

VIP Very Important Paper

Accelerated Optimization Methods for Force-Field Parametrization in Battery Electrode Manufacturing Modeling

Teo Lombardo,^[a, b] Jean-Baptiste Hooek,^[c] Emiliano N. Primo,^[a, b] Alain C. Ngandjong,^[a, b] Marc Duquesnoy,^[a, b] and Alejandro A. Franco^{*[a, b, d, e]}

The performance, durability and cost of modern Li-ion batteries (LIBs) strongly depend on their manufacturing process. In this regard, computational methods that attempt to model LIBs manufacturing have the potential to help experimentalists to reduce costs and improve performances of LIBs. To do so, the electrode slurry phase, consisting of a suspension of active material, carbon additive and binder in a solvent must be modeled at first at the mesoscopic scale. However, efforts in that sense are rare in literature due to the inherent complexity of slurries and the difficulty of setting up appropriate metrics for the validation of the modeling results. For the first time, we

propose an approach relying on experimental data that allows to validate the properties of simulated Nickel-Manganese-Cobalt-based slurries with different compositions and solid contents through force fields (FFs) parametrization. The latter was attained by Coarse-Grained Molecular Dynamics (CGMD), which enables to model the slurry phase at the mesoscopic scale and to consider explicitly the carbon binder domain. Moreover, we demonstrate that the CGMD FFs parameterization can be accelerated thanks to different types of Particle Swarm Optimization-based algorithms, which would allow faster screening of different simulated slurries fabrication conditions.

1. Introduction

Since their first commercialization in the early 90's, Li-ion batteries (LIB) are shaping the way we live, revolutionizing the world of portable device technologies, opening up new frontiers in the automotive sector and promising a widespread use of renewable energy sources.^[1–5] To meet these requirements, the performance and durability of LIBs need to be further improved through the utilization of new materials and

the optimization of their manufacturing process, under the constraints of reducing its cost and environmental impact.^[6] Nowadays, in view of its complexity, LIBs manufacturing optimization is carried out by a trial and error approach.^[7] Computational simulation tools can be very useful to “optimize the optimization” on the basis of a model capturing the slurry physics and the corresponding electrode self-organization along all the manufacturing steps.^[8] This is particularly challenging due to the discrete, multicomponent and multi-scale nature of the slurry, constituted by active material (AM), carbon additive, binder and solvent. Moreover, the importance of 3D mesoscale modeling to disclose the links between the electrode mesostructure and its electrochemical behavior starts to be widely recognized.^[8–13] In general, available 3D models are built on 3D-images arising from tomography characterizations, which only show the AM particles shapes and location, but do not provide any information about the carbon binder domains (CBD) phase location.^[14] Indeed, the low atomic weight and the very low X-ray cross section of carbon (compared to the cathode AMs typically used) makes extremely challenging to resolve the CBD phase by using the imaging techniques today available.^[15,16] The latter obliges researchers to use a “binder-bridge” approach to generate artificially the CBD phase.^[13,17,18] Instead, the use of electrode mesostructures obtained by experimentally validated mesoscale simulations taking explicitly into account the CBD phase lead to three main advantages: (i) there is no need to artificially generate *a posteriori* the CBD phase in the electrode mesostructures, (ii) it is possible to link the particles physical properties (through the parameters of the model) with the resulting mesostructure, and (iii) the time and resources costs to generate a simulated electrode mesostructure is expected to be considerably lower

[a] T. Lombardo, Dr. E. N. Primo, Dr. A. C. Ngandjong, M. Duquesnoy, Prof. Dr. A. A. Franco
Laboratoire de Réactivité et Chimie des Solides (LRCS), UMR CNRS 7314
Université de Picardie Jules Verne, Hub de l'Energie
15 rue Baudelocque, 80039 Amiens Cedex 1, France
E-mail: alejandro.franco@u-picardie.fr

[b] T. Lombardo, Dr. E. N. Primo, Dr. A. C. Ngandjong, M. Duquesnoy, Prof. Dr. A. A. Franco
Réseau sur le Stockage Electrochimique de l'Energie (RS2E)
FR CNRS 3459, Hub de l'Energie
15 rue Baudelocque, 80039 Amiens Cedex 1, France

[c] Dr. J.-B. Hooek
Université de Picardie Jules Verne
Plateforme MatriCS, UFR des Sciences
33 rue Saint-Leu, 80039 Amiens Cedex 1, France

[d] Prof. Dr. A. A. Franco
ALISTORE-European Research Institute, FR CNRS 3104
Hub de l'Energie,
15 rue Baudelocque, 80039 Amiens Cedex 1, France

[e] Prof. Dr. A. A. Franco
Institut Universitaire de France
103 Boulevard Saint Michel, 75005 Paris, France

Supporting information for this article is available on the WWW under <https://doi.org/10.1002/batt.202000049>

© 2020 The Authors. Published by Wiley-VCH Verlag GmbH & Co. KGaA. This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

than the one needed to obtain experimental 3D electrode tomography images, considering the waiting time to the instruments access (often located in synchrotrons), samples preparation, measurement and data analysis. The latter includes the challenging segmentation stage which allows separating CBD from AM and pores in the images.

Our ARTISTIC project^[19] aims to develop a digital twin of the LIB manufacturing process through a multiscale computational platform able to predict the impact of the manufacturing parameters on the final 3D-resolved electrode properties and the associated electrochemical performance. The platform encompasses discrete and continuum models and machine learning (ML) methods with the ultimate goal of disclosing the interdependencies between electrode manufacturing parameters and their electrochemical performance and, thanks to this knowledge, to develop consistent inverse-design models. In this sense, in recent publications we simulated Nickel-Manganese-Cobalt (NMC)-based cathodes manufacture through the use of Coarse Grained Molecular Dynamics (CGMD).^[9–12] The CGMD is a mesoscale simulation technique which uses Force Fields (FFs) to describe the physicochemical interaction between beads representing AM and CBD during the slurry phase equilibration and drying steps, while continuum models are able to capture the impact of the resulting electrode mesostructure on the electrochemical response.^[9–12] The starting point of such computational workflow is the simulation of the slurry phase. However, in all these works we did not elaborate in detail the procedure adopted to parameterize the CGMD FFs and to validate the simulated slurries.

