**Abstract.** We introduce the open-source library **ParScale** for the modeling of intra-particle transport processes in non-isothermal reactive fluid-particle flows. The underlying equations, the code architecture, as well as the coupling strategy to the widely-used DEM solver **LIGGGHTS®** is presented. A set of verification cases, embedded into an automated test harness, is presented that proofs the functionality of **ParScale**. To demonstrate the capabilities of **ParScale**, we perform simulations of a non-isothermal granular shear flow including heat transfer to the surrounding fluid. We present results for the conductive heat flux through the particle bed for a wide range of dimensionless cooling rates and particle volume fractions. Our data suggests that intra-particle temperature gradients need to be considered for an accurate prediction of the conductive flux in case of (i) a dense particle bed and (ii) for large cooling rates characterized by a critical Biot number of ca. $\text{Bi}_{\text{crit}} \approx 0.1$. 

**Key words:** Granular Materials, DEM, **LIGGGHTS®**, ParScale, Resolved Intra Particle Profiles, Heat and Mass Transfer, Sheared Bed
1 Introduction

Over the past ten years the coupling of the Discrete Element Method (DEM) and Computational Fluid Dynamics (CFD) has been widely adopted by academia and industry to predict fluid-particle flows [1]. Recently, the trend towards CFD-DEM has been fueled by the introduction of open-source toolboxes [2]. These tools are able to model momentum and thermal energy transport of the particles and the fluid with high computational efficiency, hence provide a detailed understanding of granular flow behavior. However, these tools typically do not take intra-particle transport phenomena into account, and hence are unable to model intra-particle processes, e.g., heterogeneous reactions and diffusion inside a porous particle. Unfortunately, in these reactive particulate systems intra-particle processes often play a central role, and hence may dictate overall reactor performance. Thus, spatially-resolved particle properties (e.g., the local gas concentration in the pores of the particle) need to be resolved to account for all relevant phenomena in the system [3].

ParScale, a newly developed open-source library implemented in a C/C++ environment and publicly available through www.github.com [4], closes this gap. At the current development state, ParScale contains a plurality of models that aim on predicting heat and mass transfer, as well as homogeneous and heterogeneous reactions inside flowing porous particles. Also, it is possible to account for a single or multiple-reactions, i.e., a whole reaction network. Due to a modular class-based structure, and the integration into an automated test harness, easy extendibility and a high software quality is ensured. Our contribution outlines the governing equations for modeling various intra-particle phenomena in Section 2. In section 3 a number of verification cases is presented in order to demonstrate and verify the functionality of ParScale. Section 4 demonstrates the coupling to LIGGGHTS® and evaluates the need to account for intra-particle temperature gradients in non-isothermal granular shear flows cooled by a surrounding fluid.

2 Simulation method and parallel coupling strategy

The key purpose of ParScale is to predict intra-particle target properties (e.g., the temperature) as a function of time and space in a spherical particle and a fixed grid consisting of equidistant grid points. The governing equations for the relevant transport phenomena within each particle are outlined in the next section.

2.1 Transport within a particle

To illustrate a typical transport equation to be solved by ParScale within a single particle, the Fourier equation in spherical coordinates with $\lambda_{eff} = const.$ is considered:

$$\rho c_p \frac{\partial T}{\partial t} = \text{div}(\lambda_{eff} \text{grad} T) + s_T$$ (1)

where $\rho$ is the density, $c_p$ is the heat capacity, $T$ represents the target property profile, $t$ is the time, $\lambda_{eff}$ is an effective conductivity (e.g., for heat), and $s_T$ is a volume-specific
source term (e.g., for thermal energy). By introducing the thermal diffusivity

\[ a = \frac{\lambda_{\text{eff}}}{\rho c_p} \]  

(2)

and for constant thermal conductivity \( \lambda_{\text{eff}} \), Eqn. (1) can be re-written as

\[ \frac{\partial T}{\partial t} = a \cdot \left( \frac{\partial^2 T}{\partial r^2} + \frac{2}{r} \frac{\partial T}{\partial r} \right) + \frac{s_T}{\rho c_p} \]  

(3)

where \( r \) is the radial coordinate.

