

# Study of particle rotation effect in gas-solid flows using direct numerical simulation with a lattice Boltzmann method

Qiang Zhou, L.-S. Fan

The Ohio State University

Kyung C Kwon

Tuskegee University

# Outline

- Background: Lift force due to particle rotation
- Background: Immersed boundary-Lattice Boltzmann method, from first order to second order
- Drag force and lift force in simple cubic arrays of spheres
- Drag force and lift force in random arrays of spheres
- Open question, how to apply lift force in practical simulations.

# Background: Lift force due to particle rotation

- Gas-solid multiphase flows are prevalent in many fossil fuel processes such as gasification and combustion.
- Advanced computational technique known as the computational fluid dynamics (CFD) has been recognized as an emerging tool that is able to reduce the cost in the design and scale-up of the multiphase reactors involved in those processes, and has been applied in typical equipment such as fluidized bed gasifiers and chemical looping combustion reactors.
- The capability of the CFD in correctly predicting multiphase flow dynamics relies heavily on accuracy of sub-models that account for **particle-fluid interactions** and particle-particle interactions.
- The overall objective of our proposed research is to improve fundamental understanding of such interactions, and to formulate **a new drag model and a new lift force model that reflects particle rotation effects.**

# Background: Lift force due to particle rotation

- Previously, it was believed that the **lift force (Magnus force)** caused by particle rotation is **insignificant** compared to the drag force.

**Rotational Reynolds number:**  $Re_r = \omega D^2 / \nu$

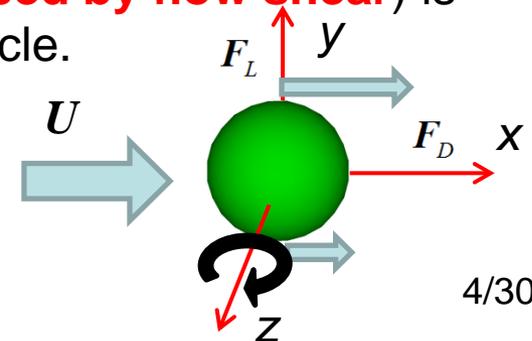
- Rubinow & Keller (1961), Stokes flow:  $F_D = 6\pi a \mu U [1 + \frac{3}{8} Re + o(Re)]$

lift-to-drag ratio  $F_L / F_D = Re_r / 24$   $F_L = -\pi a^3 \rho \omega \times U [1 + O(Re)]$

- Saffman (1965) mentioned that the lift force due to the rotation is less by an order of magnitude than that due to the shear in the flow, **unless** the rotation speed is very much greater than the rate of shear.

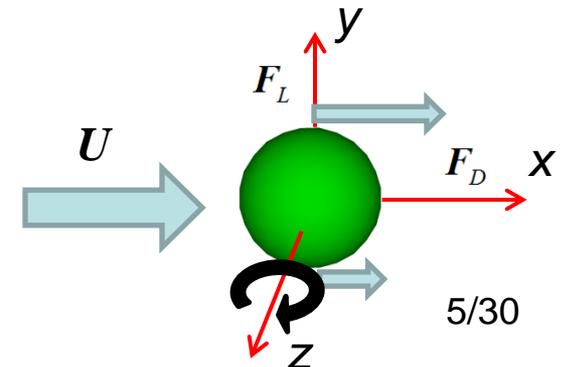
$$F_L = 6.46 \nu \rho a^2 U (|\alpha|/\nu)^{1/2} - \frac{11}{8} \rho U \alpha a^3 + \pi \rho U \omega a^3$$

- Bagchi & Balachandar (2002) found that, even in finite Reynolds number regimes, the lift due to **free particle rotation (induced by flow shear)** is less significant than shear-induced lift on the particle.



# Background: Lift force due to particle rotation

- White and Schulz (1977) obtained the particle trajectories by high-speed motion pictures (2000 frames/s) of the **saltating spherical glass microbeads** of diameter 0.35–0.71 mm, and found that
  - ✓ the typical spin rates of sand particles are the order of several hundred revolutions per second ( **$Re_r \sim 50$** );
  - ✓ the Magnus lifting force can produce good computational trajectory and increase the height of sand motion by 50%;
  - ✓ The Saffman force can be safely neglected without noticeably affecting particle trajectories;
  - ✓ They pointed out that the **high speed spinning rates are generally obtained by collisions with the sand surface**
- White (1982) and Zou et al. (2007) gave similar reports.



# Background: Lift force due to particle rotation

- **Recent high-speed imaging show that**
- Wu et al.(2008) measured the particle rotation speed in a cold pilot-scale Circulating Fluidized Bed (CFB) riser. Particle diameter: 0.5 mm.

Particle rotation speed:

Average: 300 rev/s                       $Re_r=30$

Maximum: 2000 rev/s.                       $Re_r=210$

- Shaffer et al. (2009) reported the rotation rate of particle in a riser flow. particle size is round 0.5~0.75mm

Particle rotation speed:

Average: 22700 rpm                       $Re_r=90$

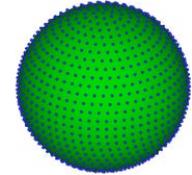
Maximum: 90400 rpm                       $Re_r=350$

- **The importance of the lift force needs to be reevaluated.**

# Outline

- Background: Lift force due to particle rotation
- Background: Immersed boundary-Lattice Boltzmann method, from first order to second order
- Drag force and lift force in simple cubic arrays of spheres
- Drag force and lift force in random arrays of spheres
- Open question, how to apply lift force in practical simulations.

# Background: Immersed boundary-Lattice Boltzmann method, from first order to second order



## Immersed Boundary Method (IBM)

### Advantage:

- The non-slip/non-penetration (ns/np) is easily imposed by adding additional force to the flow in the vicinity of the particle surface
- Does not need regriding when particles are moving

### Disadvantage:

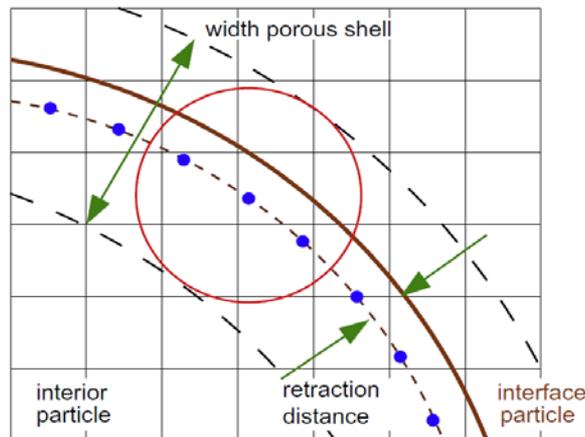
- The approximation of ns/np is hard to be exactly imposed.
- Traditional IBM only **yields first-order accuracy**.

# Background: Immersed boundary-Lattice Boltzmann method, from first order to second order

## Recent improvement by Breugem (2012)

- Multidirect forcing scheme to reduce the ns/np error (Luo et al. 2007)
- **A slight retraction** (Hofler and Schwarzer 2000) of the Lagrangian grid from the surface towards the interior of the particles is used to enhance the accuracy of IBM.

**Breugem (2012) demonstrated that the improved IBM coupled with traditional incompressible NS-solver gives a second order of grid convergence**



**Fig. 3.** Illustration of the porous shell covering a solid particle. The dots indicate the position of Lagrangian grid points, which are retracted from the actual interface (the solid line) with a fraction of the Eulerian grid spacing (about  $0.3\Delta x$  in this case). The circle depicts the range of action of the regularized Dirac delta function.