At the experimental level, it is widely known that the slurry rheological properties strongly affect the manufacturing process.^[20–22] Indeed, the slurry mesostructure and its physicochemical properties determines the coating process and the resulting electrode mesostructure. The interdependency between the slurry properties and some electrode characteristics, as its mass loading and porosity, were clearly demonstrated by us in a recent work combining classification ML methods and *in house* experimental data.^[23]

The objective of this work is two-fold. Firstly, the validation procedure of the CGMD FFs parameterization is explained in detail. Secondly, we investigate the efficiency of optimization tools able to accelerate such FFs parameterization. More specifically, with the support of *in house* viscosity experiments, the sensitivity of our CGMD model was verified as to whether it is appropriate to discriminate between different slurries with similar rheological properties using different FFs optimization approaches: manual and *in house* automated optimization algorithms based on Particle Swarm Optimization (PSO).

2. Results and Discussion

2.1. Manual Optimization

Within the CGMD formalism the AM is treated as a collection of spherical particles with a certain density and size distribution, interacting between them and with CBDs through the

FFs.^[9,11,12,24] The CBDs are modeled as effective spheres, which represent a certain mass of carbon, binder and (for the slurry phase) solvent in a volume with an effective density. The FF used to mimic the slurry physicochemical properties is a combination of the Lennard-Jones FF (LJFF) and the Granular Hertz FF (GHFF), which are both available in LAMMPS computational software^[25] and that are reported in Equations (1)–(5):

$$E_{LJ}(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right] \quad (1)$$

$$F_{LJ}(r) = -\nabla E_{LJ}(r) \quad (2)$$

$$\text{if } r \leq r_c \quad F_{LJ/SF}(r) = F_{LJ}(r) - F_{LJ}(r_c) \quad (3)$$

$$\text{if } r > r_c \quad F_{LJ/SF}(r) = 0 \quad (4)$$

$$F_{GH} = \sqrt{\delta} \sqrt{\frac{R_i R_j}{R_i + R_j}} \left[(k_n \delta n_{ij} - m_{eff} \gamma_n v_n) - (k_t \Delta s_t + m_{eff} \gamma_t v_t) \right] \quad (5)$$

The LJFF is used to describe the adhesive forces between particles and their bonds, due to the presence of binder in the suspension [Eqs. (2), (3) and (4)]; it depends on three parameters: ϵ , σ and r_c . The first one represents the potential well of the LJ potential curve, σ is the distance at which the interparticle potential is equal to zero and r_c is the cut-off distance at which two particles do not interact anymore. These three parameters are inputs of the CGMD simulation. The GHFF [Eq. (5)] is used to model the mechanical properties of the system and it is composed by a normal and a tangential term. k_n , k_t , γ_n , γ_t , v_n and v_t represent the normal and tangential elastic constants, viscoelastic damping constant and relative normal and tangential velocity components between two interacting particles, respectively. Δs_t , m_{eff} and n_{ij} represent the tangential displacement vector, the effective mass $\left(\frac{m_i m_j}{m_i + m_j} \right)$ and the unit vector along the line connecting the center of the two interacting particles. R and δ are the particles radii and the overlap distance, respectively. It should be underlined that for $\delta = 0$, F_{GH} is equal to 0. Consequently, to take into account the GHFF, a certain degree of particles overlap needs to be allowed. For this reason, in the simulations, the diameter of the AM particles is increased by 14% respect to the experimental diameter (d_{AM_exp}), mimicking the presence of surrounding solvent molecules. The last parameter needed for the simulation is the friction coefficient (χ_u), defined as the maximum ratio between the tangential and the normal forces. Lastly, because the CBD is an effective particle, its diameter (d_{CBD}) and density (ρ_{CBD}) constitute also effective parameters that can be used to optimize the FFs. For more details on these FFs, the interested readers are referred to the Supporting Information, Section 1.

The parameters described above can be used to modulate the physicochemical properties of the simulated slurry. Consequently, for their correct parametrization, experimental

viscosity curves are used. Particularly, shear viscosity curves (η in terms of the shear rate $\dot{\gamma}$) provide information relative to structural or alignment changes that the particles suspension undergoes upon the application of a stress deformation or strain, similar to what it happens during the coating process. In this work, two different slurry compositions and, for each of them, three solid contents (SCs) were studied with the CGMD model and confronted to experimental data (Table 1). Here, we define the SC as the proportion of slurry solid components (AM, conductive carbon additive and binder) relative to the slurry mass (solid components and solvent). All the information about the experimental slurries manufacturing and characterization can be found in the Experimental Section.

The initial slurry mesostructure (Figure 1A) was generated in order to match the experimental composition, in terms of AM and CBD mass ratio, and AM particle size distribution (PSD). Six AM particle sizes were defined, whose dimensions are reported in the legend of Figure 1. Section 2, in the Supporting Information, offers all the details about the method used to generate the initial structure. From that initial arrangement, the slurry equilibration was performed through a CGMD simulation in the NPT environment (Figure 1A), until the system is stable at the temperature and pressure chosen (300 K and 1 atm), leading to the 3D structure of Figure 1B. Afterwards, a non-equilibrium CGMD (NE-CGMD) simulation was performed to determine the η for selected values of $\dot{\gamma}$. Even if both equilibrium CGMD and NE-CGMD can be used for the latter scope, the first one works better with low viscosity fluids, typically below 20 mPas.^[26] Considering that the typical slurries used for LIB manufacture are highly viscous we have chosen

Table 1. Analyzed slurry compositions (NMC:carbon additive:binder - wt %) and solid contents (SC).

	96:2:2	95:2.5:2.5
SC 1	73%	70%
SC 2	71%	68%
SC 3	70%	65%

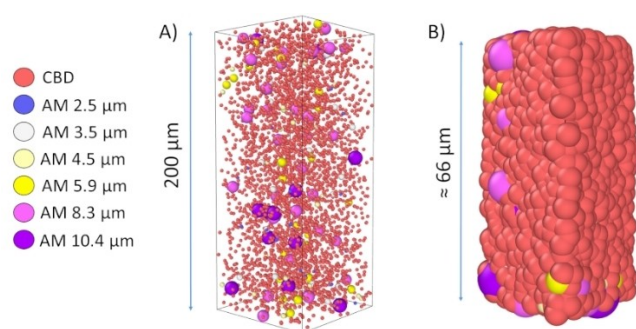


Figure 1. Example of a simulated slurry (unit cell, periodic boundary conditions are assumed along the three Cartesian directions). A) Initial structure, where the AM and CBD particles are randomly placed. B) Final equilibrated structure at 300 K and 1 atm. Both structures represent the composition 96:4 (NMC:CBD - wt %), SC 71 %. The legend on the left side reports the different particles sizes for the AM with its corresponding colors.

the NE-CGMD method, whose working principle is the same as a classical NEMD and it is schematically described in Figure S3.

The $\dot{\gamma}$ range of interest for industrial scale processing varies between tens to hundreds Hz.^[27] Each analyzed $\dot{\gamma}$ corresponds to an independent NE-CGMD simulation with identical FFs parameters and slurry mesostructure. Section 3 and 4, in the Supporting Information, gives further details regarding how the generation of the initial structure affects the uncertainty of to the simulated η - $\dot{\gamma}$ curve.