This partial differential equation (PDE) can be discretized in space, e.g., using a central-differencing scheme, in the spirit of the so-called method of lines. The resulting system of ordinary differential equations (ODEs) needs to be solved using a robust integration approach, since the source \( s_T \) might lead to a stiff behavior of the PDE. Specifically, we have chosen the flexible multi-step, variable-order solver CVODE, which uses a (modified) Newton-iteration approach to robustly integrate stiff systems of ODEs. CVODE contains a plurality of direct and iterative linear solvers for solving the resulting matrix-algebra problems, and is actively maintained by the Lawrence Livermore National Laboratory (LLNL, U.S.A) as part of the SUNDIALS package [5]. ParScale inherits the flexibility of CVODE, and hence can handle quickly changing environmental conditions, or fast, strongly exothermal reactions in porous particles.

For the second verification example (see Section 3.2), we will consider a single heterogeneous reaction in a porous particle. The corresponding transport equations are presented in the next section.

2.2 Chemistry model

The following equations model the mole-based reactive species balance equations in a particle with constant porosity \( \epsilon \). We have adopted the notation of Noorman et al. [6], and from which we have extracted typical system parameters for a relevant application.

Here we focus on a single chemical reaction, which is considered to be irreversible and involves a solid species \( A \), as well as a gas species \( B \) educt that forms a solid product \( C \) and a gaseous product \( D \):

\[ aA(s) + bB(g) \rightarrow cC(s) + dD(g) \]  

(4)

Relevant real-world examples following this scheme are (i) the reduction of iron oxides by \( H_2 \), (ii) the oxidation of ZnS and FeS with \( O_2 \) to metal oxides (i.e., ZnO, Fe₂O₃), or (iii) the combustion of coal with a high ash content. The mole balance equation for each species in the porous particle is written as:

\[ \frac{\partial \epsilon c_i}{\partial t} = -\frac{1}{r^2} \frac{\partial}{\partial r} (r^2 N_i) + s_i \text{ with } i = B, D \]  

(5)
Here $\varepsilon$ is the phase fraction, $c_i$ is the gas concentration (in $kMol$ per $m^3$ gas volume), $N_i$ the (convective and diffusive) flux, and $s_i$ is the chemical source term. A typical model for $s_i$ is

$$s_i = \sum_{j=1}^{NR} \nu_{ij} \cdot \prod_{k} c_k^{n_{k,j}}$$

(6)

where $NR$ is the number of reactions, $NG$ is the number of gas components, $\nu_{ij}$ is the stoichiometric coefficient of species $i$ in reaction $j$, $NS$ is the number of solid species, and $n_{k,j}$ is the reaction exponent. The mole balance equations for the solid phase are similar, however, exclude the flux and are based on molar concentrations in $kMol$ per $m^3$ total volume:

$$\frac{\partial c_i}{\partial t} = s_i \text{ with } i = A, C$$

(7)

For spatial discretizing a second order central differencing scheme is used. For further information about the underlying equations, available models and discretization schemes the interested reader may refer to the online documentation in the public repository of ParScale [4]. The next section briefly outlines the run modes of ParScale including the coupling to the open-source DEM-based solver LIGGGHTS®.

2.3 Parallel coupling strategy

Besides a stand-alone mode the current development state of ParScale provides coupling capabilities to LIGGGHTS® and selected solvers of CFDEMcoupling in parallel. The key idea is that ParScale acts as a slave to the master (i.e., LIGGGHTS®), and can exchange its data containers between individual processes as requested by the master. In this paper we will only focus on the coupling to LIGGGHTS®. The coupling to CFDEMcoupling is handled via LIGGGHTS® data structures, and hence is in fact unproblematic. Figure 1 illustrates the underlying coupling algorithm.

At every timestep $t_n$ LIGGGHTS® advances the particle position and velocity (and other integral particle quantities if desired) in the simulation domain. After this computation is finished, the coupling is realized by updating the particle surface temperature ($T_{surface}$) directly, or (alternatively) a heat transfer coefficient $\alpha$ together with the fluid temperature in the vicinity of the particle. In addition, a conductive heat flux due to particle-particle collisions ($\dot{q}_{cond}$) can be imposed as well. ParScale initializes from the last time step $t_{n-1}$, and calculates all internal property fields according to the imposed boundary conditions. Additional coupling options are available, e.g., LIGGGHTS® is able to push a particle-unique environment temperature (i.e., fluid temperature) to ParScale. This enables ParScale to react due to changes in the environment temperature, e.g., if the particle enters a region if a fixed temperature. Furthermore, the coupling provides the option to reset the value of the target property to a certain value. Due to the automatic sub-time
stepping of CVODE, the internal ParScale timestep does not necessarily have to correspond to the DEM timestep from LIGGGHTS®. After ParScale completed its calculation of the surface temperatures of all particles, these temperatures are updated and taken into account by LIGGGHTS® at timestep $t_{n+1}$ when computing conductive fluxes. Furthermore, source terms due to reactions, the core and the volume-averaged temperature and surface fluxes handed over to LIGGGHTS®.