# Background: Immersed boundary-Lattice Boltzmann method, from first order to second order



Our contribution: embed the improved IBM into LBM

- Using **Runge-Kutta (RK) schemes** to advance the position, the linear momentum and angular momentum of the particle. RK requires fluid field information at fractional time steps.
- In the framework of **LBM**

$$m_j(\mathbf{x} + \mathbf{e}_j \Delta t, t + \Delta t) - m_j(\mathbf{x}, t) = -\frac{1}{\tau} [m_j(\mathbf{x}, t) - m_j^{(eq)}(\mathbf{x}, t)] + \Delta t F_{f,j}(\mathbf{f}),$$

one can not directly get the flow information in the fractional time step between  $n\Delta t$  and  $(n+1)\Delta t$ . We get the flow **information by simple extrapolation**:

$$\mathbf{u}^{n+\alpha} = \mathbf{u}^n + (\text{rhs}^n + \mathbf{f}^*)\alpha \Delta t / \rho^n$$

$$\rho^{n+\alpha} = \rho^n - \nabla \cdot (\rho^n \mathbf{u}^{n+\alpha})\alpha \Delta t.$$

- These two equations are implemented locally in the cubic computational domains that circumscribe particles.

# Background: Immersed boundary-Lattice Boltzmann method, from first order to second order



Procedures for the fluid phase:

$$\begin{aligned}
 & \mathbf{f} = -\nabla p_e, \\
 & \text{do } q = 1, i \quad \text{Field extrapolation} \\
 & \mathbf{f}^{q-1, N_s} = \begin{cases} 0, & q = 1 \\ \sum_{k=1}^{q-1} \beta_{q-1, k} \mathbf{f}^{k, N_s} / \sum_{k=1}^{q-1} \beta_{q-1, k}, & q > 1 \end{cases} \\
 & \mathbf{u}^* = \mathbf{u}^0 + (\text{rhs}^0 + \mathbf{f}^{q-1, N_s}) \alpha_q \Delta t / \rho^0, \\
 & \rho^* = \rho^0 - \nabla \cdot (\rho^0 \mathbf{u}^*) \alpha_q \Delta t, \\
 & \tilde{\mathbf{u}} = (\rho \mathbf{u})^* + \text{rhs}^* \Delta t, \\
 & \tilde{U}_l = \sum \tilde{\mathbf{u}} \delta_d(\mathbf{x} - \mathbf{X}_l^{q-1}) \Delta x \Delta y \Delta z \quad \forall l, \\
 & \tilde{U}_{p,l}^{q-1} = \rho^* \mathbf{U}_{p,l}^{q-1} \quad \forall l, \\
 & \mathbf{F}_l^{q,0} = \frac{\tilde{U}_{p,l}^{q-1} - \tilde{U}_l}{\Delta t} \quad \forall l, \\
 & \mathbf{f}^{q,0} = \sum \mathbf{F}_l^{q,0} \delta_d(\mathbf{x} - \mathbf{X}_l^{q-1}) \Delta V_l, \\
 & \tilde{\mathbf{u}}^0 = \tilde{\mathbf{u}} + \Delta t \mathbf{f}^{q,0},
 \end{aligned}$$

Direct forcing scheme (Uhlmann 2005)

Multidirect forcing scheme (Luo et al. 2007)

$$\begin{aligned}
 & \text{do } s = 1, N_s \\
 & \tilde{U}_l^{s-1} = \sum \tilde{\mathbf{u}}^{s-1} \delta_d(\mathbf{x} - \mathbf{X}_l^{q-1}) \Delta x \Delta y \Delta z \quad \forall l, \\
 & \mathbf{F}_l^{q,s} = \mathbf{F}_l^{q,s-1} + \frac{\tilde{U}_{p,l}^{q-1} - \tilde{U}_l^{s-1}}{\Delta t} \quad \forall l, \\
 & \mathbf{f}^{q,s} = \sum \mathbf{F}_l^{q,s} \delta_d(\mathbf{x} - \mathbf{X}_l^{q-1}) \Delta V_l, \\
 & \tilde{\mathbf{u}}^s = \tilde{\mathbf{u}} + \Delta t \mathbf{f}^{q,s}, \\
 & \text{end do} \\
 & \mathbf{f} = \mathbf{f} + \beta_{iq} \mathbf{f}^{q, N_s}, \\
 & \text{end do}
 \end{aligned}$$

Update particle motion

$$\begin{aligned}
 m_j(\mathbf{x} + \mathbf{e}_j \Delta t, t + \Delta t) - m_j(\mathbf{x}, t) &= -\frac{1}{\tau} [m_j(\mathbf{x}, t) - m_j^{(eq)}(\mathbf{x}, t)] + \Delta t F_{f,j}(\mathbf{f}), \\
 \rho^n &\rightarrow \rho^{n+1}, \quad \mathbf{u}^n \rightarrow \mathbf{u}^{n+1}, \quad p^n \rightarrow p^{n+1},
 \end{aligned}$$

# Background: Immersed boundary-Lattice Boltzmann method, from first order to second order



Update particle motion

$$\mathbf{u}^{**} = \mathbf{u}^0 + (\mathbf{r}hs^0 + \mathbf{f}^{q-1, N_s})\gamma_q \Delta t / \rho^0,$$

$$\rho^{**} = \rho^0 - \nabla \cdot (\rho^0 \mathbf{u}^{**})\gamma_q \Delta t,$$

$$\mathbf{u}^* = \tilde{\mathbf{u}}^{N_s} / \rho^*,$$

$$\rho^* = \rho^* - \nabla \cdot (\rho^* \mathbf{u}^*) \Delta t,$$

$$\mathbf{F}^q = - \sum_l \mathbf{F}_l^{q, N_s} \Delta V_l + \left(1 - \frac{\rho_f}{\rho_p}\right) M_p \mathbf{g} + \left( \int_{V_p} (\rho \mathbf{u})^* dV^{q-1} - \int_{V_p} (\rho \mathbf{u})^{**} dV^{\gamma_q-1} \right) / [(\alpha_q + 1 - \gamma_q) \Delta t],$$

$$\mathbf{T}^q = - \sum_l \mathbf{r}_l^{q-1} \times \mathbf{F}_l^{q, N_s} \Delta V_l + \left( \int_{V_p} (\mathbf{r} dV)^{q-1} \times (\rho \mathbf{u})^* - \int_{V_p} (\mathbf{r} dV)^{\gamma_q-1} \times (\rho \mathbf{u})^{**} \right) / [(\alpha_q + 1 - \gamma_q) \Delta t],$$

$$\mathbf{x}_c^q = \mathbf{x}_c^0 + \sum_{k=1}^q \beta_{qk} \mathbf{u}_c^{k-1} \Delta t,$$

$$\mathbf{p}^q = \mathbf{p}^0 + \sum_{k=1}^q \beta_{qk} \mathbf{F}^k \Delta t,$$

$$\mathbf{L}^q = \mathbf{L}^0 + \sum_{k=1}^q \beta_{qk} \mathbf{T}^k \Delta t,$$

$$\mathbf{Q}^q = \mathbf{Q}^0 + \sum_{k=1}^q \beta_{qk} \omega_c^{k-1} \mathbf{Q}^{k-1} \Delta t / 2,$$

$$\mathbf{u}_c^q = \mathbf{p}^q / M_p,$$

$$\omega_c^q = \mathbf{I}_p^{-1} \mathbf{L}^q,$$

$$\mathbf{X}_l^q = \mathbf{x}_c^q + \mathbf{Q}^q \mathbf{X}_l^{\text{initial}} \widetilde{\mathbf{Q}}^q \quad \forall l,$$

$$\mathbf{U}_{p,l}^q = \mathbf{u}_c^q + \omega_c^q \times (\mathbf{X}_l^q - \mathbf{x}_c^q) \quad \forall l,$$

