

CFD-PBM SIMULATION OF NICKEL-MANGANESE-COBALT HYDROXIDE CO-PRECIPIATION IN CSTR

M. SHIEA*, A. QUERIO*, A. BUFFO*, G. BOCCARDO* AND
D. MARCHISIO*

* Politecnico di Torino
Department of Applied Science and Technology
Corso Duca degli Abruzzi 24, 10129 Torino, Italy
email: mohsen.shiea@polito.it, web page: <https://www.polito.it>

Key words: Precipitation, Ni-Mn-Co Hydroxide, Population balance modelling, Quadrature method of moments, Operator-splitting, load-balancing

Abstract. *The co-precipitation of $Ni_{0.8}Mn_{0.1}Co_{0.1}(OH)_2$ in a pilot-scale CSTR is simulated by adopting the CFD-PBM approach combined with the operator-splitting method. It is shown that the excessive total computational time can affect the applicability of the approach, hence necessity of using massive parallel calculations. However, the effectiveness of the parallel calculation is limited unless an algorithm is implemented to balance the load of the source integration across computing processors.*

1 INTRODUCTION

Lithium-ion batteries based on $Li(Ni_{1-x-y}Mn_xCo_y)O_2$ cathode materials have become one of the preferred battery chemistries due to their advantages such as high energy density. This fact has been driving numerous investigations in improving characteristics of these materials, which, in fact, depend on the properties of their precursor material, i.e., $Ni_{1-x-y}Mn_xCo_y(OH)_2$. As a result, the literature offers many experimental works that studied the synthesis of Ni-Mn-Co hydroxide precursors in continuous stirred-tank reactors (CSTR), which is the common system for the industrial-scale production. In particular, a lot of experimental effort has been devoted to determine the effect of operating conditions on important properties of synthesized particles, e.g., particle size distribution (PSD). However, only few modelling attempts have been made despite the fact that a predictive model can be very useful in the process scale-up and optimization. In this regard, we have recently proposed a modelling framework for the co-precipitation of Ni-Mn-Co hydroxide, which was adopted to simulate an experimental setup [1]. Here, we assess the framework by simulating the co-precipitation of $Ni_{0.8}Mn_{0.1}Co_{0.1}(OH)_2$ in a pilot-scale CSTR.

2 PROCESS DESCRIPTION

The synthesis of Ni-Mn-Co hydroxide is usually carried out in a CSTR by mixing metal sulphates with sodium hydroxide in the presence of ammonia as a chelating agent. The reactions involved are listed in Table 1, following the work by Van Bommel and Dahn [2].

Table 1: Reactions involved in the co-precipitation of Ni-Mn-Co hydroxide [2].

Reactions	Notes		
$M^{2+} + 2 OH^- \rightleftharpoons M(OH)_2$	M denotes Ni, Mn and Co		
$M^{2+} + x NH_3 \rightleftharpoons [M(NH_3)_x]^{2+}$	For Ni: $x \in 1, 2, \dots, 6$	For Mn: $x \in 1, 2, \dots, 4$	For Co: $x \in 1, 2, \dots, 6$
$NH_3 + H_2O \rightleftharpoons NH_4^+ + OH^-$			
$H_2O \rightleftharpoons H^+ + OH^-$			

The case study is a pilot-scale CSTR with three inlets and a Rushton turbine mixer (see Fig. 1). The volume of the reactor is three litres and the stirring rate 200 rpm. The concentration of the metal feed is 2M with 80% NiSO₄, 10% MnSO₄ and 10% CoSO₄. The feed solutions of NaOH and NH₃ are injected with the concentration of 5M and 10M, respectively. The flow rate of the inlets are calculated by assuming the residence time of 1 hr and the concentration ratio $[M^{2+}]:[NH_3]:[NaOH]=1:1:2$ for the mixed solution.

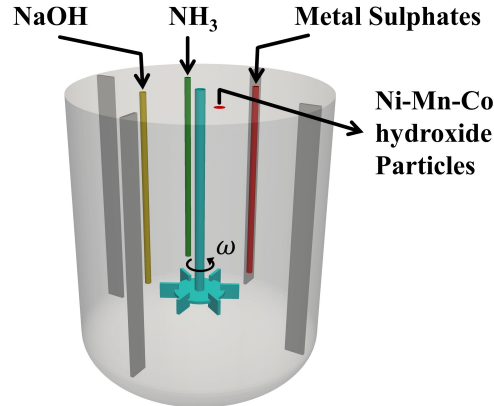


Figure 1: The geometry of the simulated CSTR

3 MODELLING AND SIMULATION

The modelling framework consists of several pieces: 1) the CFD to predict the flow fields; 2) chemical equilibria to calculate the local supersaturation generated by transported total concentrations; 3) a population balance equation (PBE) to describe the formation and evolution of particles due to the nucleation, growth, aggregation and breakage; 4) the quadrature method of moments for the solution of PBE; 5) total concentration transport equations to determine the local composition; 6) a micromixing model to consider the segregation of feed solutions. More details on the governing equations and employed models are found in our recent work [1].