Figure 2 presents the comparison between our experimental and simulated η - $\dot{\gamma}$ curves for different compositions and SCs (see Table 1). The experimental curves were obtained by us following the procedure described in the experimental section and are an average of two η - $\dot{\gamma}$ experiments for two different slurries measured by using a rheometer (Kinexus lab+, Malvern Instruments). Forouzan et al., who firstly proposed a CGMD approach for simulating slurries, reported similar results for two η - $\dot{\gamma}$ curves for pure CBD and for AM plus CBD.^[24] The main differences between the slurries analyzed in this work and in the one of Forouzan et al. are the AM PSD, the slurries composition, the chemical composition of NMC (NMC111 vs. NMC523) and the nature of the carbon additive (C-ENERGY™ super C65 vs. C45) used. Moreover, in their work the model sensitivity towards the slurry SC was not analyzed and only one

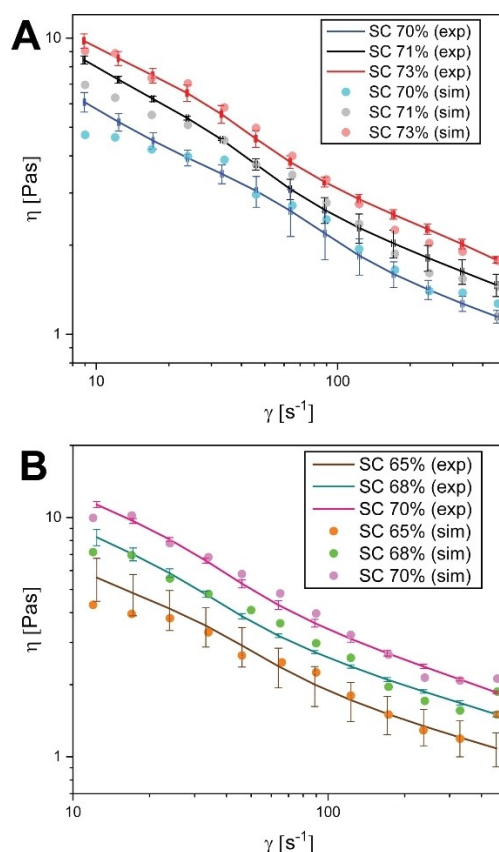


Figure 2. Viscosity curves for different compositions (NMC:carbon black: PVdF) and solid contents. A) 96:2:2; B) 95:2.5:2.5. The lines represent the experimental curves, while the dots the simulated data. The error bars represent the experimental confidence intervals with a confidence level of 95%.

composition (90:5:5, NMC:carbon additive:binder) was used to parameterize their slurry FF. In addition, the fitting correspondence between the experimental and simulated η - $\dot{\gamma}$ curves was not reached for all the shear rate values analyzed: differences (estimated to ca. 5 to 40%) can be noticed between these curves even if the graphs are not reported in log-log scale.^[24] Our results in Figure 2 demonstrate that it is possible to find different sets of parameters for the FFs that can fit in a consistent way experimental η - $\dot{\gamma}$ curves corresponding to different SCs and compositions.

As starting point in the quest of the right FFs parameters (hereafter called optimization method), the same values proposed in Faraouzan et al. work were tested. However, they were not suited for our compositions, SCs and AM PSD. Therefore, the FFs optimization method was started by focusing at one single point ($\dot{\gamma} = 14 \text{ s}^{-1}$) until reaching a reasonable η value, compared to the experimental one. Then, the entire η curve was simulated for different sets of FFs parameters. The optimal FFs parameters obtained after the manually performed parametrization are reported in the Supporting Information (Table S1 and S2).

To further test the validity of our manual parametrization method, the consistency between experimental and simulated densities was also verified (Figure 3). The densities of the simulated systems were calculated as the total mass of the particles divided by the final volume of the simulation box, while the experimental ones were acquired by using a density meter (DMA4500, Anton Paar GmbH). The results shown in Figure 3 further demonstrate the validity of the parametrized model in reproducing the slurries physical properties.

Finally, the sensitivity of the simulated slurries mesostructures relative to the different compositions and SCs of interest can be captured by calculating their associated radial distribution functions ($g(r)$). Figure 4A reports the $g(r)$ in terms of the distance between AM and CBD particles for two slurries with constant composition (95:2.5:2.5) and different SCs (68 and

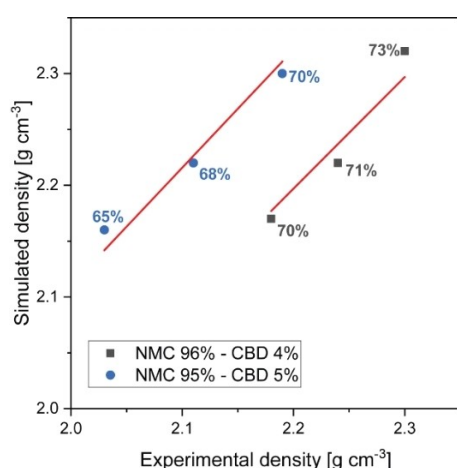


Figure 3. Correlation plot between experimental and simulated densities. For each composition, the SCs are reported next to their corresponding point. Pearson's correlation coefficients and p -values for each composition are, respectively: (NMC 96%-CBD 4%) 0.99 and 0.05, (NMC 95%-CBD 5%) 0.999 and 0.04.

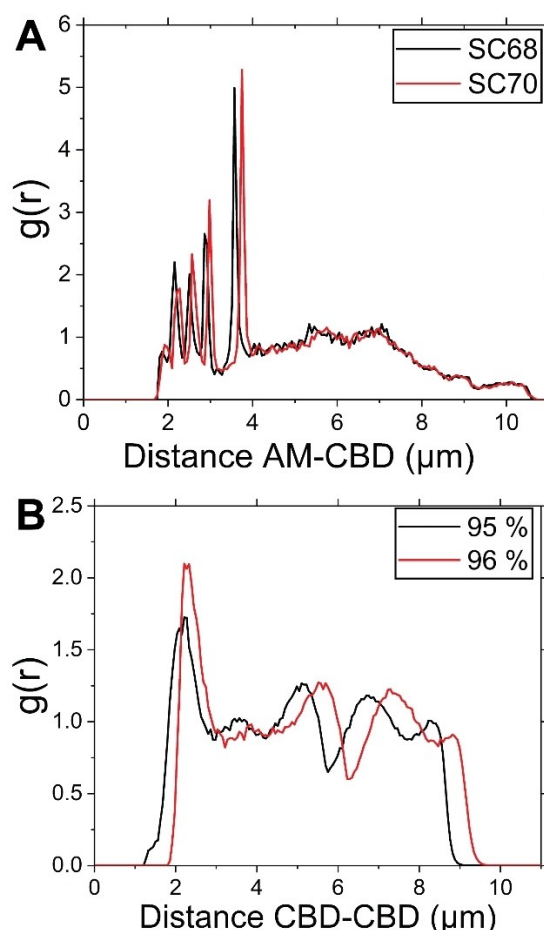


Figure 4. $g(r)$ in terms of A) the distance between AM and CBD particles for slurries with constant composition (95:2.5:2.5) but different SCs (68 and 70%) and B) the distance between CBD and CBD particles for slurries with constant SC (70%) but different composition (96:2:2 and 95:2.5:2.5).