Direct account of the inertia of the fluid contained within the particles (Kempe et al. 2009)

$$\mathbf{F}^q = - \sum_l \mathbf{F}_l^{q, N_s} \Delta V_l + \left(1 - \frac{\rho_f}{\rho_p}\right) M_p \mathbf{g} + \frac{d}{dt} \left( \int_{V_p} \rho \mathbf{u} dV \right)$$

$$\mathbf{T}^q = - \sum_l \mathbf{r}_l^{q-1} \times \mathbf{F}_l^{q, N_s} \Delta V_l + \frac{d}{dt} \left( \int_{V_p} \mathbf{r} \times (\rho \mathbf{u}) dV \right)$$

It is demonstrated, through extensive benchmark problems, that the IB-LBM has the capacity to resolve the translational and rotational motion of particles with **the second-order accuracy**.

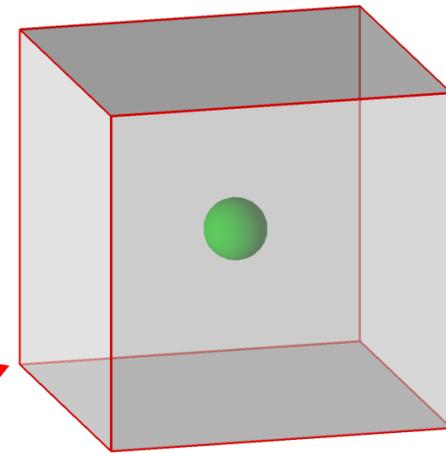
# Outline

- Background: high-speed imaging of particle rotation
- Background: Immersed boundary-Lattice Boltzmann method, from first order to second order
- **Drag force and lift force in simple cubic arrays of spheres**
- Drag force and lift force in random arrays of spheres
- Open question, how to apply lift force in practical simulations.

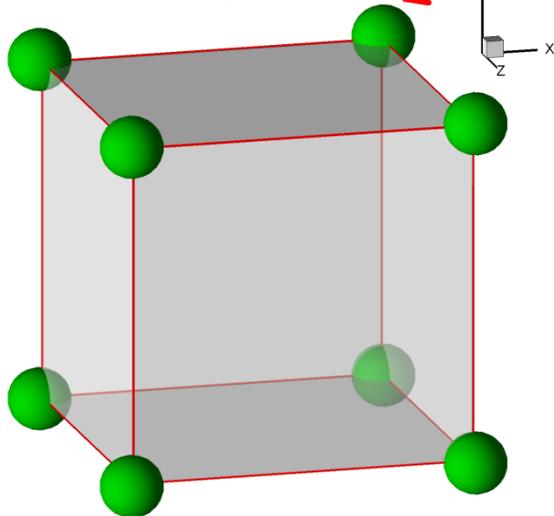
# Drag force and lift force in simple cubic arrays of spheres

## Three ordered configurations

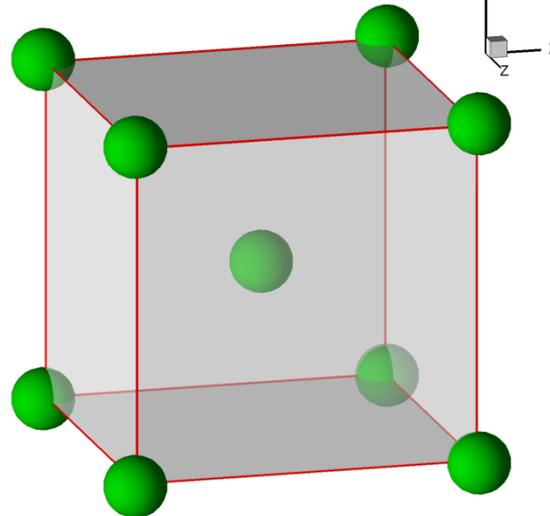
- Theoretical results of the drag force can be applied to these three configurations. (Zick & Homay 1982)



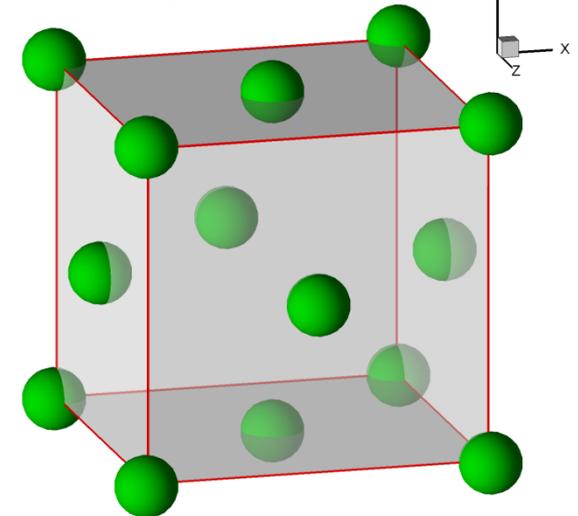
Simple-cubic packing  
(SC)



Body-centred cubic packing  
(BCC)



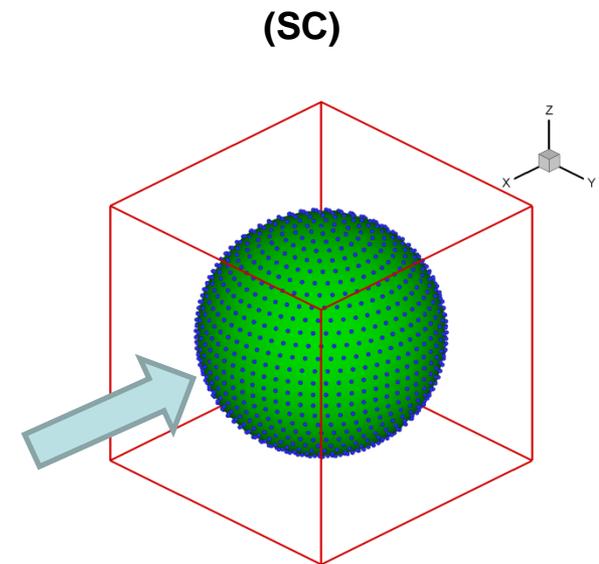
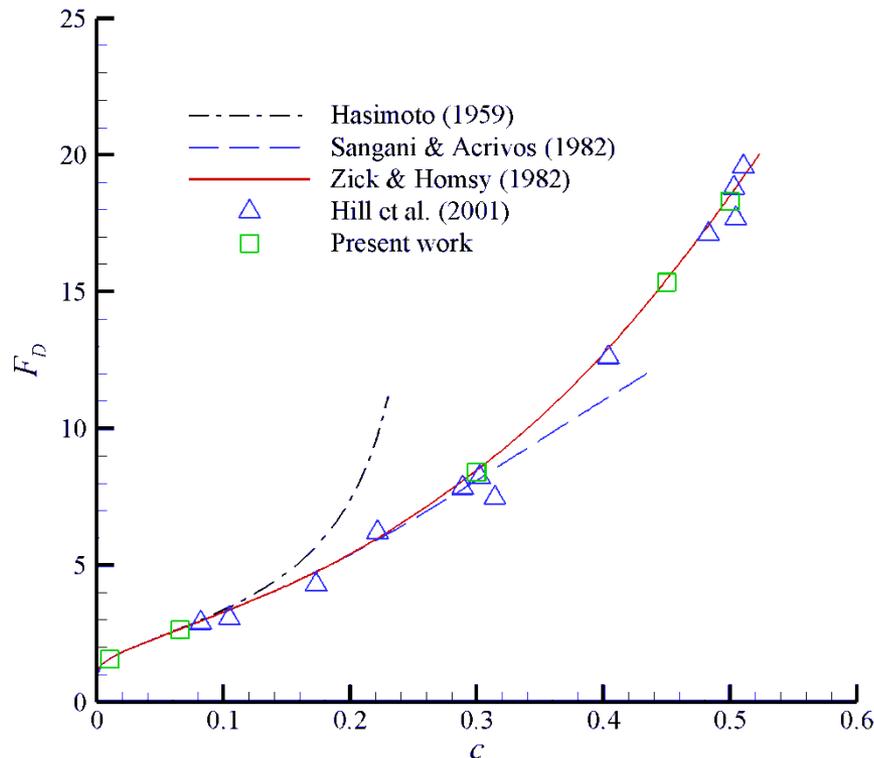
Face-centered cubic packing  
(FCC)



