A cell expansion framework for property-based automatic compartmentalization of Computational Fluid Dynamics (CFD) models

Lide Gonzalez-Sala,a Gorka Sánchez Larraona,a Paloma Grau,a Borja Hernándeza\*

a TECNUN Engineering School, University of Navarra, Manuel Lardizabal Ibibildea 13, San Sebastian, 20018, Spain.

bhernandezb@unav.es

Abstract

This work presents a framework for compartment model generation through the cell aggregation method. The framework aggregates the cells with a similar value of a target property directly from the mesh of the CFD model. This aggregation is performed directly from the results in the mesh of the CFD model, which has two main advantages against other compartmentalization frameworks. First, it identifies irregular 3D zones with similar properties. Second, it does not require the user to generate a new mesh on top of the results, which significantly improves automatizing the procedure and saves time for the user. Two case studies have been conducted to validate the framework: a Venturi that determines the cuts according to the axial velocity and an elbow that mixes two pipes with different temperatures and determines the cut according to temperature and velocity.

**Keywords**: Compartment models, CFD, scale-up, data management.

* 1. Introduction

Computational Fluid Dynamic (CFD) models have been widely used for modeling unit operations in chemical pharmaceutical or environmental industries. Despite their high computational cost, CFD provides internal fluxes and a detailed distribution of properties within the unit. One alternative to reduce its computational cost is the use of compartment models (CM) (Jourdan et al., 2018). CM can be classified in two approaches: Systemic compartment models based on the equivalent reactor network theory, and Zonal Models (ZM), which introduce a topological description within the unit. The first approach has been mainly employed for tracing and determining residence time distribution (RTD) into a unit (Danckwerts, P.V. 1953). The second approach, introduced by Bermingham et al. (1998), is based on the extraction of information on the fluxes within the unit to obtain a model based on a network of reactors that are equivalent to the finite volumes of the CFD model (Jourdan et al., 2018). These fluxes generated from the CFD model can be implemented in the reactor network following a data-driven approach or by means of surrogates. The flexibility of surrogate models has made this second approach the most interesting. Multiple surrogate modeling techniques (e.g., neural networks (Queipo et al. 2005), kriging (Boukouvala et al. 2013), dimensionless groups correlations (Hernández et al. 2022)) have been applied to modeling the fluxes in ZMs.

Apart from focusing on generating the surrogate models for the fluxes, research in ZM generation has also focused on automatizing the extraction of fluxes. The first work developed by Bezzo et al. (2004) introduced the automatic construction of compartments by aggregating the cells with similar properties. The aggregated cells were generated by the user on top of the results obtained from the CFD model, and its accuracy was limited by the number of cuts re-defined by the user. The first alternative and less automatized approach for identifying the right cuts was proposed by Alvarado et al. (2012). They measured the property through a set of lines and generated cuts at sections exhibiting changes. Due to the extensive size of the resulting zones in the ZM, they integrated the ZM approach with a systemic one. Within each zone, they incorporated an internal set of equivalent reactors. In another alternative to cell aggregation, Delafosse et al. (2014) proposed the layer-by-layer method for generating the network of reactors. Among the three approaches (cell aggregation, combined cell aggregation and systemic, and layer-by-layer), the most extended one is the one proposed by Bezzo et al.. It has been applied to different properties and is commercialized through the Multizonal tool developed by Process Systems Enterprise (Siemens, 2020). Following this approach, most recent works have also provided a higher automatization of compartment models by following a cell aggregation method with more than one property for CFD models (Tajsoleiman et al. 2019) and with particle concentration for Discrete Element Models (DEM) (Bhalode et al. 2021). However, the construction in both cases was still based on generating a mesh on top of the results. The generation of this mesh requires user intervention to define the number of cuts to be performed on top of the results. Furthermore, the results of this new mesh may not be accurate enough if the cell size is too large or when irregular zones exist. To overcome these drawbacks, this work proposes a new automatic framework that identifies the location of the compartments, directly analyzing the results obtained in each of the cells of the CFD simulation.