70%). Particularly, the curve for the case of SC 70% is shifted to higher AM-CBD distances. The latter can be explained by considering that in order to correctly fit the viscosity curve for the case of SC 70% its σ_{AM} was increased by $\sim 8\%$ (Table S1 and S2), which leads to higher equilibrium distances between these particles. Figure 4B shows the $g(r)$ in function of the distance between CBD particles for two slurries with constant SC (70%) and different compositions (96:2:2 and 95:2.5:2.5). The main difference in terms of FFs parameters for these two simulated slurries is their ϵ_{CBD} values, which was incremented from approximately zero (~ 0.006) to a significantly higher value (~ 6), respectively, in order to capture the higher amount of binder in the system. The latter leads to a shift to lower equilibrium distance between two CBD particles going from higher to lower AM mass ratio. Moreover, the results reported in Figure 4 demonstrate how the simulated slurry mesostructure is more sensitive to some parameter, as σ , than others, as ϵ .

2.2. Accelerated Optimization Methods

Through the manual FFs optimization method, it took roughly 6 months to obtain the first $\eta\text{-}\dot{\gamma}$ curve (96:2:2, SC 71 %) in accordance to the experimental one, by using 12 nodes (128 GB of RAM each), each composed of 2 processors (Intel® Xeon® CPU E5-2680 v4 @ 2.40 GHz, 14 cores). This method is therefore computationally expensive and time consuming, which would make it impractical for use in a flexible LIB manufacturing process model aiming at simulating a wide spectrum of SCs, formulations and chemistries. To tackle this problem, four different optimization methods were developed and their efficiency compared: a method based on the PSO theory, and three other methods that we call in the following Human-Driven PSO (H-D PSO), Machine Learning-Driven PSO (ML-D PSO) and Learning PSO (LPSO).

The PSO is a stochastic global optimization algorithm based on swarm behavior, firstly proposed by Kennedy and Eberhart.^[28,29] It has been already used in many domains, demonstrating its robustness.^[30–32] The most known application of PSO in material science is the CALYPSO method, firstly proposed by Ma et al.,^[33–37] which allows to predict the energetically stable/metastable crystal structure at a given chemical composition and external conditions. However, it has never been used in the context of electrode manufacturing simulations. The working principle of our PSO-based algorithms is schematized in Figure 5 for the simpler case of a simulation in which only two parameters, p_1 and p_2 , need to be optimized. In this scheme, each point (n) in the graph represents a certain slurry mesostructure, which is obtained by using the associated p_1^n and p_2^n parameters values for the CGMD simulation. The number of points (hereafter called particles, to be consistent with the nomenclature of the PSO theory) was limited to four for the sake of clarity. At the i -th iteration (Figure 5A), four different CGMD simulations are launched in parallel, by using different parameters values. At the first iteration (iteration 0), their values are given randomly. Afterwards, the code recovers automatically the slurries mesostructures and densities. The densities are stored as a result, while the mesostructures are used to launch in parallel the NE-CGMD calculations of the η at the $\dot{\gamma}$ of interest. Once these simulations are ended, the calculated $\eta\text{-}\dot{\gamma}$ curves are automatically stored. The results obtained for each PSO particle are analyzed in order to determine how “close” or how “far” the simulated results are relative to the experimental ones. In order to quantify it, a cost function [Eq. (6)] was defined:

$$\text{Cost} = (1 - \eta_{\text{norm}}^i)^2 + (1 - \rho_{\text{norm}}^i)^2 \quad (6)$$

$$\eta_{\text{norm}}^i = \frac{\eta_{\text{sim}}^i}{\eta_{\text{exp}}} \quad (7)$$

$$\rho_{\text{norm}}^i = \frac{\rho_{\text{sim}}^i}{\rho_{\text{exp}}} \quad (8)$$

where ρ_{sim} , η_{sim} , ρ_{exp} and η_{exp} represent the simulated and experimental densities and viscosities, respectively. Note that

the same weights were used for the density and viscosity. Being the FF parameters more sensitive to the latter, the viscosity will most likely drive the parametrization. However, the weights associated to these physical properties can be modulated in all the PSO-based algorithms developed along this work.

After the evaluation step the FFs parameters should be modified in order to launch the new simulations at the step $i + 1$. In order to do so, for each particle two elements are needed: (i) the particles velocity (defined in Eq. (9) and given randomly at the iteration 0), and (ii) the vector linking the particle to move with the particle for which the lowest cost value was obtained. The latter is described in Figure 5B, C and D for the case of the particle circled in brown. In this scheme, the blue vectors represent the particles velocities at the iteration i , while the red vectors link each particle to the one which has the lowest Cost value.

The position of the PSO particle of interest at the iteration $i + 1$ will be defined by the vector addition of these two vectors (represented as purple vectors in Figure 5 C–E). This operation is performed for all the particles to obtain their position at the iteration $i + 1$. Afterwards, the new simulated mesostructures are calculated and the loop restarts. It should be underlined that Figure 5 represents a simplification of the actual procedure followed by our PSO-based algorithm and it is strictly valid only for the case of $i = 0$. For the sake of clarity, this scheme does not take into account the concept of personal best, discussed below. In this work the PSO-based algorithm was used to optimize seven parameters (ε_{AM} , ε_{CBD} , σ_{AM} , k_n , γ_n , ρ_{CBD} and X_u) by using eight PSO particles (one for each node used for the optimization procedure). Overall, by incrementing the number PSO particles the algorithm would sample larger portions of the parameters space, at the expense of higher computational cost. Therefore, a trade-off between the number of PSO particles used and the computational resources available should be done.