# Drag force and lift force in simple cubic arrays of spheres

## Validation of our simulation

- The drag force at various solid volume fraction ( $c$ ) agree well with theoretical results.
- Position fixed, particle Reynolds number ( $Re_p$ ) < 0.2 (effectively zero), flow driven by uniform pressure gradient.

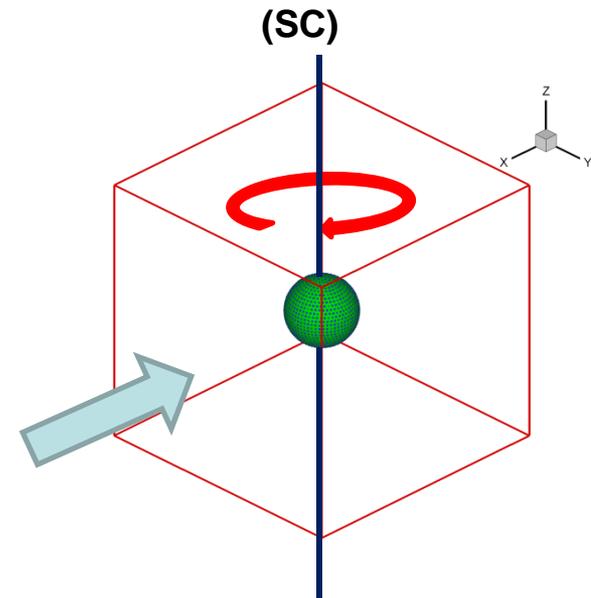
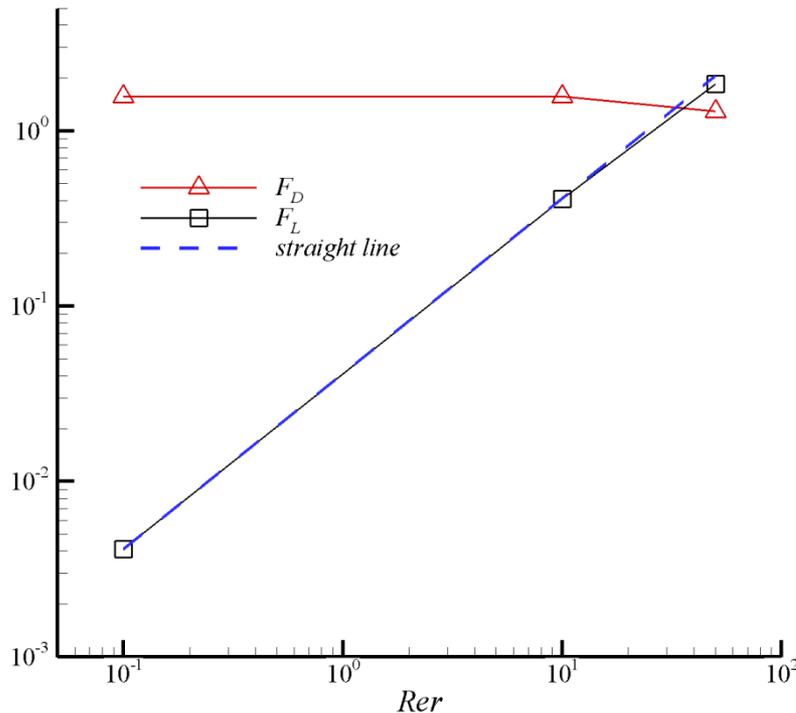


# Drag force and lift force in simple cubic arrays of spheres

## Lift force caused by particle rotation (rotating axis is perpendicular to the flow direction)

- The lift force at various solid volume fraction ( $c$ ).

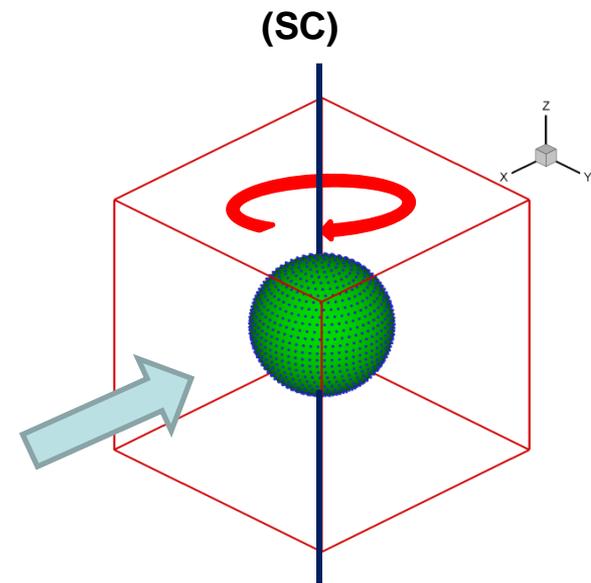
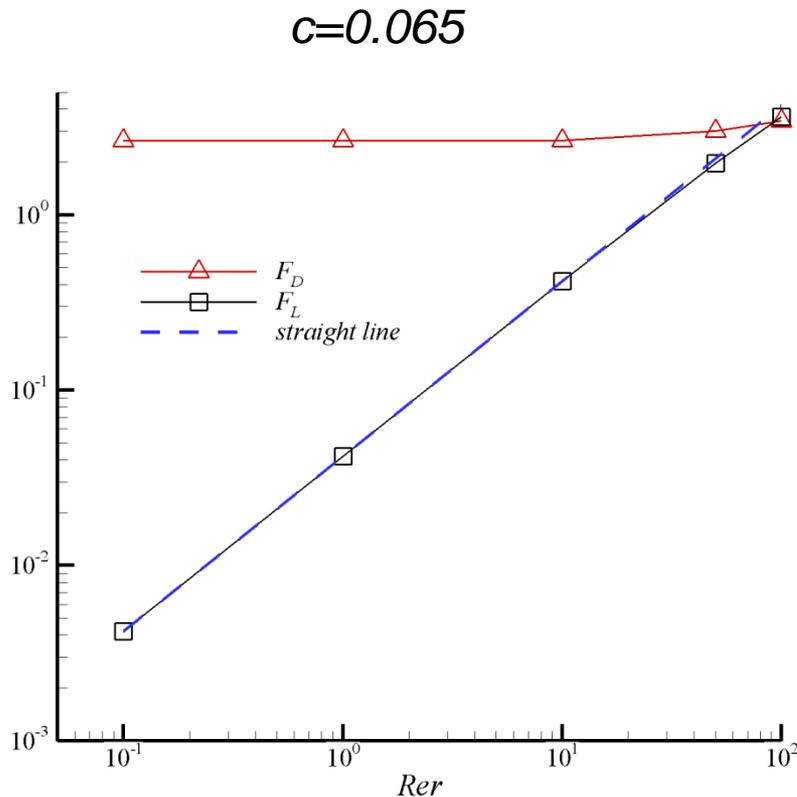
$c=0.01$