The modelling framework is implemented in OpenFOAM v8.0 by modifying the "pisoFoam" built-in solver, which is suitable for the simulation of single-phase transient incompressible turbulent flows. The reason for using a single-phase solver is the fact that particles are sufficiently small (Stokes number less than 1), and therefore, their inertia has a negligible effect on the liquid flow field. Furthermore, the turbulence is solved by employing the $k-\varepsilon$ model. Moreover, the rotation of the mixer is considered by adopting the multi-reference frame method.

It should be noted that the transport equations for the total concentrations and moments are solved by the operator-splitting approach, hence the necessity of using a transient solver. This means that the solver advances in time until obtaining the steady-state solution. With the operator-splitting approach, the transport (i.e., convection and diffusion) part of the governing equations is solved first to obtain an intermediate solution at each time-step. Then, the source term of governing equations (e.g., nucleation and growth) is integrated over the time-step, taking the intermediate solution as the initial condition. The integration of the source term is done by the CVODE solver of SUNDIALS for the stiff problems, which uses backward differentiation formulas.

4 RESULTS AND DISCUSSION

Some preliminary simulations are done to assess the performance of the implementation with a particular attention to the required computational time. Considering the operator-splitting method, it is expected that the computational time is determined by the time-step used for the transport and the stiffness of source-terms. The former is related to the CFL condition imposed by the flow field, while the latter is mainly due to rate of particle nucleation and growth. Therefore, in these preliminary simulations, only nucleation and growth of particles are included. Once the viability of the approach is verified, the addition of the aggregation and breakage of particles should not pose a serious problem.

Fig. 2 shows the evolution of the SMD in time at the reactor outlet. As can be seen from Fig. 2, the steady-state is achieved after more than five times the residence time, which requires more than 180000 time-steps of, for instance, 0.1 (s). In this simulation, each time-step of 0.1 s takes around 7 seconds by using 44 cores on a machine with two Intel(R) Xeon(R) Gold 6248R processors, which corresponds to a total computational time of two weeks. This amount of time will increase in the case of a larger reactor with more extreme conditions such as higher agitation rate and longer residence time. Therefore, it is important to find remedies for decreasing the total computational time. It should be mentioned that decreasing the total computational time by taking larger time-steps (i.e., reducing the total number of time-steps) is not as effective as expected, because the load of the integration over each time-step increases. In fact, there is a trade-off between the total number of time-steps (in other words, the length of the time-step) and the integration load over each time-step. Moreover, too much increase of the time-step can cause simulation instabilities and a larger splitting error.

A possible solution to decrease the total computational time is to reduce the calculation time of each time-step by increasing the number of processors. Unfortunately, the con-

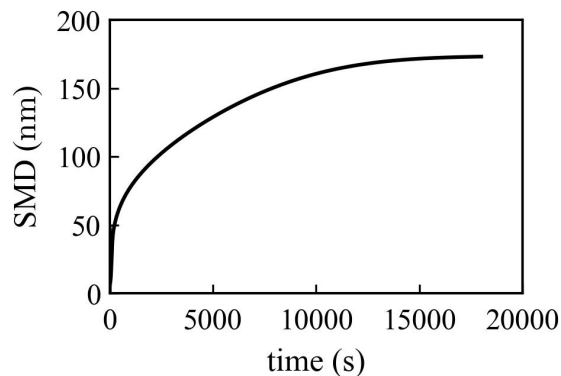


Figure 2: The evolution of the Sauter mean diameter at the reactor outlet

ducted trials show low scalability of the source integration part of the operator-splitting method when the number of processors increases. It is unexpected since the integration depends only on the variables of each cell, and therefore, it should be highly scalable. The reason lies in the fact that the integration load differs from one computational cell to another as the nucleation and growth rates can change significantly over the domain. It eventually causes an imbalance of the integration cost across processors when the common strategies for the domain decomposition in parallel CFD codes are used. These strategies usually aim at assigning the same number of cells to each processor while minimizing the communication between them. It is obvious that changing the criteria for the domain decomposition, e.g., based on the integration load, is not a promising solution, as it can cause similar load imbalance in the solution of transport equations. Therefore, we are currently investigating possible algorithms to distribute equally the integration load over processors, without changing the common practice for the domain decomposition.

5 ACKNOWLEDGEMENT

The research reported in this work was funded by European Union, Horizon 2020 Programme, SimDOME Project, Grant Agreement No 814492. The views and opinions expressed in this publication are the sole responsibility of the authors and do not necessarily reflect the views of the European Commission/Research Executive Agency.

REFERENCES

- [1] M. Shiea, A. Querio, A. Buffo, G. Boccardo, and D. Marchisio, “CFD-PBE modelling of continuous Ni-Mn-Co hydroxide co-precipitation for Li-ion batteries,” *Chem. Eng. Res. Des.*, vol. 177, pp. 461–472, 2022.
- [2] A. Van Bommel and J. Dahn, “Analysis of the growth mechanism of coprecipitated spherical and dense nickel, manganese, and cobalt-containing hydroxides in the presence of aqueous ammonia,” *Chem. Mater.*, vol. 21, no. 8, pp. 1500–1503, 2009.