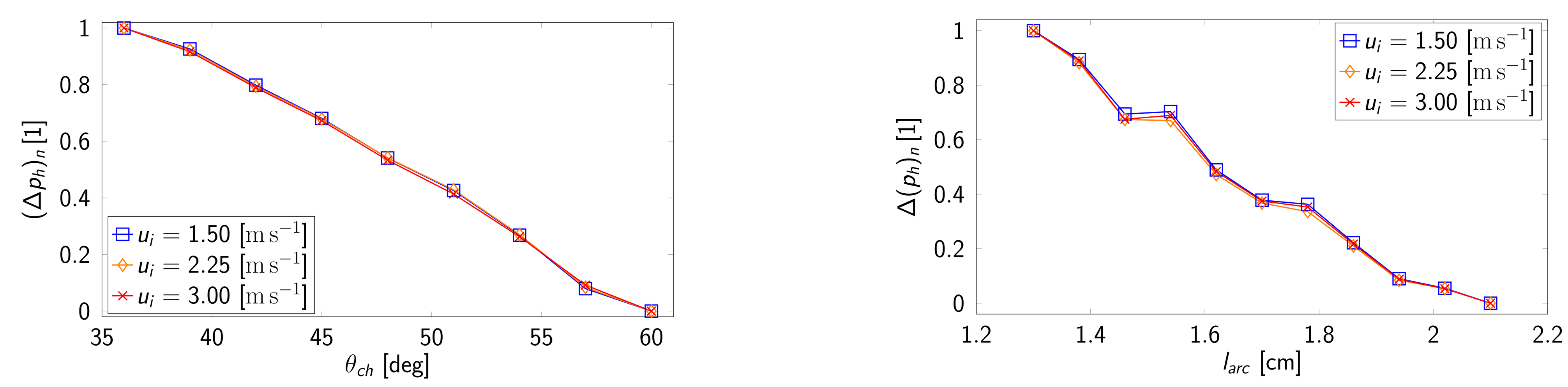


Figure 4 : Velocity and pressure fields in the SuperPak 250.Y packing. Details of velocity and pressure in the highlighted pink area are shown in the bottom left and upper right part of the image, respectively. Inlet velocity was $u_i = 1.5\,\mathrm{m\,s^{-1}}$.

$$\Delta p_h := \frac{p_{\mathrm{above}} - p_{\mathrm{below}}}{N_{pack} H_{pack}} \qquad (\Delta p_h)^i_n := \frac{\Delta p^i_h - \min_{(i)} \Delta p_h}{\max_{(i)} \Delta p_h - \min_{(i)} \Delta p_h}$$



(a) $\Delta(p_h)_n$ as a function of θ_{ch}

(b) $\Delta(p_h)_n$ as a function of l_{arc}

Figure 5 : Parametric study of SuperPak 250.Y packing.

References

[1] Materials for Separation Technologies: Energy and Emission Reduction Opportunities.
Technical report, U.S. Departmesnt of Energy's Office of Energy Efficiency and Renewable Energy, 2005.
https://www1.eere.energy.gov/manufacturing/industries_technologies/imf/pdfs/separationsreport.pdf.

[2] M. Isoz.
CFD study of gas flow through structured separation columns packings Mellapak 250.X and Mellapak 250.Y.
In D. Simurda and T. Bodnar, editors, *Proceedings of the conference TPFM'17, Topical Problems of Fluid Mechanics*, pages 171–184. IT CAS, 2017.

[3] M. Isoz.
Effects of surface textures on gravity driven liquid flow on inclined plate.
In J. M. Nobrega and H. Jasak, editors, *11th OpenFOAM Workshop*. Springer Nature, Berlin, Germany, 2018.
In print.

[4] F. R. Menter.
Two-equation eddy-viscosity turbulence models for engineering applications.
AIAA, 32(8):1598–1605, 1994.

[5] F. Moukalled, M. Darwish, and L. Mangani.
The finite volume method in computational fluid dynamics: an advanced introduction with OpenFOAM and Matlab.
Springer-Verlag, Berlin, Germany, 1st edition, 2016.