# Drag force and lift force in simple cubic arrays of spheres

## Lift force caused by particle rotation (rotating axis is perpendicular to the flow direction)

- The lift force at various solid volume fraction ( $c$ ).

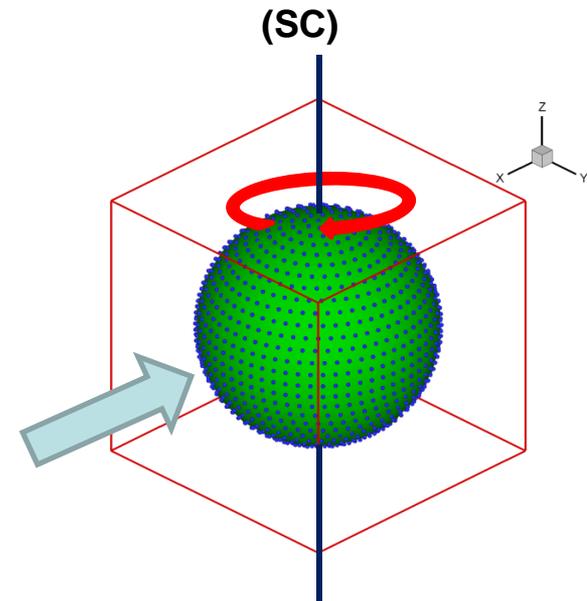
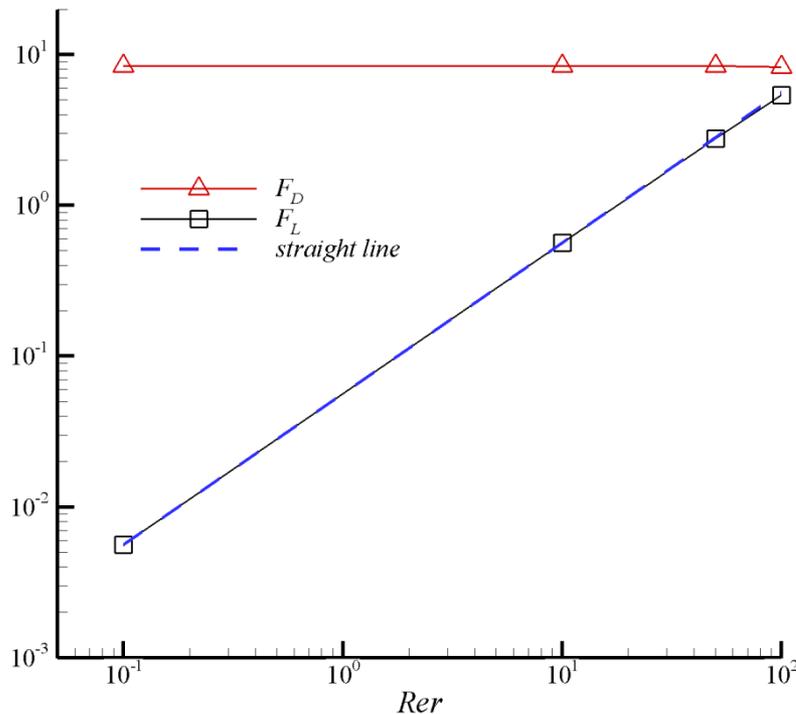


# Drag force and lift force in simple cubic arrays of spheres

## Lift force caused by particle rotation (rotating axis is perpendicular to the flow direction)

- The lift force at various solid volume fraction ( $c$ ).

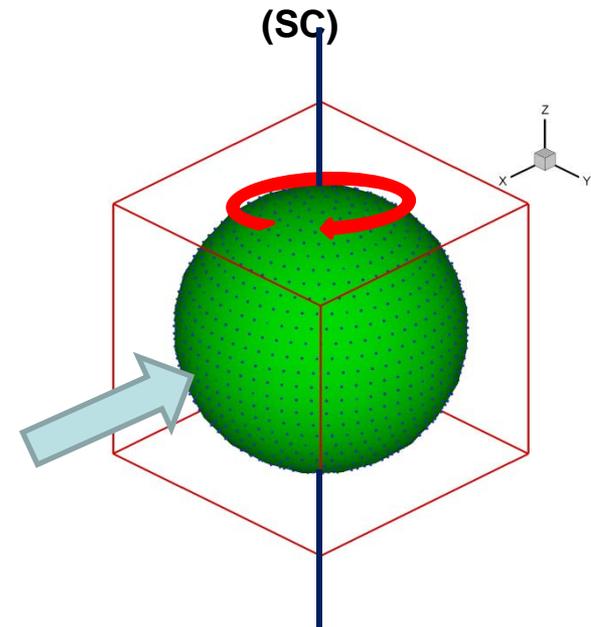
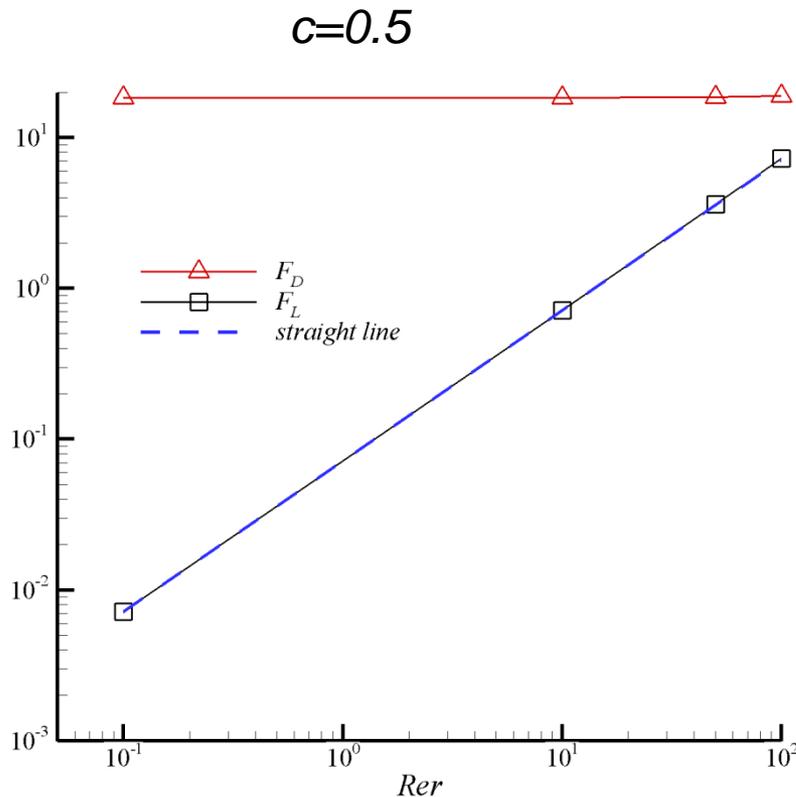
$$c=0.3$$