Equations (9) and (10) define more rigorously how the PSO particles positions (i.e., its FFs parameters values) and velocities are calculated after each iteration:

$$v_m(t+1) = wv_m(t) + r_1c_1(p_m(t) - x_m(t)) + r_2c_2(g(t) - x_m(t)) \quad (9)$$

$$x_m(t+1) = x_i(t) + v_i(t+1) \quad (10)$$

The time variable in Eq. (9) and (10) refers to the PSO algorithm iterations. The three terms in Equation (9) are commonly known as the inertia term and the cognitive and social components, respectively.^[29] $v_m(t)$ and $x_m(t)$ represent the m -th particles velocity and position at a certain time t , respectively. $p_m(t)$ and $g(t)$ represent the position of the personal and global best, respectively. The personal best is the position in the n -dimensional space in which each particle has obtained its best result (lowest value of the cost function), while the global best is the position in the space that has given the best result considering the whole optimization procedure (up to the time t). w , c_1 and c_2 are defined as the inertia,

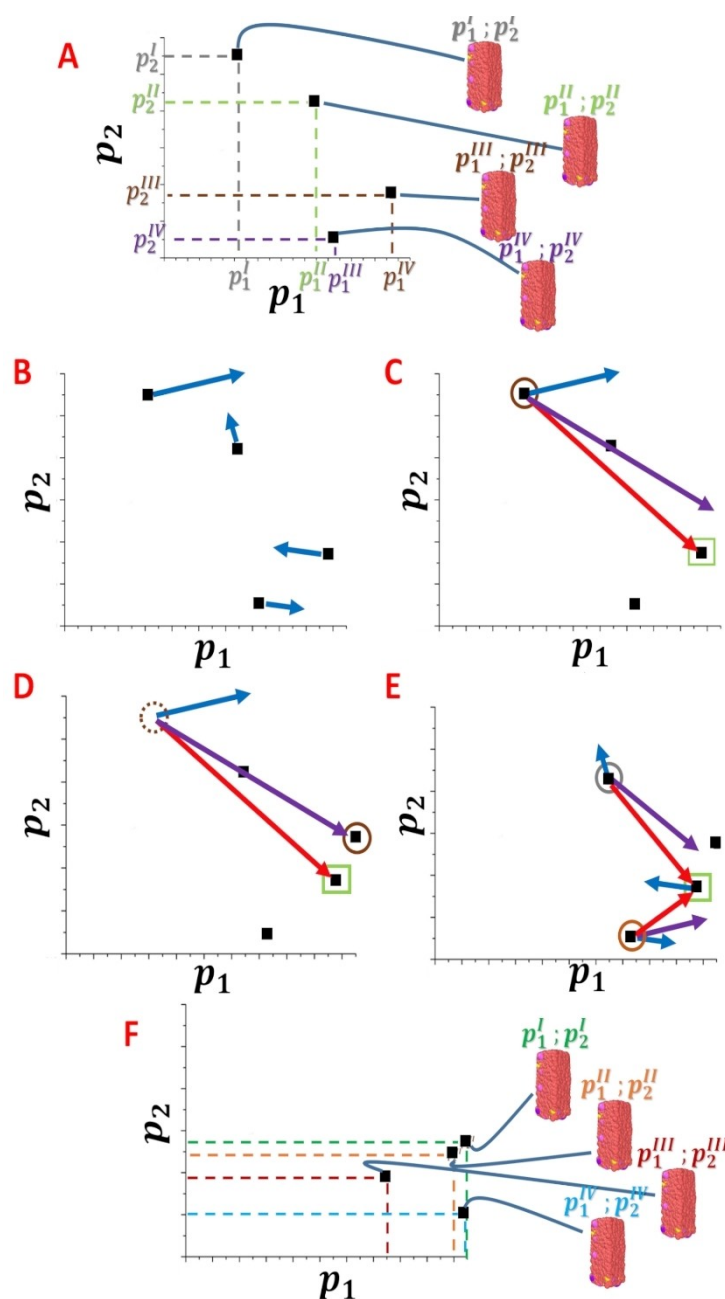


Figure 5. Scheme of the PSO-based algorithm working principle. A) Simulated slurries and associated FFs parameters at the iteration i , B–E) Evolution of the FFs parameters values in terms of the results obtained at the iteration i and F) Simulated slurries and associated FFs parameters at the iteration $i + 1$.

personal acceleration and global acceleration coefficients, respectively. Their values were fixed to 1, 2.05 and 2.05.^[38] Finally, r_1 and r_2 are random coefficients that can range between zero and one. All the PSO-based algorithms were developed in C++ programming language. For more details regarding the implementation of the PSO-based algorithm, the interested readers are referred to the Supporting Information, Section 6.

The difference between the classical PSO and the H-D PSO and ML-D PSO is that the parameters of one particle out of eight is controlled, respectively, by a human operator or a ML algorithm (Deep Neural Network), while the other seven are

controlled by the PSO algorithm. However, for the case of H-D PSO the operator selects the parameters of one particle only during the first iteration, while the following ones are totally controlled by the PSO. The latter can give to the algorithm a hint of which zone of the parameters space is the most interesting one to analyze (lower cost function respect to the ones obtained by using random FFs parameters). On the other hand, the ML algorithm determines the parameters of one particle for the whole optimization procedure. The latter is described schematically in Figure 6. The comparison between the results obtained by H-D PSO and ML-D PSO allows to verify if the latter was able to accelerate the optimization procedure