Next, two verification cases, i.e., transient cooling of a sphere and a heterogeneous reaction, is presented.

3 Verification cases

3.1 Cooled sphere

The first verification case considers a classical situation in which a spherical particle (initially having the uniform temperature $T_0$) is convectively cooled by an ambient fluid with temperature $T_{enviro}$. Table 1 summarizes the parameters of this case. Figure 2 illustrates the comparison between the numerical solution by ParScale and the analytical solution provided by [7] for a number of time coordinates.

As expected, excellent agreement (i.e., an average error of $10^{-6}$, and a maximum error of $10^{-5}$) between analytical and numerical solution can be found.
Table 1: Parameters for the verification case ‘cooling of a sphere’.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>( c_p )</td>
<td>300</td>
<td>J m(^{-3}) K(^{-1})</td>
</tr>
<tr>
<td>( \rho )</td>
<td>1000</td>
<td>kg m(^{-3})</td>
</tr>
<tr>
<td>( \lambda_s )</td>
<td>1</td>
<td>W m(^{-1}) K(^{-1})</td>
</tr>
<tr>
<td>( \alpha_p )</td>
<td>100</td>
<td>W m(^{-2}) K(^{-1})</td>
</tr>
<tr>
<td>( r_p )</td>
<td>5 \cdot 10(^{-3})</td>
<td>m</td>
</tr>
<tr>
<td>( T_0 )</td>
<td>800</td>
<td>K</td>
</tr>
<tr>
<td>( T_{enviro} )</td>
<td>300</td>
<td>K</td>
</tr>
<tr>
<td>( t_1, t_2, t_3, t_4 )</td>
<td>2, 5, 8, 10</td>
<td>sec</td>
</tr>
</tbody>
</table>

Figure 2: Numerical (symbols) and analytical results (lines) for the temperature distribution in a convectively cooled sphere at \( t_1 = 2 \) s, \( t_2 = 5 \) s, \( t_3 = 8 \) s, and \( t_4 = 10 \) s.

### 3.2 Heterogeneous reaction

This verification case considers a single reaction, and follows the analytical solution provided by Wen [8] for a relative reaction speed (characterized by the Thiele Modulus) of \( \approx 3.16 \). All parameters are chosen following the copper oxidation case considered in Noorman et al. [6]. The three basic assumptions are (i) an isothermal particle, (ii) a reaction rate that is first-order with respect to the gas-phase species, (iii) and a reaction of: \( 2 CO + O_2 \rightarrow 2 CuO \). Figure 3 shows a comparison of the solid and fluid concentration inside the particle for two characteristic times.

Figure 3 (a) shows an excellent agreement for the early stage of the reaction. The small, but noticeable, differences in the later stage (see Figure 3 (b)) are due to the pseudo-steady state assumption that needs to be adopted when deriving the analytical solution provided...
Figure 3: Solid and gas-phase concentration profiles inside a porous copper particle for the early stage (a) and the late stage (b) of a heterogeneous reaction (lines: analytical solution, symbols: predictions by ParScale).

by Wen [8]. Figure 4 shows the overall conversion, again indicating only minor differences that can be explained by the shortcomings of the analytical solution. The comparison of the analytical and numerical solution shows good agreement, and the mean difference is below 2%. We note here that previous work [6] came to a similar conclusion.