# Drag force and lift force in simple cubic arrays of spheres

## Lift force caused by particle rotation (rotating axis is perpendicular to the flow direction)

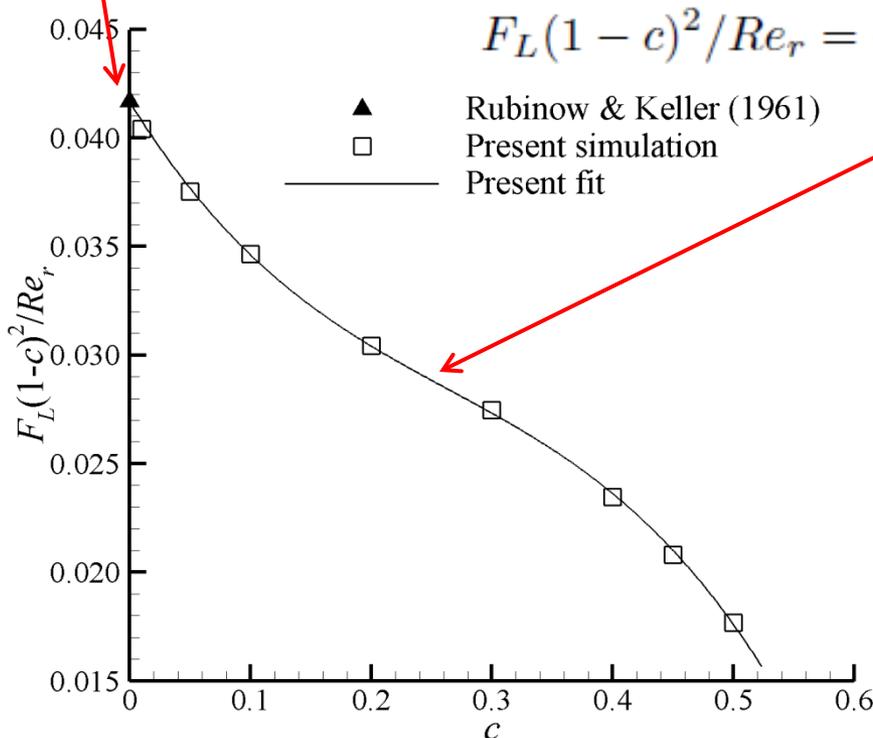
- The lift force at various solid volume fraction ( $c$ ).



# Drag force and lift force in simple cubic arrays of spheres

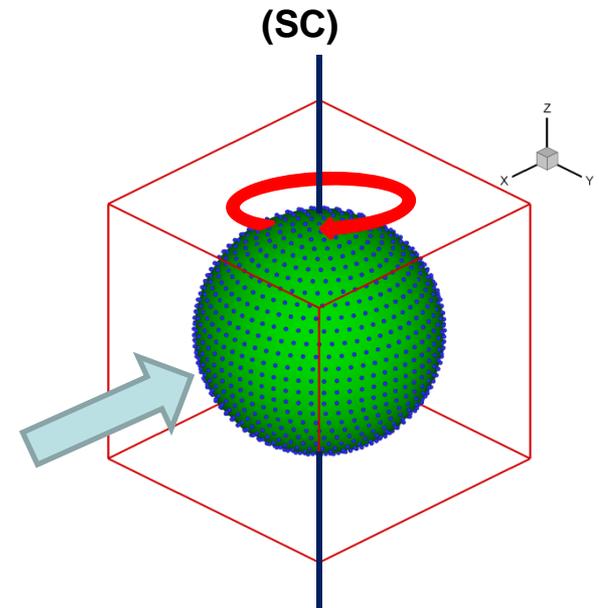
## Lift force caused by particle rotation (rotating axis is perpendicular to the flow direction)

- Based on the simulation results and the theoretical value for  $c \rightarrow 0$  (Rubinow & Keller (1961)). The lift force at various solid volume fraction ( $c$ ) can be expressed as ( $Re_r < 100$ )



$$F_L(1-c)^2/Re_r = (-0.287c^3 + 0.228c^2 - 0.0904c + 1/24)$$

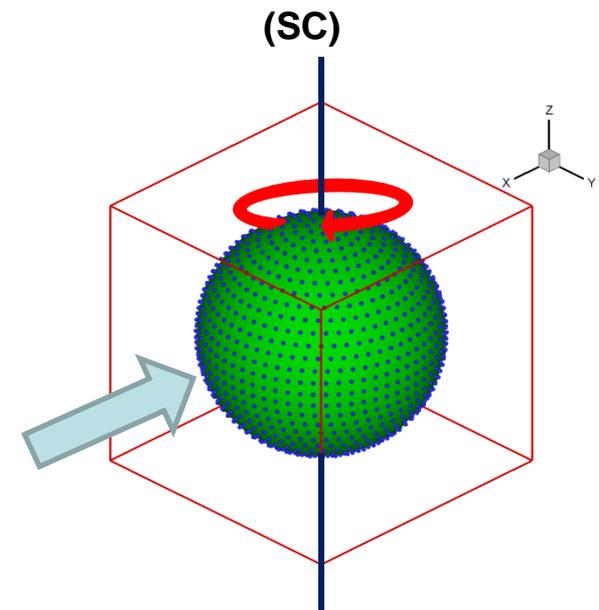
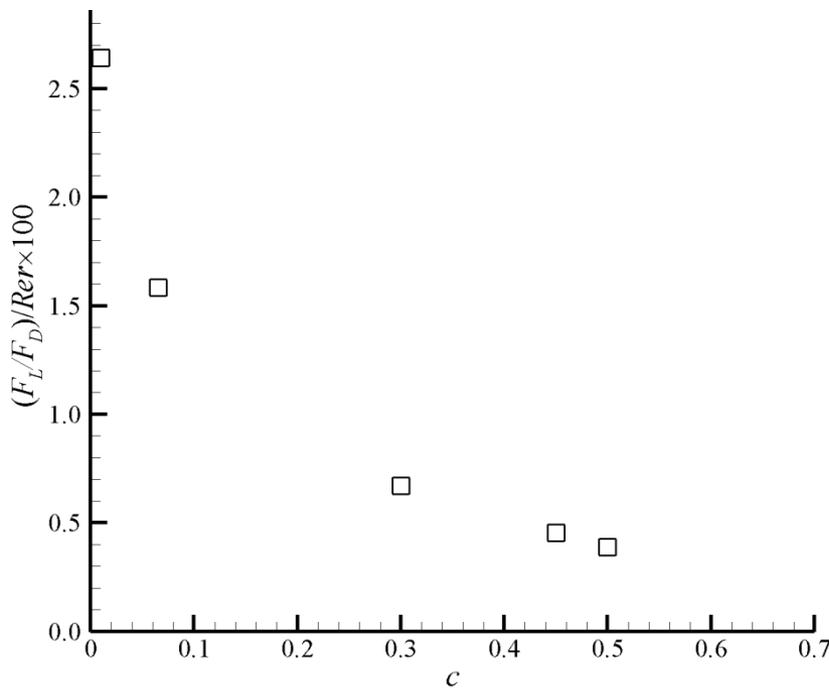
▲ Rubinow & Keller (1961)  
□ Present simulation  
— Present fit