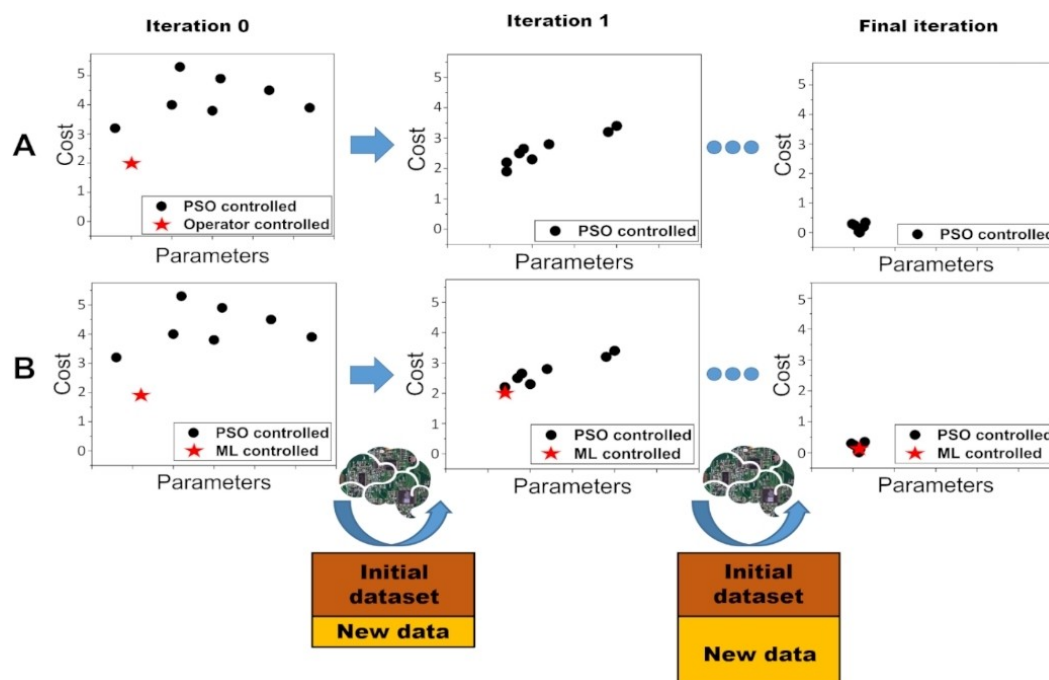


Figure 6. Scheme of the working principle of A) Human-Driven PSO and B) Machine Learning-Driven PSO.

at least as much as an expert operator, but with the advantage of being completely automatized. The ML algorithm was built using a Python-based code that relies on the Keras library. Finally, the ML algorithm dataset was constructed with the results previously obtained during the PSO and the manual optimization procedures (by excluding the optimal ones that lead to the results showed in Figures 2 and 3). It should also be underlined that this dataset is updated iteration by iteration during the ML-D PSO by adding to the dataset the results obtained during the previous iteration, as showed in Figure 6 B.

Figure 7 reports the average cost function [Eq. (6)] for the eight particles (per iteration step) as function of the number of iterations for PSO, H-D PSO and ML-D PSO. The curves show clearly that our PSO-based algorithms are effective in terms of FFs parameters optimization, *i.e.* the average cost function decreases rapidly during the optimization procedure, especially during the first iterations. However, even if the classical PSO displays acceptable performance during the first iterations, it reaches a local minimum from which it was not able to get out from. This drawback could be solved by introducing in the PSO code a procedure “to escape” more easily from local minima, by changing randomly the FFs parameters of the eight particles, if the global best does not change after a certain number of iterations. Concerning the H-D and ML-D PSO, both of them were able to reach the global minimum and the cost function decreases even faster than the classical PSO. Moreover, the performances of the H-D and ML-D PSO are comparable, which shows that in this context a ML algorithm can be used to accelerate the optimization procedure at least as much as an expert operator, but through a fully automated approach. Furthermore, the performances of the ML-D PSO are expected

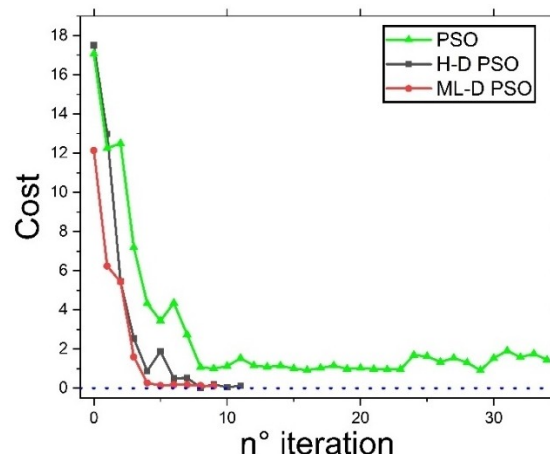


Figure 7. Average cost function for the eight PSO particles per iteration step as a function of the number of iterations for three Particle Swarm Optimization algorithms: classical PSO (green), Human-Driven PSO (black) and ML-Driven PSO (red). The blue dotted line serves as a reference for zero cost. The results refer to the slurry composition 96:2:2 (AM:C:PVdF), SC 71 %.

to improve by increasing the dataset size and by improving the ML model.

We compare in the following the computational cost of the PSO, H-D PSO and ML-D PSO. The calculations were performed using eight nodes (384 GB of RAM each), each composed of 2 processors (Intel® Xeon® Gold 6148 CPU @ 2.40 GHz, 20 cores). To estimate the time needed to obtain the curves showed in Figure 7, it should be taken into account that each iteration step needs around 36 h. Considering this, in order to reach an average cost equal to zero the HD-PSO and ML-D PSO needed *ca.* 14 and 8 days, respectively. The latter demonstrate that the

PSO-based approaches proposed in this work are considerably faster than the manual optimization approach (needing approximately six months by using similar computational resources, as mentioned in the previous Section). Furthermore, it should be stressed that to decrease the time needed to test and validate all the PSO-based algorithms developed, during the optimization we focused on the slurry density and on the η for only two $\dot{\gamma}$, i.e., 18 s^{-1} and 125 s^{-1} . This simplification was sufficient to obtain a good fitting relative to the full experimental η - $\dot{\gamma}$ curves, as shown in Section 6.1 of the Supporting Information.

The previously discussed PSO-based algorithms optimize the FFs parameters for the η - $\dot{\gamma}$ curve of a certain composition and SC (96:2:2 and SC equal to 71% for Figure 7). This would mean that the PSO-driven FFs optimization should be repeated when optimizing the η - $\dot{\gamma}$ curve of another slurry with the same composition but different SC. However, Figure 2 shows that by changing the SC, the η - $\dot{\gamma}$ curve does not change dramatically. Consequently, the information obtained during the previous optimization procedure can be used to accelerate the FFs parametrization in other similar conditions. The Learning-PSO (LPSO) optimization method was developed following this idea. At the beginning of the optimization procedure the LPSO recovers all the results obtained from the previous optimization (s), i.e., the density and η values associated to specific sets of FFs parameters. These simulated results are compared to the new experimental conditions and the eight best results (the closest to the new η - $\dot{\gamma}$ curve and slurry density) are identified. The latter corresponds to the iteration 0 of the LPSO. Afterwards, the FFs parameters associated to these results are modified randomly to launch eight new CGMD simulations, whose results will be associated to the iteration 1 of the LPSO. From this point onwards, the optimization procedure follows the workflow of the classical PSO (Figure 5).