* 1. Methodology
		1. Description of the Framework

The novel framework does not require generating a mesh on top of the results since all the information is extracted directly from the cells. The framework combines User Defined Functions (UDFs) from Ansys Fluent® and data analysis in Python for extracting the data and determining the optimal aggregation of cells based on a set of target properties and a desired increment. A general description of the framework is given in Figure 1, and more details with two case studies are given in the following paragraphs. The framework consists of four steps: First, all the information contained in the CFD model is extracted through UDFs. Second, the properties are set, and their difference between adjacent cells is computed. Third, the zones according to a defined maximum allowed difference are defined. Fourth, the fluxes are extracted between the zones.

* + - 1. Extraction from CFD.

The first step after running the CFD simulation consists of extracting information from all the cells. The extraction of information is divided into two steps. The first one extracts all the target properties of every cell with its identification number (ID) and location, generating an ASCII file named “*Properties-File*”. The extraction of all these properties is performed from *Export → Solution Data* in Ansys Fluent ® and selecting the values in the Cell Center for all the properties. The second extraction, “*Fluxes-File*”, employs a UDF that tracks all the cells extracting the ID and the mass flow rate through each face. The mass flow rate is calculated using the function *F\_FLUX*, which provides the value calculated internally in Fluent®. Therefore, mass conservation is ensured with the same degree of accuracy as in the CFD simulation.

 **Figure 1**. General description of the framework.

* + 1. Data joint and analysis of property differences

The two data files are first merged into a unique *dataframe* in Python. The data extracted in “*Fluxes-File*” contains a column with the ID of the tracked cell (ID-1) and a second column with the IDs of its adjacent cells (ID-Adj). For each ID and its adjacent cell, a search is performed in “*Properties-File*” extracting the values of the target property in the two cells and computing the difference, $∆p$. The difference is stored in a column of the *dataframe,* see Step 3 of Figure 1. After storing the values, the distribution of differences is determined which allows us to set up the value for cutting the zones as a quantile of the distribution or as a specified set given by the user, see the example in step 2 of Figure 1.

* + 1. Cell aggregation algorithm

The user defines a quantile or a threshold difference, *q*, that is employed in determining the regions by cell aggregation. Then, the cell aggregation algorithm begins from a random cell of the column ID-1. The cell is compared with its adjacent ones, and in case the value of the difference is below the threshold, $∆p<q$, the cell aggregates are part of a new zone (Zone=1). An additional column is generated in the *dataframe* to store the zone of each cell. The aggregation of every cell to a given zone continues by evaluating the cell with the smallest $∆p$ by obtaining its ID from the “*Adjacent Cells”* column. Then, the adjacent cells of the first adjacent cell are evaluated, and the definition of the zone is expanded. In case the difference is above the threshold, the cell preserves its original value in the Zone column (e.g. Zone=0 at the beginning), and the next cell evaluated is taken from one of the previous adjacent IDs. The procedure is performed until all the adjacent cells of the original random cell and its expansions have been evaluated.

Once all the adjacent cells have been identified as part of the first (Zone=1), they are internally evaluated with a second property. To determine the possible cuts according to this second property, the algorithm described in the previous paragraph is run for only those cells that are part of Zone=1. The procedure has been defined for only two properties but can be extended to more.

After identifying the first zone, Zone=1, and its internal subsets generated by the second property, the procedure continues evaluating the cells that have not been assigned to a zone yet (Zone=0). The procedure described in the first paragraph is applied to identify a new zone, and it continues until all the cells are assigned to a zone.

* + 1. Flux extraction

The last part of the data analysis extracts the fluxes between the zones. For extracting the fluxes, the procedure starts from the cell with the first ID. The zone of each cell is compared with the zone of its adjacent IDs. If any of the adjacent cells belong to a different zone, the code accesses a new column where the fluxes are stored. This is performed for all the cells of the mesh, being able to obtain all the fluxes of the boundaries. The resultant code also takes into account duplicates since the flow from cells 1→2 is the opposite of cells 1→2. To avoid duplicates, the ID of the pair is also stored for not considering the opposite pair. The last part of the code sums up all the fluxes between the zones.

* + 1. Case Studies

The framework has been evaluated in two case studies with different properties for the generation of the zones: a Venturi tube and an elbow with two inlets (one with hot air and another with cold air). The physical property considered for the division of the zones is the axial velocity in the Venturi, and the temperature and velocity for the elbow.