# Drag force and lift force in simple cubic arrays of spheres

## The ratio of the lift force to the drag force

- $c=0.01, \text{Re}r=1, F_L/F_D \approx 2.6\%$ ,  $\text{Re}r=10, F_L/F_D \approx 26\%$ ,  $\text{Re}r=100, F_L/F_D \approx 260\%$
- $c=0.3, \text{Re}r=1, F_L/F_D \approx 0.67\%$ ,  $\text{Re}r=10, F_L/F_D \approx 6.7\%$ ,  $\text{Re}r=100, F_L/F_D \approx 67\%$
- $c=0.5, \text{Re}r=1, F_L/F_D \approx 0.39\%$ ,  $\text{Re}r=10, F_L/F_D \approx 3.9\%$ ,  $\text{Re}r=100, F_L/F_D \approx 39\%$
- For intermediate and dense system, the lift force became significant when  $\text{Re}r > 30$ .



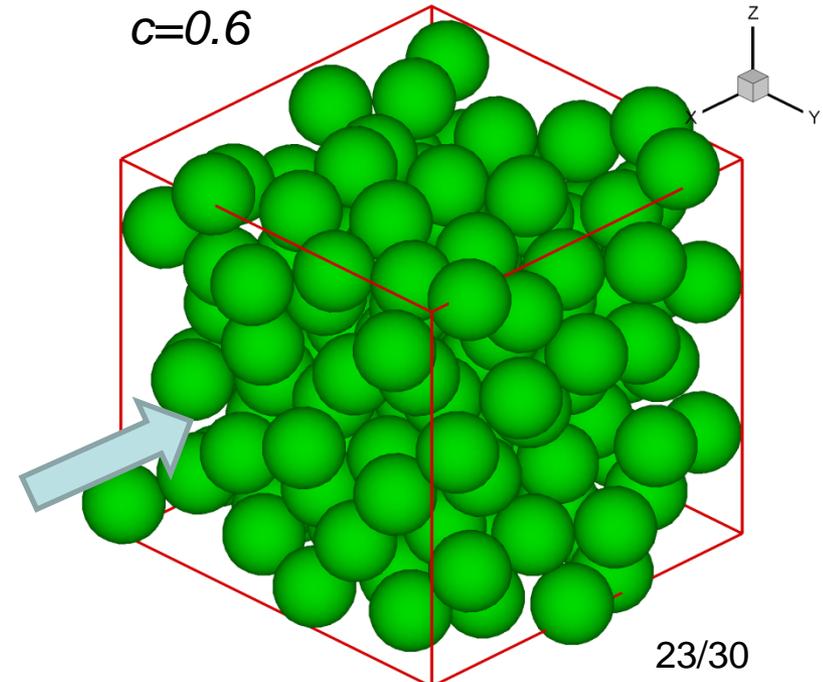
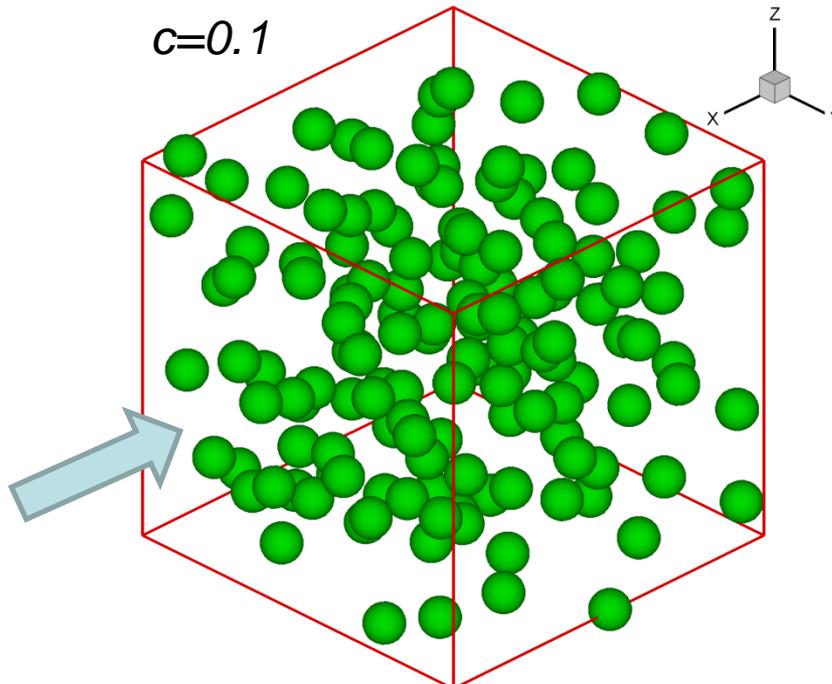
# Outline

- Background: Lift force due to particle rotation
- Background: Immersed boundary-Lattice Boltzmann method, from first order to second order
- Drag force and lift force in simple cubic arrays of spheres
- **Drag force and lift force in random arrays of spheres**
- Open question, how to apply lift force in practical simulations.

# Drag force and lift force in random arrays of spheres

## Random configuration

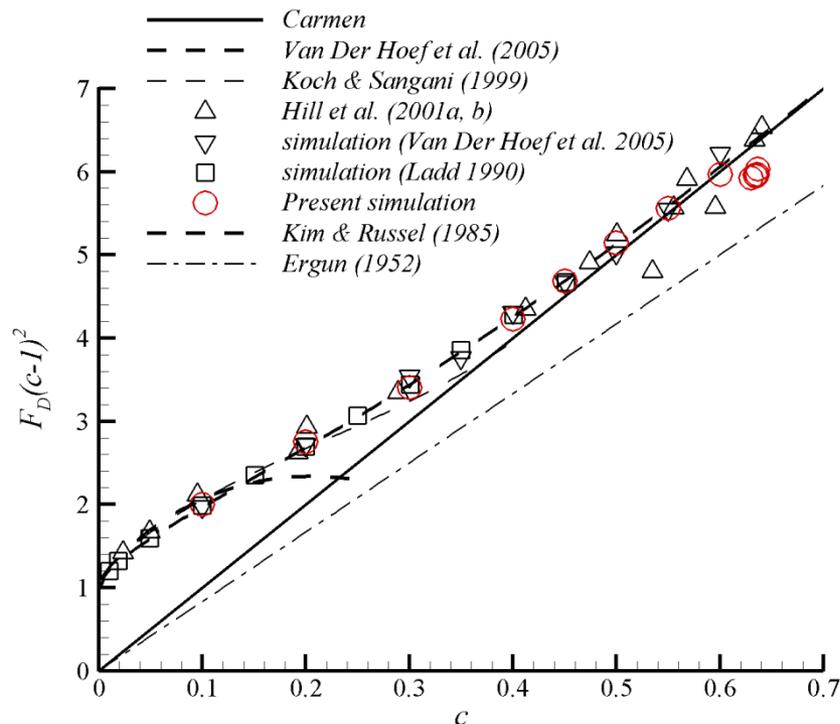
- Random configuration is generated using Zinchenko's method (1994). A standard Monte Carlo procedure may give a crystallized configuration of spheres when  $c$  is large. (*the number of particles: 144*)
- Many independent configurations should be simulated to yield accurate results (*usually more than 5*).