Figure 4: Conversion characteristics during a typical oxidation cycle of a porous copper particle (line: analytical solution, dots: prediction by ParScale).
4 Simple shear flow

We now investigate the influence of ParScale under well-controlled flow conditions. Therefore, particles are placed in a cubic periodic box (H/d_p = 15) at various particle volume fractions \( \phi_p = (0.3 - 0.64) \). LeesEdwards boundary conditions are applied to drive a homogeneous shear flow, which is typically used in studies of granular rheology [9]. In the current contribution, the shear gradient is pointing in the y-direction, and we analyze all quantities of interest (e.g., the conductive flux) only in this direction. Along with the shear gradient, a temperature gradient is applied. We determine the Biot (Eqn. 8) and the Peclet (Eqn. 9) number as the two main non-dimensional influence parameters as already mentioned by [10]. With the usage of ParScale at low Biot numbers, we expect that our results agree with previous work [10]. However, for higher Biot numbers the transferred flux to the ambient fluid is much larger than that sustained by conduction inside the particle. Consequently, the influence on the heat transfer rate on the particles’ shell temperature, and hence the conductive flux becomes important. Therefore, we expect that intra particle temperature profiles should be considered above some critical Biot number \( Bi_{crit} \).

\[
Bi_t = \frac{\alpha d_p}{\lambda_p} \quad (8)
\]

where \( \alpha \) is the heat transfer coefficient that characterized the rate of cooling by the ambient fluid.

\[
P_e = \frac{(d_p/2)^2}{\lambda_p / \rho_c c_p} \cdot \dot{\gamma} \quad (9)
\]

where \( \dot{\gamma} \) is the shear rate. The conductive reference flux is expressed as [9]

\[
\dot{q}_{cond,ref} = -\lambda_p (\partial_y T)_{middle}. \quad (10)
\]

where \( \dot{q}_{cond,ref} \) is the reference conductive heat flux and \( y_{middle} \) is the length of the region over which the temperature gradient is applied. Table 2 shows the main non-dimensional parameters of the sheared bed simulation. All other parameters, e.g., the particle stiffness, the coefficient of restitution and time step are in agreement with our earlier simulations [10].

| Table 2: Dimensionless properties of the sheared bed particle case. |
|-------------|------------------|
| \( \phi_p \) | 0.3...0.64       |
| \( Bi_t \)  | \( 10^{-6}...10 \) |
| \( Pe \)    | 0.25             |
It can be seen in Figure 5 that the influence of the Biot number on the overall conductive flux needs to be considered above a certain value for the Biot number, which depends on $\phi_p$. Even at low Biot number regimes (i.e., $Bi_t \approx 10^{-3}$) the conductive flux is underpredicted by up to 10%. For the highest particle volume fractions considered, for which conductive fluxes are most relevant since they are comparable to the particle-convective flux, the critical Biot number is $Bi_{crit} \approx 10^{-2}$. Above this Biot number intra-particle profiles should be taken into account for the moderately fast sheared particle bed that we considered. It is also shown that the influence of the Biot number on the conductive heat flux is becoming more important in case the particle volume fraction is decreasing. Thus, the slope of the curve indicating the relationship between $Bi$ and $q_{cond}$ is becoming larger with decreasing $\phi_p$ for the high $Bi$ regime. However, in the rather dilute flow at which these extreme dependency is observed, the conductive flux is negligibly small compared to the particle-convective flux.

5 Conclusions

We presented a novel open-source simulation tool ParScale which is published under LGPL licence and can be linked to any particle-based solver. We outlined the coupling to the open-source DEM solver LIGGGHTS® and demonstrated the usage of ParScale with selected verification cases. A good agreement is found between numerical results produced
by ParScale and analytical solutions available in literature. We extended the analysis of
the well-known sheared particle bed by taking intra-particle temperature gradients into
account. A key result is that even at comparably low Biot numbers the intra-particle
property profiles have a substantial influence on the conductive flux. The physical reason
is that the particle surface temperature is lower than the particle-average temperature
in case cooling by the ambient fluid is taken into account. This leads to smaller surface
temperature differences in the event of a particle-particle collisions. Thus, the transferred
heat flux to the environment should be considered when predicting the particle-particle
conductive fluxes. This is especially true for high particle concentrations and fast cooling
conditions, since the conductive flux for low particle concentrations is anyhow very low.
The current study was limited to selected particle volume fractions and Peclet numbers.
Future work will consider wider ranges of these parameters.

6 Acknowledgement

The authors acknowledge support by the European Commission through FP7 Grant
agreement no. 604656 (NanoSim). T.F. and S.R. acknowledge support from "NAWI
Graz" by providing access to dcluster.tugraz.at. LIGGGHTS® is a registered trademark of
DCS Computing GmbH.
References