Figure 8 A depicts the average cost function in terms of the number of iterations for the LPSO procedure, in which all the previous results (PSO, H-D PSO and ML-D PSO) were used as initial dataset. For both 70% and 73% SCs, the average cost function is approximately equal to zero from the iteration 0. At the first iteration the function increases due to the random modification of the parameters value, while with further iteration steps it decreases and oscillates around zero. Particularly, the cost function approximately equal to zero at iteration 0 indicates that the initial dataset (obtained by using the PSO, ML-D PSO and H-D PSO) already contained set of FFs parameters values suited for all the SCs analyzed for the slurry composition 96:2:2. Consequently, the results obtained at the iteration 0 would have been enough to determine FFs parameters values suited for both SC 70 and 73%. The further iterations were performed only to verify that the LPSO code would correctly converge to approximately the same FFs parameters values obtained at the iteration 0. All the above represents a further advantage of the PSO approaches in terms of saving time and computational resources. Indeed, it indicates that during the FFs parametrization for a specific slurry it is possible to find parameters sets suited for slurries similar to the one studied (as in the case of slurry with constant composition

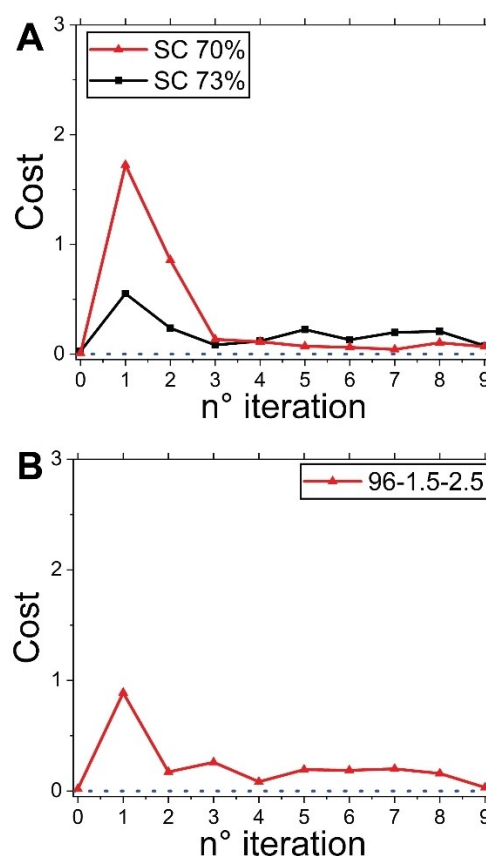


Figure 8. Average cost function for the eight PSO particles as a function of the number of LPSO iterations performed for the A) composition 96:2:2 (AM:C:PvDF), SC 70% (red curve) and 73% (black curve) and B) compositions (AM:C:PvDF) 96:1.5:2.5 (red curve), SC 69%. The blue dotted lines serve as a reference for zero cost.

but different SCs). Moreover, this procedure does not only apply to the case of different SCs and constant composition, but also for different carbon and binder ratios (keeping constant the amount of AM), as showed in Figure 8 B for the case of the composition 96:1.5:2.5, SC 69%.

We further tested the possibility of using only ML algorithms to directly predict the FFs parameters for a certain slurry composition and SC of interest. This approach gave accurate results for the case study 96:2:2 – SC 70%, as showed in the Supporting Information, Section 7. The use of ML algorithms promises to strongly accelerate the parametrization, considering their extremely low computational cost. However, the dependence of ML methods on the quantity of the data available could limit the applicability of this approach: a deeper investigation by using ML methods will be carried out by us in the future.

Finally, all the PSO-based approaches developed in this work can be adapted to other modeling approaches for other electrode fabrication steps, such as solvent evaporation or calendering.^[39–42] The latter can be used to accelerate the current efforts in developing a trustable digital twin of the LIBs manufacturing line, which has the potential to boost the optimization of the manufacturing parameters in order to

improve LIBs performance, lifetime, recyclability and safety, at minimal cost.

3. Conclusions

In this manuscript, a trustable approach to validate CGMD-calculated 3D slurry mesostructures was presented. We believe that the validation of the slurry phase is of paramount importance for any simulation approach based on physical models whose goal is to capture the effect of manufacturing parameters on the resulting electrodes properties. The latter is of preeminent relevance to develop a digital twin of the LIB manufacturing process in order to accelerate the optimization of the manufacturing parameters. Two approaches to correctly parametrize the FFs parameters were tested in order to find the most efficient one in terms of time saved. Firstly, a manual optimization approach has led to excellent fittings. However, this approach is computationally very expensive. Second, several *in house* algorithms based on the PSO theory were developed and tested. The results arising from this approach demonstrated that it allows to reduce drastically the time needed to validate the slurry phase by using the H-D PSO, ML-D PSO and LPSO algorithms, allowing faster screening of different slurry manufacturing conditions, paving the way towards a systematic and fast approach to model slurries in LIB manufacturing. Furthermore, this approach could be adapted to boost the parametrization of other models.

In order to help other researchers to take into account the slurry phase in their models, or to apply the approaches proposed in this work to other case studies, all the results and algorithms generated during this work will be released for free in the ARTISTIC project webpage.^[43]

Experimental Section

LiNi_{1/3}Mn_{1/3}Co_{1/3}O₂ (NMC) was supplied from Umicore. C-ENERGY™ super C65 carbon black was from IMERYs. Solef™ polyvinylidene fluoride (PVdF) was purchased from Solvay and N-methylpyrrolidone (NMP), from BASF. The slurry solid components (NMC, carbon black and PVdF) were premixed with a soft blender overnight. Afterwards, NMP was added until reaching the desired solid content (see Table 1). The total mass of solid components was always 65 g. The mixture was performed in a Dispermat CV3-PLUS high-shear mixer for 2 h at 3000 rpm in a water-bath cooled recipient at 25 °C. The η - $\dot{\gamma}$ curves were acquired by applying a shear rate ramp (0.1–500 Hz) immediately after the slurry preparation by using a rheometer (Kinexus lab+, Malvern Instruments). The slurry density were acquired immediately after the slurry preparation by using a density meter (DMA4500, Anton Paar GmbH).

Computational Methods

All the PSO-based codes were developed in C++ programming language, while the ML algorithms used in this work were developed using Python programming language. Further information concerning the CGMD simulations and structure generation, the FF parametrization, PSO-based algorithms and the ML

optimization approach can be found in the Supporting Information.

Acknowledgements

The authors acknowledge the European Union's Horizon 2020 research and innovation program for the funding support through the European Research Council (grant agreement 772873, "ARTISTIC" project). A.A.F. acknowledges the Institut Universitaire de France for the support. Results have been achieved thanks to the MatriCS HPC platform from Université de Picardie-Jules Verne, particularly using the ARTISTIC project-dedicated computational nodes.

Conflict of Interest

The authors declare no conflict of interest.