# Drag force and lift force in random arrays of spheres

## Drag force without particle rotation

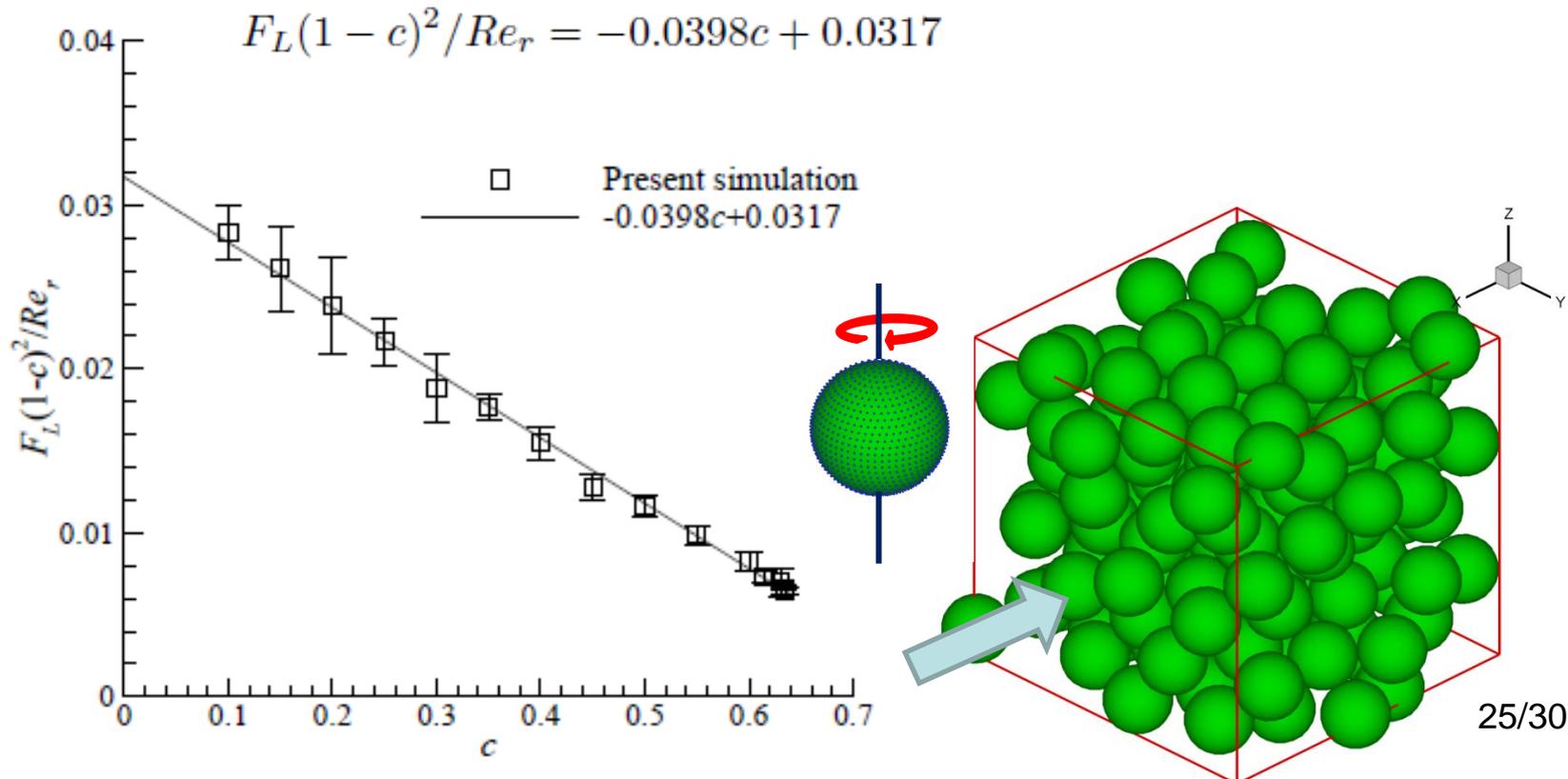
- Agree with previous simulation results as well as the fitting expression proposed by Van Der Hoef et al.(2005)



# Drag force and lift force in random arrays of spheres

## Drag force and lift force in the presence of particle rotation

- $F_L(1-c)^2/Re_r$  is linearly dependent on  $c$ .
- The drag force does not change appreciably.
- Based on our simulation results:



# Drag force and lift force in random arrays of spheres

## Lift-to-drag ratio

- $c=0.1, \text{Rer}=1, F_L/F_D \approx 1.5\%$ ,  $\text{Rer}=10, F_L/F_D \approx 15\%$ ,  $\text{Rer}=100, F_L/F_D \approx 150\%$
- $c=0.3, \text{Rer}=1, F_L/F_D \approx 0.57\%$ ,  $\text{Rer}=10, F_L/F_D \approx 5.7\%$ ,  $\text{Rer}=100, F_L/F_D \approx 57\%$
- $c=0.6, \text{Rer}=1, F_L/F_D \approx 0.15\%$ ,  $\text{Rer}=10, F_L/F_D \approx 1.5\%$ ,  $\text{Rer}=100, F_L/F_D \approx 15\%$
- For  $c < 0.3$ , the lift force became significant when  $\text{Rer} > 10$
- For close packing system, the lift force became significant only when  $\text{Rer} > 50$

# Outline

- Background: Lift force due to particle rotation
- Background: Immersed boundary-Lattice Boltzmann method, from first order to second order
- Drag force and lift force in simple cubic arrays of spheres
- Drag force and lift force in random arrays of spheres
- Open question, how to apply lift force in practical simulations.

# Open question, how to apply lift force?



## How to obtain the rotating speed of the particles?

For discrete particle method and computational fluid dynamics (**DPM-CFD, DEM-CFD**), **The lift force can be added readily**. (collision models, rotating speeds are resolved.)

For the two-fluid method:

*The first approach* (Lun 1991; Jenkins & Zhang 2002)

- Typical two-fluid **governing** equations include, conservation equations for
  - mass
  - translational momentum
  - translational granular temperature
  - rotational momentum**
  - rotational granular temperature**

*The second approach* (Lun 1991; Jenkins & Zhang 2002; Sun & Battaglia 2006)

- Do not add additional equations, let the mean rotating speed of spheres be equal to **half the vorticity of their mean velocity**

# Concluding Remarks

- LBM and IBM is coupled through **Runge-kutta schemes**. The overall accuracy **reaches second-order**.
- The drag force and the lift force are calculated both in ordered arrays and random arrays. The computed drag force is in good agreement with existing theories and published numerical results. Based on the simulation results, lift laws are proposed for the simple-cubic (SC) configuration and random configuration, respectively.
- It is found that **the lift force due to particle rotation can be very significant** relative to the magnitude of the drag force when the rotational Reynolds number is relatively large but within the range of the practical gas-solid flow systems.
- The lift force can be applied in discrete particle method (DPM) readily. Two approaches that can include the lift force in two-fluid simulations are discussed.
- Future work: install the new drag law and lift law into MFIX-DEM to simulate some gas-solid problems in real-world reactors.

# Acknowledgement

- The work is supported by the U.S. Department of Energy Grant FE0007520.