* 1. Results

The framework has been tested with the case studies provided, and shows consistency with the cell results reported in Ansys Fluent® as shown in Figure 2 a). Here, the case of a Venturi is presented where the threshold for cutting the regions has been defined as the quantile 85% of the distribution in differences in the axial velocity; see Figures 2,b) and 2,c). If the difference between velocities accepted for aggregation is smaller, quantile 80%, we can see that the number of zones increases, see Figure 2, c). For validating the fluxes, we have evaluated if continuity is ensured. This has been done by comparing the inflow between zones 1 and 3, which has reported no difference in the mass flow.

**Figure 2**. Comparison of axial velocity profile reported in a plane by a) Ansys Fluent®, b) zones with similar properties identified by the algorithm with a threshold defined as the quantile 85% of all the differences, and c) zones identified with a threshold defined as the quantile 80%.

The second case study evaluating the elbow considers two properties for cutting the regions, the temperature and the velocity magnitude of the flow. In this case, the zones with similar properties are more irregular than in the Venturi, highlighting the importance of the proposed tool. Generating the cuts with a single property is easier to visualize, but under more than one property, it is not that straightforward. A summary of the zones with these two properties is given in Figure 3.



**Figure 3**. Compartment model generated for the elbow with two properties. A) Front and B) back views of the elbow.

* 1. Conclusions

This work has presented a framework for compartment model generation through the cell aggregation method. For the first time, a framework aggregates the cells directly from the results in the mesh of the CFD model. This allows us to identify better the cutting regions, especially if those zones are in 3D with irregular shapes. Furthermore, this significantly advances the automatization procedure and can reduce the time in generating CMs since the novel approach minimizes user interaction. The work has only been validated for CFD schemes and it only focuses on momentum and heat transfer. Future work can extend the framework to other applications (e.g., multiphase flow simulations), coupling chemical reactions with CFD, or be part of new computational methodologies that aim to speed up CFD simulations.

**Acknowledgments**

Authors acknowledge to “Ministerio de Ciencia e Innovación” of Spanish Government for the economic support provided through the project MODYPHOS - PID2019-108378RB-I00 and the project HOLIWATER -RED2022-134350-T.

References

A. Alvarado, S. Vedantam, P. Gethals, I. Nopens (2012) A compartmental model to describe hydraulics in a full-scale waste stabilization pond. Water Research, 46 (2), 521-530.

S. K. Bermingham, H.J.M. Kramer, G.M. van Rosmalen (1998) Towards on-scale crystalliser design using comparmental models. Computers & Chemical Engineering, 22, 1, S355-S362.

F. Bezzo, S. Macchietto (2004) A general methodology for hybrid multizonal/CFD models: part II. Automatic zoning. Computers & Chemical Engineering, 28 (4), 513-525.

P. Bhalode, M. Ierapetritou (2021) Hybrid multi-zonal compartment modeling for continuous powder blending processes. International Journal of Pharmaceutics, 602, 120643.

F. Boukouvala, Y. Gao, F. Muzzio, MG. Ierapetritou (2013) Redued-order discrete element method modeling. Chemical Engineering Science, 95, 12-26.

P.V. Danckwerts (1953) Continuous flow systems: distribution of residence times. Chemical Engineering Science, 2 (1), 1-13.

A. Delafosse, M.L. Collignon, S. Calvo, F. Delvigne, M. Crine, P. Thonart, D. Toye (2014) CFD-based compartment model for description of mixing in bioreactors. Chemical Engineering Science, 106, 76-85.

B. Hernández, M.A. Pinto, M. Martin (2022) Generation of a surrogate compartment model for counter-current spray dryer. Fluxes and momentum modeling. Computers & Chemical Engineering, 159, 107664.

N. Jourdan, T. Neveux, O. Potier, M. Kanniche, J. Wicks, I. Nopens, U. Rehman, Y. Le Moullec (2019) Compartmental Modelling in chemical engineering: A critical review. Chemical Engineering Science, 210, 115196.

N.V. Queipo, R.T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, P.K. Tucker (2005) Surrogate-based analysis and optimization. Progress in Aerospace Science, 41 (1), 1-28.

Siemens (2020) Multiscale simulation in the process industry – better together. Available in: <https://blogs.sw.siemens.com/simcenter/multiscale-simulation-in-the-process-industry-better-together/>

T. Tajsoleiman, R. Spann, C. Bach, KV. Genaey, J.K. Huusom, U. Krühne (2019) A CFD based automatics method for compartment model development. Computers & Chemical Engineering, 123, 236-245.