Keywords: Li-ion battery · NMC cathode · slurry phase · Particle Swarm Optimization · Machine Learning

- [1] A. Opitz, P. Badami, L. Shen, K. Vignarooban, A. M. Kannan, *Renewable Sustainable Energy Rev.* **2017**, *68*, Part 1, 685.
- [2] G.-L. Zhu, C.-Z. Zhao, J.-Q. Huang, C. He, J. Zhang, S. Chen, L. Xu, H. Yuan, Q. Zhang, *Small* **2019**, *15*, 1805389.
- [3] D. & J.-M. T. Larcher, *Nat. Chem.* **2015**, *7*, 19.
- [4] M. Winter, B. Barnett, K. Xu, *Chem. Rev.* **2018**, *118*, 11433.
- [5] N. Nitta, F. Wu, J. T. Lee, G. Yushin, *Mater. Today* **2015**, *18*, 252.
- [6] G. E. Blomgren, *J. Electrochem. Soc.* **2017**, *164*, A5019.
- [7] D. L. Wood, J. Li, C. Daniel, *J. Power Sources* **2015**, *275*, 234.
- [8] A. A. Franco, A. Rucci, D. Brandell, C. Frayret, M. Gaberscek, P. Jankowski, P. Johansson, *Chem. Rev.* **2019**, *119*, 4569.
- [9] M. Chouchane, A. Rucci, T. Lombardo, A. Ngandjong C, A. A. Franco, *J. Power Sources* **2019**.
- [10] A. Shodiev, E. N. Primo, M. Chouchane, T. Lombardo, A. C. Ngandjong, A. Rucci, A. A. Franco, *J. Power Sources* **2020**, 227871.
- [11] A. Rucci, A. C. Ngandjong, E. N. Primo, M. Maiza, A. A. Franco, *Electrochim. Acta* **2019**, *312*, 168.
- [12] A. C. Ngandjong, A. Rucci, M. Maiza, G. Shukla, J. Vazquez-Arenas, A. A. Franco, *J. Phys. Chem. Lett.* **2017**, *8*, 5966.
- [13] M. E. Ferraro, B. L. Trembacki, V. E. Brunini, D. R. Noble, S. A. Roberts, *J. Electrochem. Soc.* **2020**, *167*, 013543.
- [14] M. Ebner, F. Geldmacher, F. Marone, M. Stampanoni, V. Wood, *Adv. Energy Mater.* **2013**, *3*, 845.
- [15] P. Pietsch, V. Wood, *Annu. Rev. Mater. Res.* **2017**, *47*, 451.
- [16] P. Pietsch, M. Ebner, F. Marone, M. Stampanoni, V. Wood, *Sustain. Energy Fuels* **2018**, *2*, 598.
- [17] P. P. Mukherjee, D. R. Noble, M. E. Ferraro, S. A. Roberts, B. L. Trembacki, A. N. Mistry, *J. Electrochem. Soc.* **2018**, *165*, E725.
- [18] B. L. Trembacki, D. R. Noble, V. E. Brunini, M. E. Ferraro, S. A. Roberts, *J. Electrochem. Soc.* **2017**, *164*, E3613.
- [19] <https://www.u-picardie.fr/erc-artistic/>.
- [20] K. B. Hatzell, M. B. Dixit, S. A. Berlinger, A. Z. Weber, *J. Mater. Chem. A* **2017**, *5*, 20527.
- [21] W. Bauer, D. Nötzel, *Ceram. Int.* **2014**, *40*, 4591.
- [22] G. W. Lee, J. H. Ryu, W. Han, K. H. Ahn, S. M. Oh, *J. Power Sources* **2010**, *195*, 6049.
- [23] R. P. Cunha, T. Lombardo, E. N. Primo, A. A. Franco, *Batter. Supercaps* **2020**, *3*, 60.
- [24] M. M. Forouzan, C. W. Chao, D. Bustamante, B. A. Mazzeo, D. R. Wheeler, *J. Power Sources* **2016**, *312*, 172.
- [25] <https://lammps.sandia.gov/>.
- [26] E. J. Maginn, R. A. Messerly, D. J. Carlson, D. R. Roe, J. R. Elliott, *Living J. Comput. Mol. Sci.* **2018**, *1*, 6324.

- [27] T.-J. Liu, C. Tiu, L.-C. Chen, D. Liu, In *Printed Batteries*; Lancers-Méndez, S.; Costa, C. M., Eds.; John Wiley & Sons Ltd., 2018; pp. 63–79.
- [28] J. Kennedy, R. C. Eberhart, *Proc. IEEE Int. Conf. Syst. Man Cybern.* **1997**, *5*, 4104.
- [29] R. C. Eberhart, Y. Shi, *Proc. IEEE Conf. Evol. Comput. IEC* **2001**, *1*, 81.
- [30] N. Chakraborti, R. Jayakanth, S. Das, E. D. Çalişir, Ş. Erkoç, *J. Phase Equilib. Diffus.* **2007**, *28*, 140.
- [31] M. R. Bonyadi, Z. Michalewicz, *Evol. Comput.* **2017**, *25*, 1.
- [32] S. Janson, D. Merkle, M. Middendorf, *Appl. Soft Comput. J.* **2008**, *8*, 666.
- [33] Y. Wang, J. Lv, L. Zhu, Y. Ma, *Phys. Rev. B* **2010**, *82*, 1.
- [34] Y. Wang, J. Lv, L. Zhu, Y. Ma, *Comput. Phys. Commun.* **2012**, *183*, 2063.
- [35] Y. Wang, J. Lv, L. Zhu, S. Lu, K. Yin, Q. Li, H. Wang, L. Zhang, Y. Ma, *J. Phys. Condens. Matter* **2015**, *27*.
- [36] Q. Tong, L. Xue, J. Lv, Y. Wang, Y. Ma, *Faraday Discuss.* **2018**, *211*, 31.
- [37] H. Wang, Y. Wang, J. Lv, Q. Li, L. Zhang, Y. Ma, *Comput. Mater. Sci.* **2016**, *112*, 406.
- [38] <https://fr.mathworks.com/matlabcentral/fileexchange/57286-video-tutorial-of-particle-swarm-optimization-pso-in-matlab> (accessed in december 2018).
- [39] Z. Liu, P. P. Mukherjee, *J. Electrochem. Soc.* **2014**, *161*, E3248.
- [40] C. Sangrós Giménez, C. Schilde, L. Froböse, S. Ivanov, A. Kwade, *Energy Technol.* **2019**.
- [41] C. Sangrós Giménez, B. Finke, C. Nowak, C. Schilde, A. Kwade, *Adv. Powder Technol.* **2018**, *29*, 2312.
- [42] C. Sangrós Giménez, B. Finke, C. Schilde, L. Froböse, A. Kwade, *Powder Technol.* **2019**, *349*, 1.
- [43] <https://www.u-picardie.fr/erc-artistic/computational-portal/>.

Manuscript received: March 5, 2020
Revised manuscript received: April 8, 2020
Accepted manuscript online: April 9, 2020
Version of record online: April 22, 2020